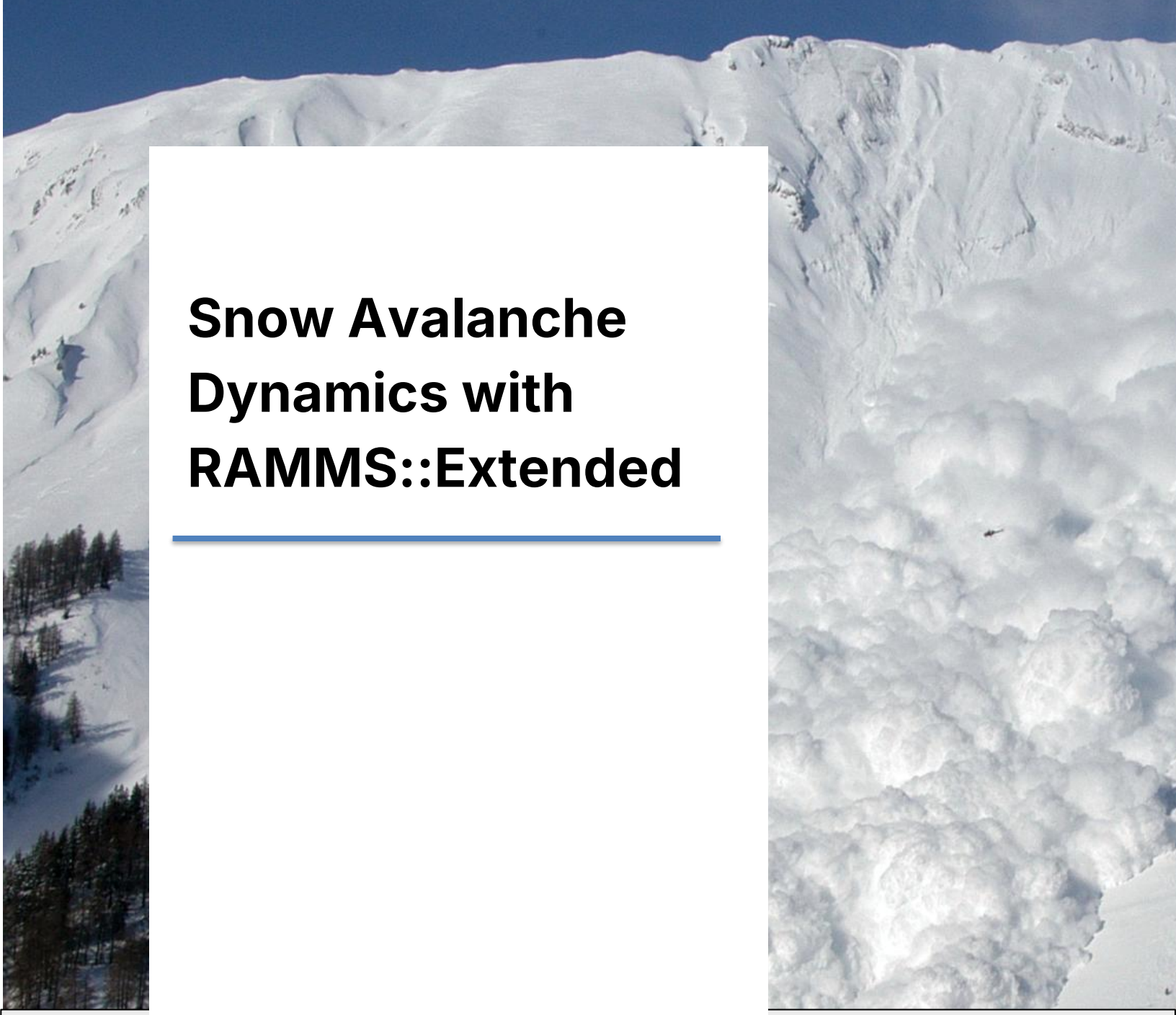


Snow Avalanche Dynamics with RAMMS::Extended

September, 2025

RAMMS AG

Perry Bartelt & Marc Christen

Contents

Chapter 1 Overview

1.1 Introduction to the RAMMS::Extended Model

The **RAMMS::Extended** model is designed to simulate mixed flowing/powder snow avalanches in three-dimensional terrain. Building upon the foundation of **RAMMS::Avalanche**, this advanced model incorporates several key features to enhance the accuracy and detail of avalanche calculations:

1. **Powder Cloud Dynamics**: **RAMMS::Extended** simulates the detachment and propagation of ice-dust clouds that accompany flowing avalanches. This capability allows for precise modeling of snow deposition beyond the dense avalanche core and assessment of high-impact air-blast pressures on structures and vegetation.
2. **Snow Entrainment Modeling**: The model includes mechanisms to simulate snow entrainment, ensuring accurate mass balance and alignment with observational data, particularly in large, erosive avalanche events. The model incorporates algorithms to determine the spatial distribution of snow height, density, and temperature across three-dimensional terrain. This includes a comprehensive representation of three-day new snow accumulation, which is essential for calculating scenarios for 10-year, 30-year, 100-year, and 300-year avalanches.
3. **Environmental Interactions**: **RAMMS::Extended** broadens its applicability in diverse hazard assessments by accounting for interactions with forests, wet snow behavior, and climate-related effects. The model tracks the mean temperature of the avalanche snow, enabling the simulation of flow regime transitions and the onset of melting due to frictional heating.
4. **Advanced Friction Modeling**: **RAMMS::Extended** employs stochastic grain flow theory to simulate the dispersive behavior of the avalanche core. This approach enables the formation of the powder cloud and facilitates more realistic rheological models for the avalanche core, integrating insights from chute experiments and direct field observations. Consequently, the model accurately represents various flow regimes, enhancing the precision of avalanche simulations.
5. **Engineering Applications**: With its enhanced features, **RAMMS::Extended** is better suited for detailed engineering analyses, including impact pressure calculations, infrastructure interactions, and advanced hazard mitigation scenarios.

The **RAMMS::Extended** module is structured around two main components: a **computational kernel** and a **graphical user interface (GUI)**. The GUI retains a user-friendly design like that of the **RAMMS::Avalanche** module, ensuring a smooth transition for experienced users. However, because the **Extended** model incorporates powder cloud simulations with snow entrainment, it introduces additional input and output capabilities. These enhancements include advanced parameter specifications for powder cloud behavior, entrainment characteristics, and more detailed visualization options. Together, these features ensure the **RAMMS::Extended** module provides both

the sophistication and usability needed for modern avalanche simulation and engineering applications.

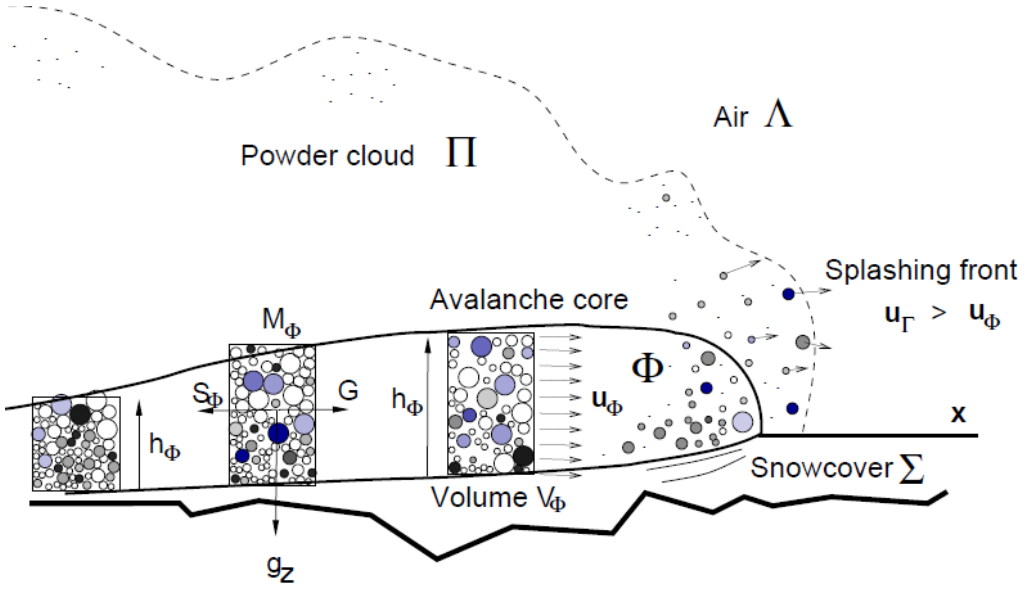


Figure 1.1 : In RAMMS::Extended a mixed flowing/powder avalanche consists of the avalanche core (Φ), the splashing front (Γ) and the powder cloud (Π). The snowcover (Σ) is entrained by the avalanche.

1.2 Six Fundamental Avalanche Components

In **RAMMS::Extended** a snow avalanche consists of **six physical components** (Figure 1.1, Table 1.1). Three of these represent the snow avalanche itself: the **avalanche core** Φ, the **powder cloud** Π, and the pre-front or **splashing front** Γ (which is like a "saltation" layer). In the mathematical model, the front of the avalanche consists of dispersed snow particles, followed by a dense core, all enveloped within a turbulent powder cloud (Figure 1.1). The dynamics of these three components are governed by depth-averaged mass, momentum, and energy balances.

The remaining three components represent terrain and environmental boundary conditions: the **snowcover** Σ, **the surrounding air** Λ, and the **mountain forest** Ψ. The snowcover and air impose mass and thermal energy constraints on the avalanche system. The avalanche's runout behavior is determined by the interaction between its three dynamic components (Φ, Π, Γ) and these three slope/environmental/climatic boundary conditions (Σ, Λ, Ψ).

Symbol	Component	Definition
Φ	**Avalanche Core**	We use the capital Greek letter *Phi* (Φ) to represent the **flowing avalanche** or avalanche core. The core consists of dense snow granules that disperse under shear. This component closely resembles **RAMMS::Avalanche**, serving as the driving force of the avalanche and being strongly influenced by the terrain.
Π	**Powder Cloud**	The Greek letter Pi (Π) denotes the "powder cloud" above the core—'P' for powder. Though less terrain-dependent, it's driven by momentum from the core. Its turbulence is modeled as energy that forms, grows with air entrainment, and dissipates over time.
Γ	**Splashing Front**	The Greek letter Gamma (Γ) represents the "splashing front" in **RAMMS::Extended**, capturing accelerated particles driven by momentum and energy fluxes ahead of the avalanche core—a brief but key feature of its evolving structure.
Σ	**Snowcover**	A key feature of **RAMMS::Extended** is its ability to model snow entrainment, a fundamental aspect of avalanche mass balance. We use the Greek symbol Σ to represent both added mass and the snow cover. Entrainment is modeled as an elastic-plastic collision, where the plastic part captures the incorporation of stationary snow into the moving avalanche core.
Λ	**Air**	The powder cloud is a turbulent suspension of ice-dust particles. Air entrainment, denoted by the Greek letter Lambda (from *Luft*, German for air), dilutes the cloud and controls its height, density, and velocity.
Ψ	**Forest**	**RAMMS::Extended** also simulates avalanche-forest interactions, capturing energy loss from tree impacts, mass reduction from snow interception, and flow changes due to vegetation. The Greek letter Psi (Ψ) is used, as it resembles a tree.

Table 1.1: Definition of the three avalanche components and three boundary conditions. The symbols are used as subscripts in the governing differential equations.

To define the mathematical conditions for the initial state of the calculation: Avalanche simulations are conducted **from initiation to runout** on a three-dimensional terrain (Figure 1.2). The user specifies release zones at designated locations within the calculation domain. At the initial time $t = 0$, the released snow descends the terrain and eventually reaches the deposition zone.

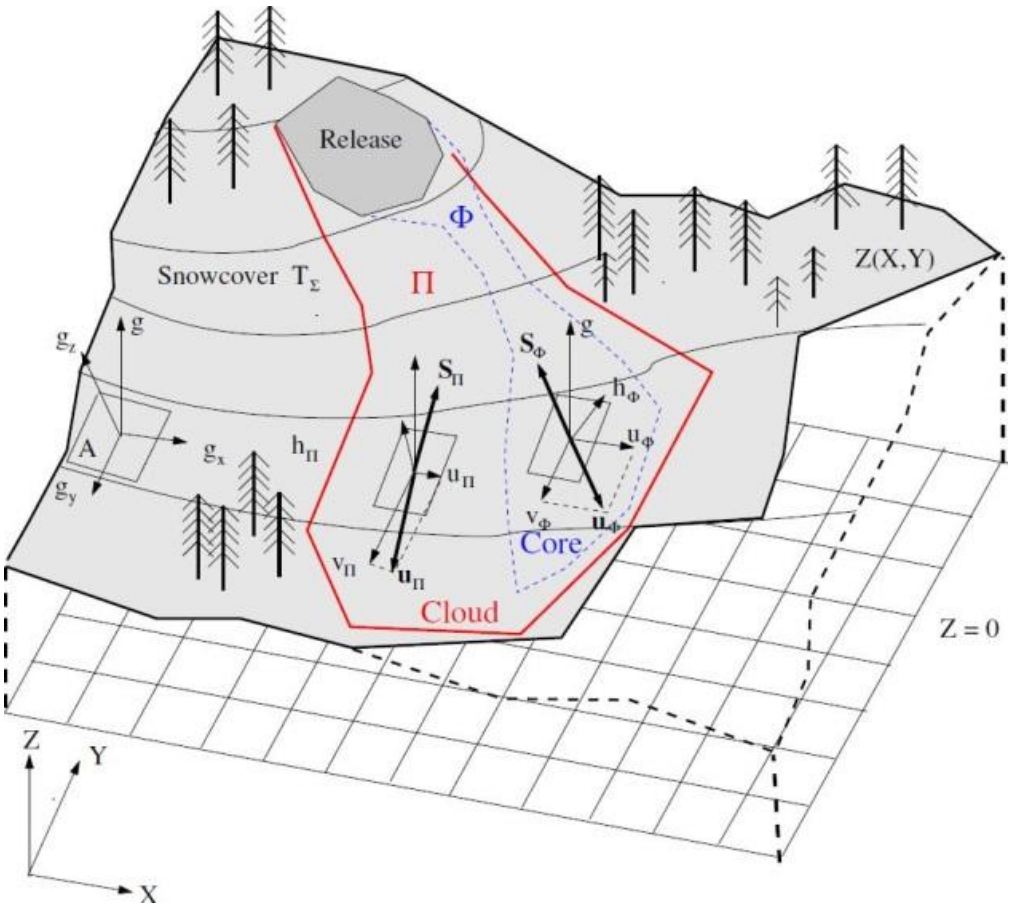


Figure 1.2: A depth-averaged system of equations is solved for the core (Φ) and cloud (Π) on three-dimensional grids, Z(X, Y), to simulate the motion of a mixed flowing/powder avalanche over time and space. The core and cloud can affect different areas, exerting varying pressures on objects within the runout zone.

The primary calculation variables include the **avalanche core's flow height, velocity, density and temperature** (h_Φ, $\vec{u}_\Phi$, ρ_Φ, T_Φ), as well as the **powder cloud's height, velocity and density** (h_Π, $\vec{u}_\Pi$, ρ_Π). The two main avalanche components, the core and the cloud are strongly coupled, but move independently, inundating different areas in the calculation domain (Figure 1.2). **Additional initial/boundary conditions account for the distribution of snow on the mountain slope** at the time of release, including snow cover height, density, and temperature (h_Σ, ρ_Σ, T_Σ).

With the expanded physical components in **RAMMS::Extended**, the graphical user interface has been significantly updated to accommodate new input and output features, enhancing both usability and analytical capabilities. While many concepts from **RAMMS::Avalanche** have been retained—most notably the avalanche return period and the snow fracture heights (d_0 and d_0^*)—the interface now includes additional tools for defining the snowcover and forest boundary conditions.

The **RAMMS::Avalanche** model simulated only the avalanche core **Φ** without the powder cloud or considering entrainment or splashing. In **RAMMS::Avalanche**, forest interaction is accounted for only by adjusting the friction parameters that influence the motion of the avalanche core. However, **RAMMS::Extended** provides more detailed treatment, incorporating mass and momentum loss as forests directly intercept and stop avalanche material. The model can also predict tree destruction caused by the avalanche core or powder cloud. Notably, observations of forest destruction from avalanche air blasts served as a key method for calibrating the magnitude of powder cloud impact pressures.

1.3 The Governing Differential Equations

RAMMS::Extended solves the following system of twelve scaler/vector partial differential equations.

1.3.1 Mass balance avalanche core, co-volume height, $\hat{h}_\Phi$:

$$\left(\hat{h}_\Phi\right)_t + div\left(\hat{h}_\Phi \vec{u}_\Phi\right) = \frac{\rho_\Sigma}{\hat{\rho}_\Phi}\dot{M}_{\Sigma\to\Phi} - \dot{M}_{\Phi\to\Psi} - \frac{\hat{\rho}_\Pi}{\hat{\rho}_\Phi}\,\dot{M}_{\Phi\to\Pi} \tag{1}$$

The **co-volume height** $\hat{h}_\Phi$ represents the **densest possible packing of snow granules**, corresponding to a maximum compaction density $\hat{\rho}_\Phi$. This tightly packed configuration undergoes **dispersive expansion**, giving rise to the actual flow height h_Φ and density ρ_Φ of the avalanche core as it evolves dynamically.

To accurately track mass distribution within the core, the co-volume height $\hat{h}_\Phi$ is employed as a reference variable, governed by three fundamental source terms: **mass entrainment from the snowpack** into the avalanche ($\dot{M}_{\Sigma\to\Phi}$), **detrainment due to interactions with forested terrain** ($\dot{M}_{\Phi\to\Psi}$) and **mass suspension into the powder cloud** ($\dot{M}_{\Phi\to\Pi}$). These interactions dictate the transformation of mass within the system, influencing both the flow regime and overall avalanche dynamics.

When an avalanche **impacts** the snowcover, the interaction produces two distinct outcomes: a portion of the disturbed snow is **entrained** into the avalanche (**plastic interaction**), while the remainder is **splashed forward** (**elastic response**). The rate of mass incorporation, or **entrainment rate**, is denoted as $\dot{M}_\Sigma$. The total mass set into motion by the impact is given by:

$$\dot{M}_\Sigma = \dot{M}_{\Sigma\to\Phi} + \dot{M}_{\Sigma\to\Gamma} = [1-\gamma_\Gamma]\dot{M}_\Sigma + \gamma_\Gamma\dot{M}_\Sigma$$

Here, $\dot{M}_{\Sigma\to\Phi}$ corresponds to the **mass entrained** into the avalanche, while $\dot{M}_{\Sigma\to\Gamma}$ describes the **mass splashed forward**, forming the **pre-front** or **splashing front**. The **partitioning parameter** γ_Γ determines how the disrupted snow is distributed between these two processes.

The total energy dissipated during entrainment, $\dot{L}_\Sigma$, is given by:

$$\dot{L}_\Sigma = \frac{1}{2}[(1-\gamma_\Gamma) + \gamma_\Gamma(1-{r_\Gamma}^2)]\,\dot{M}_\Sigma\|\vec{u}_\Phi\|^2 = \frac{1}{2}[\,(1-\gamma_\Gamma{r_\Gamma}^2)]\dot{M}_\Sigma\|\vec{u}_\Phi\|^2$$

here $\dot{L}_{\Sigma\to\Phi} = \frac{1}{2}[(1-\gamma_\Gamma)]\,\dot{M}_\Sigma\|\vec{u}_\Phi\|^2$ represents the **energy dissipated as heat** during entrainment, while $\dot{L}_{\Sigma\to\Gamma} = \frac{1}{2}[\gamma_\Gamma(1-{r_\Gamma}^2)]\,\dot{M}_\Sigma\|\vec{u}_\Phi\|^2$ accounts for the **kinetic energy lost** through splashing. The **coefficient of restitution**, r_Γ, characterizes the efficiency of the splashing interaction. In our formulation, we assume r_Γ=1, meaning the ejected mass achieves **twice the velocity** of the mean avalanche speed, maximizing the pre-front's forward motion.

1.3.2 Volume (air) balance avalanche core, core height h_Φ:

$$(h_\Phi)_t + div(h_\Phi\vec{u}_\Phi) = \mathbb{D}\left(t, k_z, \dot{k}_z, \ddot{k}_z\right) = 2\dot{k}_z(k_z, \dot{k}_z, \ddot{k}_z) \qquad (2)$$

The function $\mathbb{D}(t, k_z, \dot{k}_z, \ddot{k}_z) = 2\dot{k}_z(k_z, \dot{k}_z, \ddot{k}_z)$ **characterizes the dispersive expansion of the avalanche core** (Eq. 2), capturing its dynamic evolution over time. The factor two in the equation emerges as a direct consequence of the assumption of a **homogeneous particle distribution** within the granular mass (M_Φ) —a condition that ensures symmetry in the statistical treatment of the system. Moreover, **the center-of-mass of the granular ensemble is located in the middle of the volume** (half the core height $k_z = \frac{h_\Phi}{2}$). Importantly, unlike the **RAMMS::Avalanche** model, where the center-of-mass of the granular ensemble follows a strictly slope-parallel trajectory, the **RAMMS::Extended** model allows for non-parallel movements of the avalanche.

This formulation accounts not only for gravitational acceleration perpendicular to the slope (g_z) but also for variations in acceleration induced by internal shearing interactions within the granular ensemble. Two additional differential equations are introduced. **These equations explicitly control the evolution of the center-of-mass' slope-perpendicular velocity $\dot{k}_z$, and dispersive acceleration $\ddot{k}_z$.** (Eqs. 3 and 4):

$$\left(\dot{k}_z\right)_t + div\left(\dot{k}_z \vec{u}_\Phi\right) = \ddot{k}_z \tag{3}$$

$$\left(\ddot{k}_z\right)_t + div\left(\ddot{k}_z \vec{u}_\Phi\right) = \frac{\zeta_\Phi \dot{\mathcal{P}}_\Phi}{M_\Phi} - \left[g_z + \ddot{k}_z\right]\frac{\dot{k}_z}{k_z} \tag{4}$$

As the avalanche transcends steep slopes and accelerates to high velocities, shearing interactions disrupt the coherence of the granular core, leading to a natural tendency for snow particles to disperse. This dispersion is not merely a mechanical consequence of shearing but a process that facilitates the entrainment and acceleration of air, **ultimately giving rise to the formation of the powder cloud**.

Particle dispersion is governed by the energy flux $\zeta_\Phi \dot{\mathcal{P}}_\Phi$, where $\dot{\mathcal{P}}_\Phi$ represents the net mechanical energy flux generated by granular shearing (see below). A portion of this energy manifests as kinetic energy, fueling the random movements of particles within the granular core. However, when this chaotic mechanical energy flux encounters the rigid basal boundary, it is reflected upward. In effect, the initially directionless random energy becomes redirected along the slope-perpendicular axis, ultimately shifting the center of mass of the granular ensemble. The fraction $\zeta_\Phi \dot{\mathcal{P}}_\Phi$ of the energy flux $\dot{\mathcal{P}}_\Phi$ is responsible for driving the expansion of the core, while the remaining portion, $[1 - \zeta_\Phi]\dot{\mathcal{P}}_\Phi$, persists as residual random kinetic energy within the granular ensemble.

The dispersive accelerations within the granular core **occur in sudden, non-equilibrium bursts.** They do not follow a smooth progression but instead experience (spectacular) **overshoots** followed by rapid collapses. This behavior is captured in the term $\left[g_z + \ddot{k}_z\right]\frac{\dot{k}_z}{k_z}$, which represents the exchange of potential and kinetic energy fluxes in the z-direction. Rather than a steady transfer, this exchange reflects the dynamic and transient nature of granular motion, where shifts in energy occur in abrupt, irregular intervals.

1.3.3 Stochastic internal fluctuation energies R_Φ, E_Φ:

RAMMS::Extended considers **two internal energies: Random kinetic energy** (**granular temperature**) R_Φ and internal **heat energy** E_Φ (Eqs 5 and 6). The two forms of fluctuation energy generated by shearing play distinct roles in avalanche dynamics. **Macroscopic fluctuation energy**, denoted as **granular temperature** R_Φ, represents large-scale random kinetic energy associated with the chaotic motion of granules within the flow (Eq. 5). In contrast, **microscopic fluctuation energy** E_Φ corresponds to **heat** arising from internal friction and inelastic particle collisions at the grain scale (Eq. 6). While R_Φ enhances particle agitation, driving dispersion and flow expansion, heat accumulation promotes

melting and lubrication at granular interfaces, altering the avalanche's mechanical properties. Avalanche temperature is simply the scale of internal energy, $E_\Phi = \varrho_\Phi c_\Phi h_\Phi T_\Phi$.

$$\begin{aligned}\left(\hat{h}_\Phi R_\Phi\right)_t + div\left(\hat{h}_\Phi R_\Phi \vec{u}_\Phi\right) \\ = [1-\zeta_\Phi]\left[\alpha_\Phi \dot{W}_\Phi + \rho_\Sigma \varepsilon_\Phi \dot{L}_{\Sigma\to\Phi} - \beta_\Phi \hat{h}_\Phi R_\Phi\right] \\ - \dot{M}_{\Phi\to\Pi} R_\Phi = [1-\zeta_\Phi]\dot{\mathcal{P}}_\Phi - \dot{M}_{\Phi\to\Pi} R_\Phi\end{aligned} \tag{5}$$

$$\begin{aligned}\left(\hat{h}_\Phi E_\Phi\right)_t + div\left(\hat{h}_\Phi E_\Phi \vec{u}_\Phi\right) \\ = [1-\alpha_\Phi]\dot{W}_\Phi + \rho_\Sigma[1-\varepsilon_\Phi]\dot{L}_{\Sigma\to\Phi} + \beta_\Phi \hat{h}_\Phi R_\Phi \\ + \varrho_\Sigma c_\Sigma T_\Sigma \dot{M}_{\Sigma\to\Phi} - \varrho_w L\dot{S}_m - \dot{M}_{\Phi\to\Pi} E_\Phi\end{aligned} \tag{6}$$

The terms on the right-hand side of the equation represent distinct contributions to the rise in internal energy within the avalanche core. These include (1) the energy input from **shear work** ($\dot{W}_\Phi = \vec{S}_\Phi \cdot \vec{u}_\Phi$), (2) the work required to **accelerate snow** from rest ($\rho_\Sigma \dot{L}_{\Sigma\to\Phi}$), (3) the internal energy entrained during **snow erosion** ($\varrho_\Sigma c_\Sigma T_\Sigma \dot{M}_{\Sigma\to\Phi}$), (4) **latent-heat sink** ($\varrho_w L\dot{S}_m$) and (5) the internal energy lost to the **powder cloud** ($\dot{M}_{\Phi\to\Pi} F_\Phi$) as mass is transferred from the core to cloud. Each of these terms plays a role in shaping the energy distribution within the avalanche, influencing both its bulk motion and its microscopic thermodynamics.

With these definitions in place, we can now state **the random mechanical energy flux**,

$$\dot{\mathcal{P}}_\Phi = \alpha_\Phi \dot{W}_\Phi + \rho_\Sigma \varepsilon_\Phi \dot{L}_{\Sigma\to\Phi} - \beta_\Phi \hat{h}_\Phi R_\Phi$$

The model parameters α_Φ, β_Φ, and ε_Φ in Eqs. 5 and 6 act as **partitioning parameters**, governing how the total random mechanical energy is distributed between granular agitation and heat dissipation.

The coefficient α_Φ determines how shear-generated energy is split between **random kinetic energy** (granular temperature) and **heat**, while ε_Φ controls the partitioning of energy during entrainment. The **decay coefficient** β_Φ regulates the rate at which granular fluctuations dissipate into heat, influencing how quickly the avalanche loses its ability to sustain internal agitation.

The parameters α_Φ and ε_Φ impose a fundamental separation between the roles of **shearing and entrainment** in the generation of granular fluctuations. Rather than merely

reflecting different physical mechanisms, they **define** spatially distinct dynamical regimes within the avalanche (Figure 1.3).

In **RAMMS::Extended**, this separation is what gives rise to a highly fluidized, turbulent front—ε_Φ governs the generation of random kinetic energy during entrainment—and a dense, shearing tail, where α_Φ controls the production of granular agitation through internal deformation.

This division is not incidental but intrinsic to the avalanche's spatial evolution: the front, continuously energized by entrainment, expands and disperses, while the tail, driven by internal shearing, remains more cohesive. The avalanche **thus self-organizes** into a structure governed by the interplay of these competing energy fluxes, illustrating how non-equilibrium processes dictate large-scale avalanche flow behavior, including the formation of the powder cloud at the avalanche front.

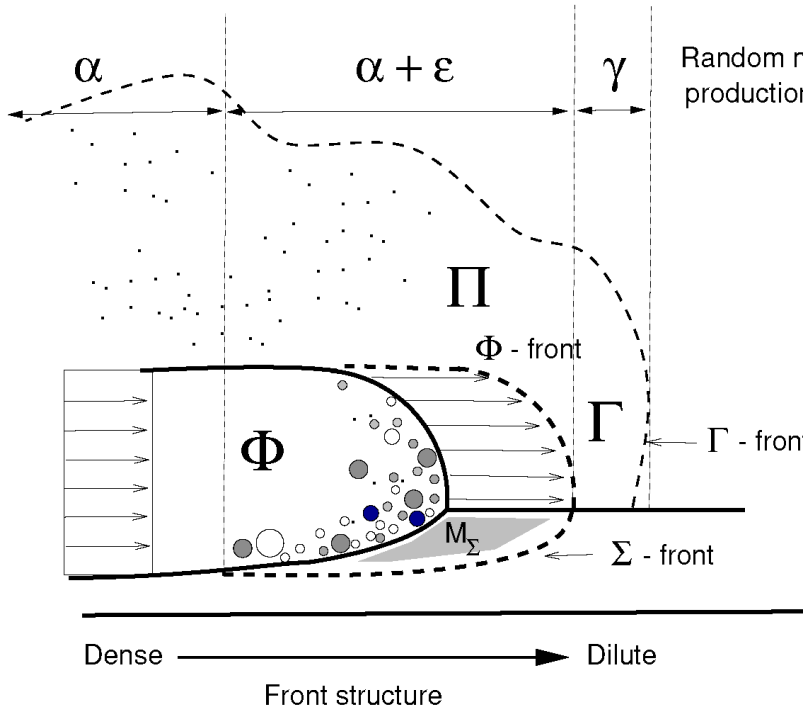


Figure 1.3: The structure of the avalanche front in RAMMS::Extended is controlled by three partitioning parameters (α_Φ, ε_Φ, γ_Γ).

Conversely, in the avalanche tail, shearing serves as the dominant mechanism of energy redistribution. The coefficient α_Φ governs the **generation** of granular fluctuations. Here, energy input remains localized within the internal structure of the flow, producing a dense, shearing mass that retains a higher degree of cohesion. However, the persistence of this energy, particularly in the highly fluidized avalanche front—is determined by the coefficient β_Φ, which regulates the **decay** of random kinetic energy.

The role of β_Φ is crucial: it dictates how quickly granular agitation dissipates into heat, thereby controlling how long the avalanche front can sustain itself as a dispersive flow. A **low** β_Φ value allows fluctuations to persist, extending the lifetime of the fluidized front and enhancing the energy available for suspension, **ultimately intensifying the powder cloud**. Conversely, a **higher** β_Φ accelerates dissipation, causing the front to lose its dispersive character more rapidly, reducing both the extent and impact of the powder cloud. In **RAMMS::Extended**, β_Φ is modeled as a function of avalanche temperature T_Φ, enabling the prediction of transitions between temperature-driven dry and wet avalanche flow regimes.

Recalling that the fraction $[1-\zeta_\Phi]\dot{\mathcal{P}}_\Phi$ remains within the granular ensemble as residual random kinetic energy (Eq. 5), while the fraction $\zeta_\Phi\dot{\mathcal{P}}_\Phi$ actively drives the dispersive expansion of the avalanche core (Eq. 4). The parameter ζ_Φ serves as a partitioning factor, distinguishing between directionless (random) mechanical energy and the portion that is redirected in the slope-perpendicular direction, contributing to the core's expansion.

Thus, while α_Φ and ε_Φ define the **generation** of random kinetic energy in distinct flow regions, β_Φ governs its **longevity**, shaping the avalanche's overall evolution. The parameter ζ_Φ reflects the **interplay of the granular ensemble with the basal boundary**. The interplay of these parameters determines not only the internal structure of the flow but also the transition between dense, shearing motion and the formation of an airborne, turbulent suspension.

The competition between these two energy forms—granular fluctuations R_Φ and heat E_Φ—defines avalanche behavior across all scales. The values of these coefficients are closely tied to the **flow regime**, as they dictate the transition between different flow states, influence rheology, and ultimately determine whether the avalanche remains a dense, shearing mass or **transforms into a mixed flowing/powder avalanche**.

1.3.4 Momentum balance avalanche core, core velocity $\vec{u}_\Phi$

$$\left(\hat{h}_\Phi\vec{u}_\Phi\right)_t + div\left(\hat{h}_\Phi\vec{u}_\Phi\otimes\vec{u}_\Phi + p_\Phi I\right) = \vec{G}\hat{h}_\Phi - \frac{\vec{u}_\Phi}{\|\vec{u}_\Phi\|}S_\Phi(R_\Phi, T_\Phi, m_\Phi) - \left[(1+r_\Gamma)\frac{\rho_\Sigma}{\hat{\rho}_\Phi}\dot{M}_{\Sigma\to\Gamma} + \dot{M}_{\Phi\to\Psi} + \frac{\hat{\rho}_\Pi}{\hat{\rho}_\Phi}\dot{M}_{\Phi\to\Pi}\right]\vec{u}_\Phi \quad (7)$$

The **velocity of the avalanche core** (Eq. 7) is governed by the dynamic interplay between the slope-parallel gravitational components $\vec{G} = (g_x, g_y)$ and internal shearing forces $S_\Phi(R_\Phi, T_\Phi, m_\Phi)$.

RAMMS::Extended employs a dynamic approach to model shear resistance, building upon the foundational principles of the **Voellmy model**:

$$S_\Phi = \mu(R_\Phi, T_\Phi, m_\Phi)N_\Phi + \frac{\varrho g U_\Phi^2}{\xi(R_\Phi, T_\Phi, m_\Phi)}$$

In this formulation, the **friction coefficients μ and ξ** are functions of the granular temperature (R_Φ), snow temperature (T_Φ) and moisture content (m_Φ).

These coefficients are **dynamically adjusted** during simulations to account for **variations in fluidization, temperature, and moisture of the flowing snow**. The rheological model distinguishes between baseline values of the Voellmy parameters (μ_0, ξ_0) and their dynamic counterparts (μ, ξ), which evolve in response to changes in R_Φ, T_Φ, and m_Φ throughout the simulation. This approach captures the dynamic nature of shear resistance under varying avalanche conditions, encompassing both dry and wet scenarios, aligning with Voellmy's original concepts. Furthermore, this approach enables a direct comparison with the well-established **RAMMS::Avalanche** model, **leveraging decades of empirical knowledge and practical application in Switzerland**. By employing the widely recognized Voellmy friction law—long used as a standard in Swiss avalanche modeling—we ensure that the determination of friction parameters is grounded in **extensive experience and validated methodologies**. This foundation of accumulated expertise enhances both the reliability and accuracy of **RAMMS::Extended**.

A deeper exploration of these shearing dynamics, especially the recommended values of (μ_0, ξ_0) is the focus of the following chapter 3, which is dedicated to the **RAMMS::Extended** shearing model.

Momentum loss within the avalanche core arises from **three additional mechanisms**. First, the formation of the **splashing front**, represented by the mass flux $\dot{M}_{\Sigma\to\Gamma}$, which involves an elastic-plastic collision between the avalanche core and the snow cover. Second, **forested terrain acts as a braking force**, introducing momentum loss via $\dot{M}_{\Phi\to\Psi}$, where interactions with trees slow the avalanche and redistribute energy. Finally, a critical **momentum transfer occurs between the avalanche core and the powder cloud $\dot{M}_{\Phi\to\Pi}$.** This process is particularly significant, as it is responsible for energizing and expanding the powder cloud, ultimately endowing it with its destructive potential.

1.3.5 Moisture content (wetness) of the avalanche core, m_Φ

In the **RAMMS::Extended** module, an additional **mass transport equation** has been implemented to explicitly account for **the moisture content bound to snow granules**. This moisture component may be initialized at the start of a simulation—either as a property of the **release zone** or as part of the **snowcover** itself. Over the course of the avalanche flow, the moisture content can dynamically evolve, increasing in response to frictional heating generated by shear forces within the moving snow mass.

$$(m_\Phi)_t + div(m_\Phi \vec{u}_\Phi) = \dot{S}_m + \frac{\varrho_w}{\hat{\rho}_\Phi}\phi_{\Sigma w}\dot{M}_{\Sigma\to\Phi}$$

where

$$\int_{\Delta t}\dot{Q}_m dt = \varrho_\Phi c_\Phi h_\Phi [T_\Phi - T_m] \qquad \dot{S}_m = \dot{Q}_m/\varrho_w L \tag{8}$$

The **moisture content of the core** m_Φ (Eq. 8) has dimension [m], and thus it is similar to the core height variable $\hat{h}_\Phi$ or h_Φ. When the mean temperature of the avalanche core, T_Φ, rises above the melting point T_m, a **phase transition** occurs, initiating the formation of liquid water $\dot{S}_m$. As a result, meltwater begins to **accumulate on the surfaces of individual snow granules** (Figure 1.4), altering the rheology of the flow. Water can also enter the avalanche by **entraining wet snowcovers** ($\phi_{\Sigma w}$= volumetric water content of entrained snow).

The **RAMMS::Extended** model incorporates a meltwater conservation equation to track the distribution and movement of this liquid phase within the avalanche. The energy required for melting is governed by the **latent heat of fusion**, L, which dictates the rate at which solid ice transforms into water under the influence of frictional heating and compressive forces within the core.

The temperature T_Φ represents the **mean depth-averaged temperature** of the avalanche core. However, due to intense granular shearing and frequent particle collisions, heat is not uniformly distributed throughout the flow. Instead, frictional heating is concentrated at the surfaces of individual snow clods, where energy dissipation is most intense. As a result, melting can begin even before the mean temperature of the core reaches the physical melting point of ice, $T_m = 0°\mathrm{C}$.

In **RAMMS::Extended**, we account for this effect by **setting an empirical melting threshold** of $T_m = -1°\mathrm{C}$. The difference between this empirical threshold and the physical melting point serves as a **measure of the heterogeneity of heat distribution** within the avalanche. Rather than warming uniformly, the granular nature of the flow ensures that **heat generation is localized** at the particle interfaces, where shear forces are highest. This

localized heating leads to the immediate formation of a thin water film on the granule surfaces (Figure 1.4), even if the bulk temperature of the snow remains below freezing. This effect fundamentally alters the flow properties of the avalanche, influencing its mobility, viscosity, and overall dynamics.

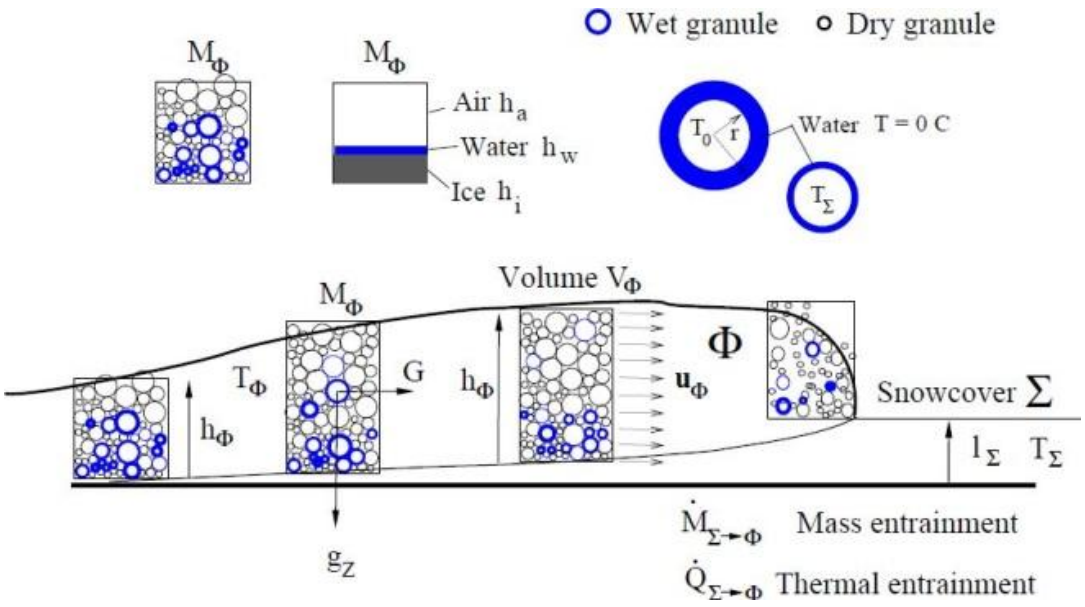


Figure 1.4: Temperatures on the granule surfaces can reach the melting temperature even before the mean temperature has reached the temperature $T_\Phi = 0$ °C.

1.3.6 Mass balance avalanche cloud, co-volume height, $\hat{h}_\Pi$

$$\left(\hat{h}_\Pi\right)_t + div\left(\hat{h}_\Pi \vec{u}_\Phi\right) = \dot{M}_{\Phi\to\Pi} \tag{9}$$

The co-volume height $\hat{h}_\Pi$ serves as a measure of the **total mass of ice-dust** suspended within the avalanche cloud, normalized to the reference density $\hat{\rho}_\Pi$. This density represents the starting, or "seeding" density of the cloud which is formed within the avalanche core.

Crucially, **the avalanche core is the sole source of ice-dust** within the cloud $\dot{M}_{\Phi\to\Pi}$. Unlike the core, which actively entrains mass from the snow cover, the cloud does not directly draw ice-dust from the terrain. Instead, snow must first be incorporated into the core, where it undergoes mechanical fragmentation before being suspended into the cloud. The

interaction between the core and cloud—characterized by shearing, collisions, and turbulent mixing—breaks down the snow cover into fine ice-dust particles, which are then lofted into suspension. This process defines the mass exchange between the core and the cloud, determining the rate at which the cloud gains material.

1.3.7 Volume (air) balance avalanche cloud, cloud height, h_Π

$$(h_\Pi)_t + div(h_\Pi \vec{u}_\Pi) = \frac{(\rho_i - \hat{\rho}_\Pi)}{(\rho_i - \rho_\Lambda)} \dot{M}_{\Phi\to\Pi} + \dot{M}_{\Lambda\to\Pi} \quad (10)$$

The avalanche cloud acquires air from two distinct sources. The first is the air fraction contained within the mass of ice-dust that becomes suspended in the cloud—**air that is carried along as part of the mixture ejected from the avalanche core**, $\dot{M}_{\Phi\to\Pi}$. The second source is the entrainment of additional **ambient air**, drawn in as the cloud expands and mixes with its surroundings $\dot{M}_{\Lambda\to\Pi}$. This continuous intake of air fuels the cloud's growth, influencing its density, velocity, and overall dynamics. It is important to note that **no reduction mechanisms, such as the cloud settling to the ground**, are accounted for in the extended formulation.

1.3.1 Momentum balance avalanche core, powder velocity $\vec{u}_\Pi$

$$\left(\hat{h}_\Pi \vec{u}_\Pi\right)_t + div\left(\hat{h}_\Pi \vec{u}_\Pi \otimes \vec{u}_\Pi + p_\Pi I\right) = \frac{(\hat{\rho}_\Pi - \rho_\Lambda)}{\hat{\rho}_\Pi} \vec{G} \hat{h}_\Pi + \dot{M}_{\Phi\to\Pi} \vec{u}_\Phi - \frac{\vec{u}_\Pi}{\|\vec{u}_\Pi\|} S_\Pi(R_\Pi) \quad (11)$$

The slope-parallel velocity of the powder cloud $\vec{u}_\Pi$ is governed primarily by the **momentum imparted by the avalanche core**, $\dot{M}_{\Phi\to\Pi}\vec{u}_\Phi$. Acting as a momentum engine, the core draws in ambient air (if the avalanche front is dispersed), saturating it with fine ice-dust particles and accelerating the mixture to near-core velocity. This initial suspension, once formed, rapidly entrains additional air $\dot{M}_{\Lambda\to\Pi}$, driving a rapid increase in cloud height as the interface expands upwards and outwards.

Beyond the direct momentum transfer from the core, an additional force acts on the cloud—**namely gravity**, represented by $\vec{G}$. However, because the cloud's bulk density is only slightly higher than that of the surrounding air, it behaves as a buoyant suspension, where the submerged weight of the ice-dust mixture must be considered, $(\hat{\rho}_\Pi - \rho_\Lambda)$. This buoyant effect influences the cloud's motion but plays **a secondary role** compared to the dominant momentum injection from the core.

Given these dynamics, we treat the avalanche cloud as an **inertial flow**—a mass of air and ice-dust propelled forward by an initial burst of momentum from the core. Unlike the core, which remains in contact with the terrain and continues to accelerate downslope, the cloud becomes self-sustaining once airborne, capable of propagating over large distances. Without continued input from the core, however, it gradually slows as drag forces $S_{\Pi}(R_{\Pi})$ dissipate its energy over time. Though capable of traveling far beyond the limits of the core, the cloud ultimately disperses as its energy spreads and weakens, marking the final stage of its evolution.

1.3.8 Turbulent energy balance of the powder cloud, R_{Π}

$$\left(\hat{h}_{\Pi}R_{\Pi}\right)_t + div\left(R_{\Pi}\hat{h}_{\Pi}\vec{u}_{\Pi}\right) = \dot{M}_{\Phi\to\Pi}R_{\Phi} + [\hat{\rho}_{\Pi}S_{\Pi}]\|\vec{u}_{\Pi}\| + \frac{1}{2}\rho_{\Lambda}\dot{M}_{\Lambda\to\Pi}u_{\Pi}^2 - \beta_{\Pi}\hat{h}_{\Pi}R_{\Pi} \quad (12)$$

This equation encapsulates the balance between the **generation and dissipation of turbulence**, providing a framework for predicting cloud drag and air-blast evolution with precision. Turbulent energy in the powder cloud originates from three primary source terms: **direct energy transfer from the avalanche core** $\dot{M}_{\Phi\to\Pi}R_{\Phi}$, the **shearing work** within the cloud $[\hat{\rho}_{\Pi}S_{\Pi}]\|\vec{u}_{\Pi}\|$, and the energy imparted during the **entrainment of ambient air** $\frac{1}{2}\rho_{\Lambda}\dot{M}_{\Lambda\to\Pi}u_{\Pi}^2$.

A key assumption underpinning our model is that both shearing work and entrainment energy are entirely converted into turbulence, without any immediate dissipation into heat. In other words, the **turbulence production mechanisms are assumed to be maximally efficient**, with no intermediate thermal losses. Furthermore, we postulate that all three source terms contribute to the generation of large-scale turbulence, which cascades down to smaller eddies before ultimately dissipating as heat through the term $\beta_{\Pi}\hat{h}_{\Pi}R_{\Pi}$, where β_{Π} **represents the turbulence-to-heat decay coefficient**. This assumption aligns with the classical energy cascade model, wherein turbulence is first generated at large scales before progressively breaking down into finer structures, culminating in molecular dissipation.

1.4 Snowcover Parameters

One of the defining strengths of the **RAMMS::Extended** module is its ability to simulate dynamic interactions between avalanches and the existing mountain snowcover. In contrast to **RAMMS::Avalanche**, where the snowcover is not explicitly defined and snow entrainment is excluded, **RAMMS::Extended incorporates a physically realistic**

snowcover model. This model is grounded in **observed meteorological data**, **extreme value statistics**, and **methodologies** developed for Swiss **hazard mapping procedures**. These procedures are specifically designed to estimate the runout of avalanche events for a specific return period.

At the core of this approach is the methodology that links **measured three-day snow accumulation (**$\Delta HS_{3\mathrm{d}}$) with **extreme value statistics** to characterize the snowcover during winter storms. This correlation forms the basis for estimating both the **fracture depth** of release zones and the **depth of the erodible snow layer**—the snow available for entrainment. The definition of the erodible snow layer in **RAMMS::Extended** thus follows the same **fundamental strategy** used in Switzerland to construct avalanche hazard maps.

The erodible snow depth h_{Σ}, is first determined at a reference elevation based on measured three-day snow accumulation. This value is adjusted for **elevation differences** between **the measurement station** and the **release zone**, as well as for slope angle, since observations are typically made on flat terrain. To distribute this reference depth across the model domain, snow depth gradients (**Δ**h_{Σ}) are applied. Additional corrections account for **complex terrain features**—such as **cliffs**—and surface conditions, including vegetation effects like **snow interception by mountain forests** (Figure 1.5,Table 1.2).

Temperature plays a pivotal role in **RAMMS::Extended** simulations. Because **avalanche flow regimes are temperature-dependent**, thermal boundary conditions must be carefully defined. In many cases, the initial flow regime is governed by the **temperature of the erodible snow layer** at the time of release. These temperatures are specified at a **reference elevation**, based on **measured storm temperatures**, and are then adjusted using **temperature-elevation gradients** to reflect warming as elevation decreases. This gradient-based approach allows the model to simulate snow temperature variations across different terrain features and elevations. As the avalanche descends and entrains warmer snow, it may undergo a transition from **cold, dry flow** to **warm, wet flow**, significantly altering flow behavior, mobility, and deposition patterns. This feature enables **RAMMS::Extended** to account for **terrain aspect** and varying **climatic regimes**, such as **maritime (wet)** or **continental (dry)** snow climates, enhancing the realism of complex avalanche scenarios.

In **RAMMS::Extended**, snowcover is defined using **measured three-day snowfall accumulation** linked to a specified return period. A challenge with this approach is that **snow density** in the release zone and within the erodible layer is not directly measured and must be derived. To address this, **RAMMS::Extended** employs analytical snow density models that estimate density based on **snowfall amount**, **temperature**, and **slope angle (**Figure 1.5**)**. These models allow the density of the erodible snow layer to be calculated consistently across the entire model domain. Presently **wind-blown snow is accounted for only in the release zone**.

The final climatic variable in **RAMMS::Extended** is the **air temperature** T_{Λ}**. Cold air temperatures support the formation of powder avalanches**, while warmer air

can trigger transitions from dry to wet flow regimes. This parameter was introduced to enable the simulation of **high-elevation avalanches** and to support the analysis of **climate change impacts** on avalanche behavior.

1.5 Model Parameters

While **boundary conditions** define the external environment—such as snowcover, terrain, and weather—**model parameters** govern the **internal physics of avalanche motion**. These include how mass, momentum, and energy evolve during flow and are given by the model differential equations. Model parameters are related to the three avalanche components: **core**, **cloud** and **splashing front**.

Although conceptually distinct, the boundary conditions and model parameters are closely linked. For example, snow temperature (a boundary condition) directly influences flow behavior by affecting friction and fluidization (model parameters). Likewise, forest cover impacts entrainment and drag, altering parameters tied to mass and momentum balances and flow properties of the avalanche.

Model parameters in **RAMMS::Extended** fall into three main categories.

- The first are **user-defined parameters**, which can be selected within a specified range to tune the simulation to a particular scenario.
- The second are **constants** or **temperature-dependent parameters**, which are **automatically set** as they are **constants,** or **based entirely on the snow temperature.** These parameters reflect internal processes—such as phase transitions or fluidization thresholds—and do not require user input beyond specifying temperature.
- The third set of parameters are **scenario recommendations** linked to avalanche return period.

The Table 1.3 lists all model parameters used for both the **dense core, powder cloud and splashing front**.

- Parameters shown in **green** can be selected and adjusted by the user.
- Parameters shown in **orange** are determined internally by the model and **are not user editable**.
- Parameters shown in **yellow** are **scenario-based parameters**.

The table indicates that there are **five significant user adjustable parameters** (one for the core, four for the cloud). There are **six non-editable parameters** (constants) and two sets of **scenario-based parameters**, the Voellmy parameters for the avalanche core and drag for the powder cloud. Each green field corresponds to an input field in the **RAMMS::Extended** graphical user interface.

Snowcover Parameter	Symbol	Comments
Erodible Snow Depth	h_Σ	The **erodible snow depth** is initially defined at a reference elevation based on measured three-day snow accumulation, and then automated algorithms distribute this depth across the entire model domain.
Snow Density	ϱ_Σ	In **RAMMS::Extended**, analytical snow density models are used to estimate the density of the erodible layer as a function of elevation, temperature, and snowfall amount.
Snow Temperature	T_Σ	The **temperature** of the erodible snow layer at the reference elevation, typically based on measured storm temperatures and used to determine the initial avalanche flow regime.
Snow Depth Gradient	Δh_Σ	The **snow depth gradient** accounts for the decrease in snow depth with elevation and is derived from winter storm data specific to the regional climate.
Snow Temperature Gradient	ΔT_Σ	The **rate of temperature increase** with decreasing elevation, used to adjust snow temperature throughout the model domain and account for terrain and climatic variation.
Air Temperature	T_Λ	**Air temperature** influences flow regime transitions and powder cloud formation.
Forest Interaction	K_Ψ	**Tree interception** can significantly reduce snow accumulation in mountain forests and alter the erodibility of the snow that accumulates in the forest.

Table 1.2: Snowcover properties in RAMMS::Extended for defining avalanche scenarios, including parameters for snow depth, density, temperature, and spatial distribution.

1.6 Numerical Algorithms in RAMMS::Extended

The numerical solvers in **RAMMS::Extended** represent the culmination of decades of research and development, beginning with the early avalanche modeling efforts at the **WSL Institute for Snow and Avalanche Research SLF** in the mid-1990s.

From Finite Elements to Finite Differences

Surprisingly, the first attempts to solve the **depth-averaged equations of motion** numerically made use of **finite element methods**, stabilized with **numerical viscosity** terms. This approach was employed in early one-dimensional models like **AVAL-1D** but proved inadequate for more complex flow behavior. These methods were soon replaced by more robust **upwinded finite difference schemes**, which offered better stability and accuracy, particularly in the presence of shocks and flow fronts.

Advancing to Two Dimensions

The growing demand for realistic, two-dimensional avalanche simulations led to the development of **AVAL-2D**, which implemented **first-order Total Variation Diminishing (TVD) schemes** on **triangular grids**. While triangular elements have geometric advantages for complex terrain, they suffered from significant **numerical diffusion**. This artificial smoothing made it challenging to define clear stopping criteria for avalanche motion, prompting a shift in modeling strategy.

Second-Order Schemes and the Birth of HLLE Solvers

To overcome the diffusion issues, a new generation of solvers was developed between 2005 and 2010. These employed **second-order schemes on rectangular grids**, based on the **Harten–Lax–van Leer–Einfeldt (HLLE)** Riemann solvers. These methods dramatically improved simulation accuracy and are still at the core of the numerical engine in **RAMMS::Extended** today.

Incorporating Terrain Effects: Curvature and Centripetal Acceleration

During this period, **centripetal acceleration terms** were introduced into the second-order HLLE framework to better capture flow dynamics in **narrow, winding channels**. With the emergence of **high-resolution digital elevation models (DEMs)**, it became possible to compute **terrain curvature** directly**. RAMMS::Extended** offers users the flexibility to **enable or disable curvature effects**, depending on the scenario. Interestingly, it has been observed that **avalanche snow cover tends to smooth out sharp terrain features**, suggesting that turning **curvature OFF** often yields more realistic results.

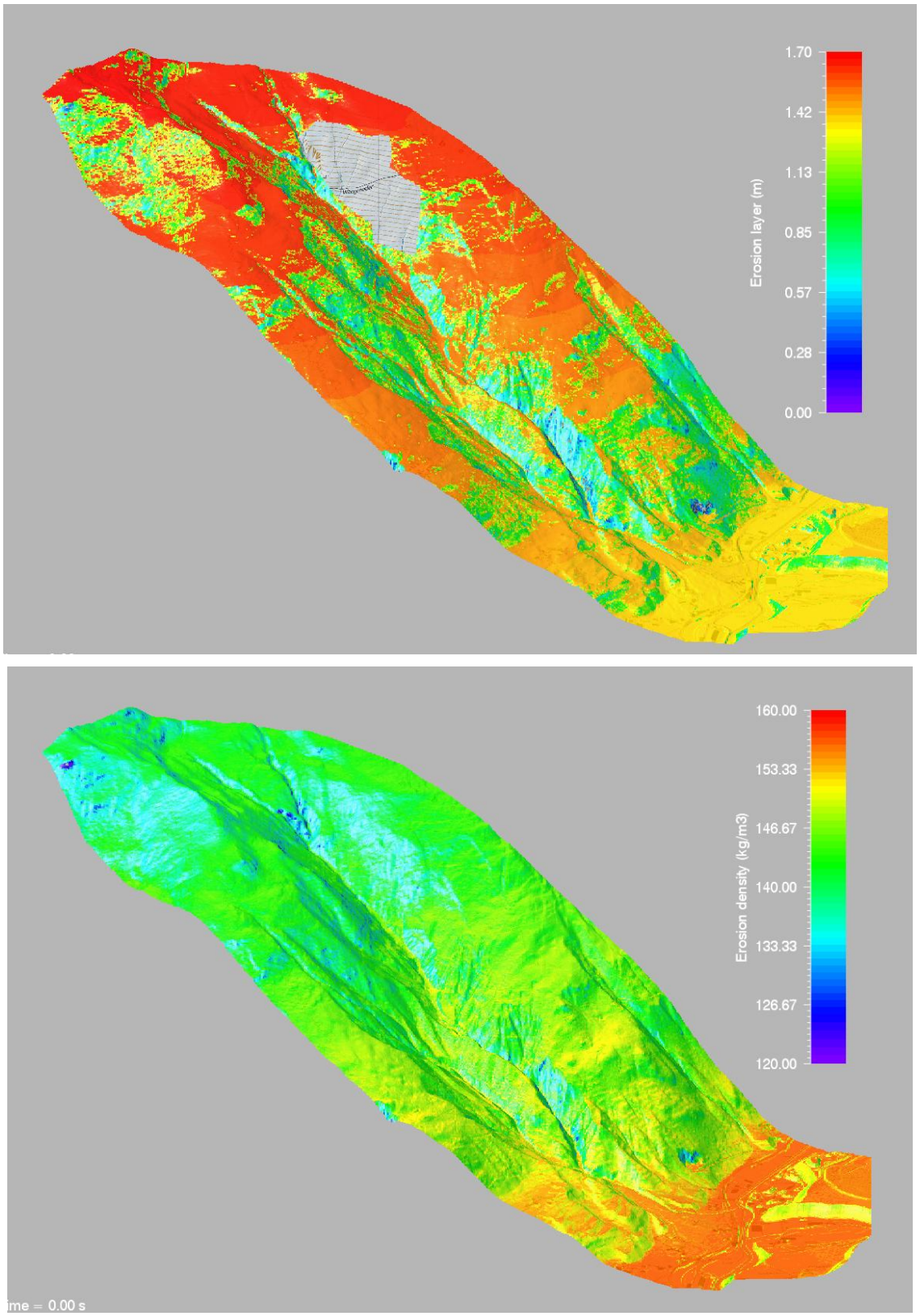


Figure 1.5: Spatial distribution of snowcover depth (top) and density (bottom).

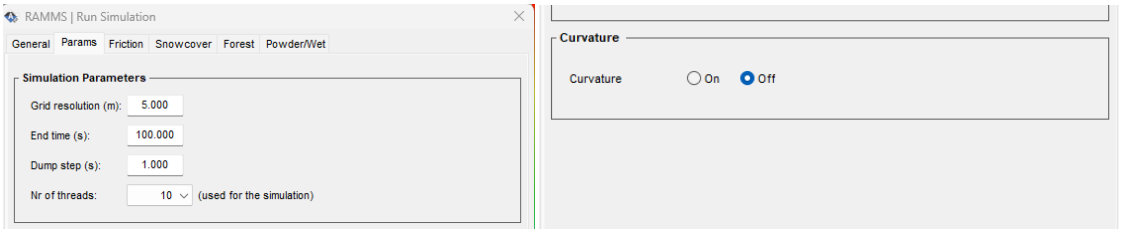


Figure 1.6:The Params Tab of the Run Simulation Menu contains input fields for the numerical solution: Grid resolution, no of threads and Curvature ON/OFF.

Parallelization and Grid Resolution

Modern versions of **RAMMS::Extended are fully parallelized**, and users can now specify the **number of computational threads**. This has significantly reduced simulation times and enabled the use of **high-resolution computational grids**. Most large-scale avalanche simulations are run on **5 m x 5 m grids**, which have proven to strike an optimal balance between computational efficiency and physical realism. Finer resolutions do not necessarily yield better results—in fact, for avalanches running over a deformable snow surface**, too-fine grids can misrepresent the smoothed nature of the terrain**. This same logic applies to curvature: just as snow smooths topographical roughness, it can dampen sharp curvature effects, reinforcing the case for selectively disabling curvature modeling.

Model Parameter	Component	Symbol	Parameter Range Constant Scenario
Voellmy Friction Parameters	Φ	$S_\Phi(\mu_0, \xi_0, N_0)$	Scenario-based return period recommendations
Production of RKE by Shearing	Φ	α_Φ	Production of fluctuation by shearing 0.05 wet $\leq \alpha_\Phi \leq$0.08 dry
Production of RKE by Entrainment	Φ	ε_Φ	Production of fluctuation energy during avalanche-snowcover interaction.
Decay of Stochastic Random Energy	Φ	β_Φ	Determines the longevity of the fluctuation energy in the avalanche core. Cold snow lower decay rates.
Dispersion	Φ	ζ_Φ	Fraction of random energy creating dispersion in avalanche core $\zeta_\Phi = 0.20$

Surface Melting Temperature	Φ	T_m	Melting temperature $T_m = -1$℃.
Activation Energy	Φ	$S_\Phi(R_0)$	Change of friction with increasing fluctuation energy $R_0 = 2$ kJ/m³
Cloud Drag	Π	$S_\Pi(\delta_\Pi)$	Determines the force of the avalanche air-blast $2.0 \leq \boldsymbol{\delta} \leq 4.0$
Cliff factor	Π	$\dot{M}_{\Phi\to\Pi}(\lambda_\Pi)$	Determines momentum exchange of plunging powder avalanches $2.0 \leq \lambda_\Pi \leq 5.0$
Decay of Turbulent Energy	Π	β_Π	Determines longevity of turbulence in powder cloud $0.5 \leq \beta_\Pi \leq 3.0$
Turbulent Air Entrainment	Π	$\dot{M}_{\Lambda\to\Pi}(\psi_\Pi)$	Determines air entrainment by turbulence in powder cloud $0.5 \leq \psi_\Pi \leq 4.0$
Splashing	Γ	γ_Γ	Determines how much erodible mass enters the splashing front $0.0 \leq \gamma_\Gamma \leq 0.5$
Particle restitution coefficient	Γ	r_Γ	Particles reach double the speed of the avalanche core, $r_\Gamma = 1$

Table 1.3: Modelling parameters for the avalanche core, cloud and splashing front.

Chapter 2 The RAMMS::Extended GUI

2.1 Overview

This chapter provides an overview of the key input and output features of the **RAMMS::Extended** graphical user interface (GUI). Built upon the foundation of the **RAMMS::Avalanche** GUI, it retains many of the intuitive, user-friendly input features that experienced users will recognize. As a result, those familiar with **RAMMS::Avalanche** should find navigating **RAMMS::Extended** seamless.

However, **RAMMS::Extended** introduces several advanced modeling capabilities, requiring additional input parameters such as **snow cover characteristics and mountain forest data**. To accommodate these complexities, the GUI has been enhanced with specialized input features designed to streamline data entry and facilitate efficient workflow.

This chapter is structured into two main sections:

- **Sections 2.2–2.6** focus on setting up a simulation, covering all essential input parameters.
- **Sections 2.7–2.14** explore the output features of the GUI, detailing the tools available for analyzing and visualizing simulation results.

By the end of this chapter, users will have a comprehensive understanding of how to configure and interpret simulations within **RAMMS::Extended**.

2.2 Installation and Licensing

Multilingual Installation

The **RAMMS::Extended installer** is designed to support multiple languages, facilitating a user-friendly installation experience for a diverse user base. Upon launching the installer, users can select their preferred language, which will guide them through the installation process in that language.

New Update Procedure

RAMMS::Extended incorporates a streamlined update mechanism to ensure users have access to the latest features and improvements.

- **Web Update**: The application can automatically check for available updates online (Help →Check for Updates). If an update is found, users can download and install it directly through the application interface.

Administrative Privileges Requirement

To install or update **RAMMS::Extended**, **administrative privileges are required**. This is necessary because the installation process involves modifying system configurations and installing essential components, such as the *IDL Visual Studio Merge Modules*. Without administrative rights, the installer cannot make these critical changes, and the installation or update will fail.

It's important to note that while **administrative privileges are required** for installation and updates, **they are not needed to run RAMMS::Extended** after installation is complete.

Licensing

RAMMS uses a personal-use licensing model. Licenses are time-limited and bound to a single machine via its unique host ID. To obtain a license, generate a license request file from the target machine and send it to RAMMS AG. You will receive a license key that, once entered, activates full functionality on that specific system.

Alternatively, a license can be installed on a Windows Server and accessed via Remote Desktop Connection (RDC). In this setup, the license can be used by different users, but only one user per module can access it at a time.

First-Time Startup and License Registration

When launching **RAMMS::Extended** for the first time, a splash screen for the IDL Virtual Machine will appear. Click anywhere on the image to proceed.

You will then see the **RAMMS | Licensing** window (Figure 2.1):

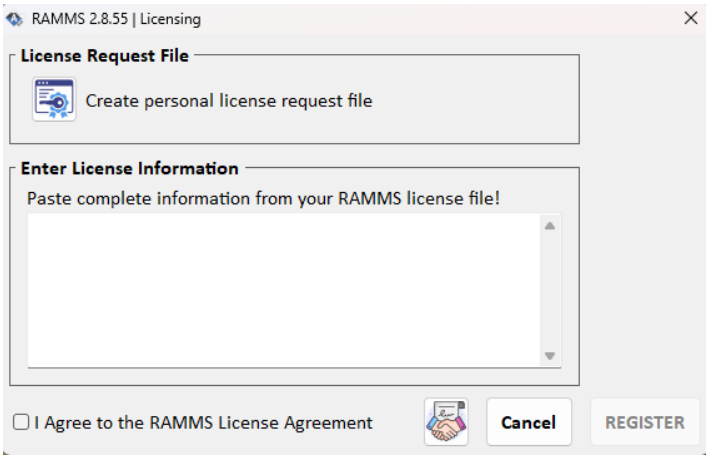
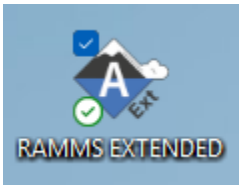


Figure 2.1: Enter the full contents of your license file in this dialog box.

Paste the full contents of your RAMMS license file into the designated field. Check the box to confirm **I Agree to the RAMMS License Agreement**, then click **REGISTER**. To review the license agreement before registering, click the handshake icon.

2.3 Starting the Graphical Interface

When the user clicks the **RAMMS::Extended** icon,



the **Graphical User Interface** (GUI) appears, providing a structured and user-friendly workspace. The interface is divided into five distinct regions, see Figure 2.2:

1. **Display, Animation and Preference Icons (Left Side):** A vertical list of icons for quick access to display features.

2. **Action Toolbar (Top Side):** A horizontal row of icons, including key functions for project management and data input/output handling.

3. **Tab Field (Right Side):** A dynamic space for working with multiple tabs, used for file management, user information and display options.

4. **Time Slider & Information Displays (Bottom Side):** Timeline control and key important user information.

5. **Input/Output Graphics Display:** In this window the input/output graphics is displayed in two-dimensional, or full three-dimensional mode.

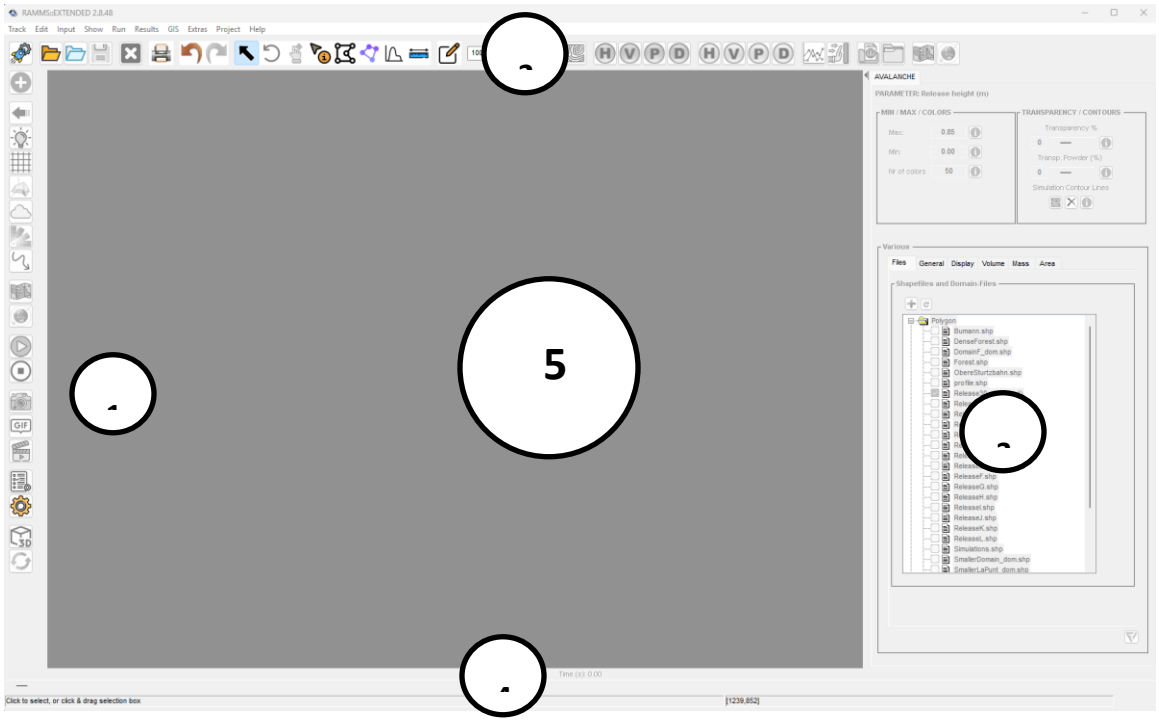


Figure 2.2: RAMMS::Extended GUI at startup. The interface is structured into five main regions: (1) a vertical panel on the left side featuring icons for display settings, animations, and preferences, (2) the action toolbar at the top for key functions, (3) the tab field on the right for managing files, adjusting display options, and accessing numerical data such as volume and mass balance, (4) the time slider and information displays along the bottom for timeline control and data visualization, and (5) the central display field for input and output graphics, currently empty.

At this point, users have **four primary options**, conveniently located in the upper action toolbar:

1. **Create a New Project** – Begin a fresh project from scratch.
2. **Open an Input File** – Load an existing input dataset for processing.
3. **Open an Output File** – Access previously generated results.
4. **Open a Recent File** – Quickly resume work from a recently accessed file

2.4 Project Creation / Opening Input and Output Files

	**Create Project**	Creates a project, selects a DEM
	**Open Input File**	Opens an existing input file
	**Open Output File**	Opens an existing output file

2.4.1 Project Creation

Every **RAMMS::Extended** workflow begins with the **creation of a project**. A project is inherently linked to a specific terrain model, which is defined by a **Digital Elevation Model (DEM)**. To establish a new project, users must consider three essential elements:

1. **Project Name**: A unique identifier for the project.

2. **Project Location**, or **Home Directory**: The directory on the user's computer where project files will be **stored**.

3. **Digital Elevation Model (DEM)**: A topographical dataset that forms the foundation of the simulation.

Initiating a New Project

To create a **new project**, click the "**Create Project**" icon, located on the upper toolbar. This icon is easily recognizable, featuring a rocket ship in front of a cogwheel. Upon selecting this option, the **Create Project menu** will appear (see Figure 2.3).

The user can also launch the **Create Project Wizard** via the pull-down menu by navigating to **Track → New → Project Wizard** (see Figure 2.3).

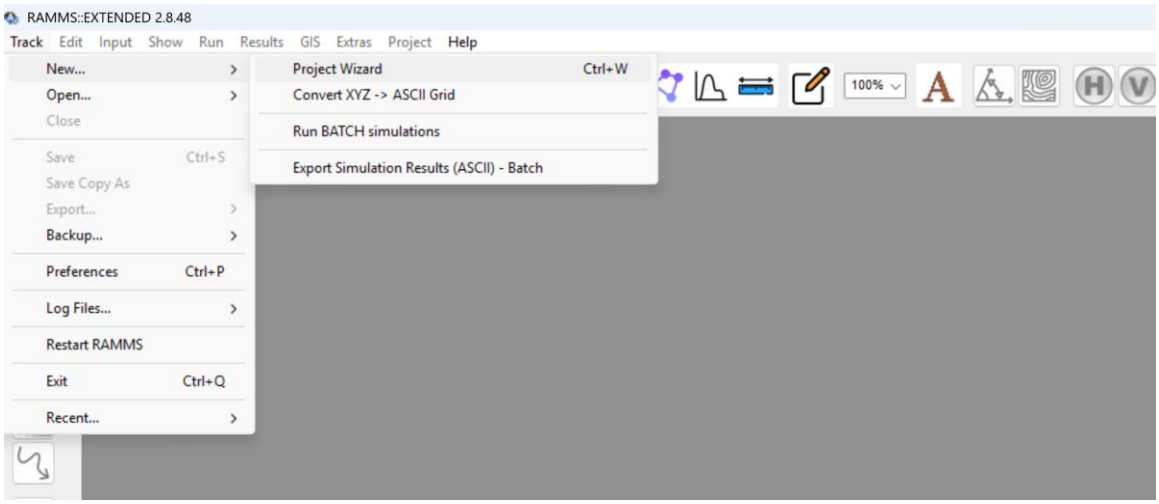


Figure 2.3: Activating the project wizard via pull-down menus.

2.4.2 Understanding the Create Project Wizard/Window

The **Create Project** interface is divided into three distinct sections (**Figure** 2.4), each serving a crucial role in the project setup process:

1. **Project Name & Home Directory (Upper Section)**
    - Users must define a project name and select a home directory where all simulation files will be stored.
2. **Project Description (Middle Section)**
    - This section allows users to input descriptive text about the project.
    - While often overlooked, this feature can be highly valuable—especially when revisiting simulations years later. Documenting important details ensures continuity and understanding for future analysis.
3. **DEM File Selection (Lower Section)**
    - Here, users import the DEM file, which defines the terrain for the simulation.
    - Once selected, **RAMMS::Extended** automatically analyzes and reports the DEM resolution, typically 2m or 5m. The **Grid Resolution** tab in the **DEM Input** section allows users to adjust the resolution of a DEM to better suit their simulation needs. High-resolution DEMs **are often specified, but their level of detail may be excessive for avalanche dynamics simulations**, leading to unnecessary computational complexity.

To modify the **DEM resolution**, simply click on **"Grid Resolution"**, enter the desired resolution, and apply the change. For instance, if the original DEM has a resolution of **0.25 meters**, it can be resampled to a coarser resolution, such as **2 meters**, using **bilinear interpolation**. Similarly, if the DEM resolution is too coarse (e.g., **30 meters**), it can be refined to a finer resolution (e.g., **2 meters**) using the same functionality. This flexibility ensures that the DEM is appropriately scaled for efficient and accurate simulations.

By carefully setting up a project with a clear name, an appropriate directory, and an accurate DEM, users establish a solid foundation for successful simulations in **RAMMS::Extended**.

2.4.3 DEM Files *.asc and *.tif

RAMMS::Extended supports **two different types of file formats for DEM files**: *.asc and *.tif. The **ASC format**, an ASCII Grid format developed by Esri, is a plain text file that stores elevation data in a structured grid with human-readable values, making it easy to edit but often larger in size. It includes metadata such as cell size, coordinates, and nodata values. On the other hand, the **TIFF format**, specifically GeoTIFF (*.tif), is a raster format that supports **georeferencing** and can store elevation data in a more compact, binary form. GeoTIFFs are widely used due to their efficiency, ability to store multiple bands, and compatibility with GIS software. While ASC files are simpler and useful for certain applications, GeoTIFFs are generally preferred for large-scale DEM storage and processing due to their superior compression, metadata handling, and performance.

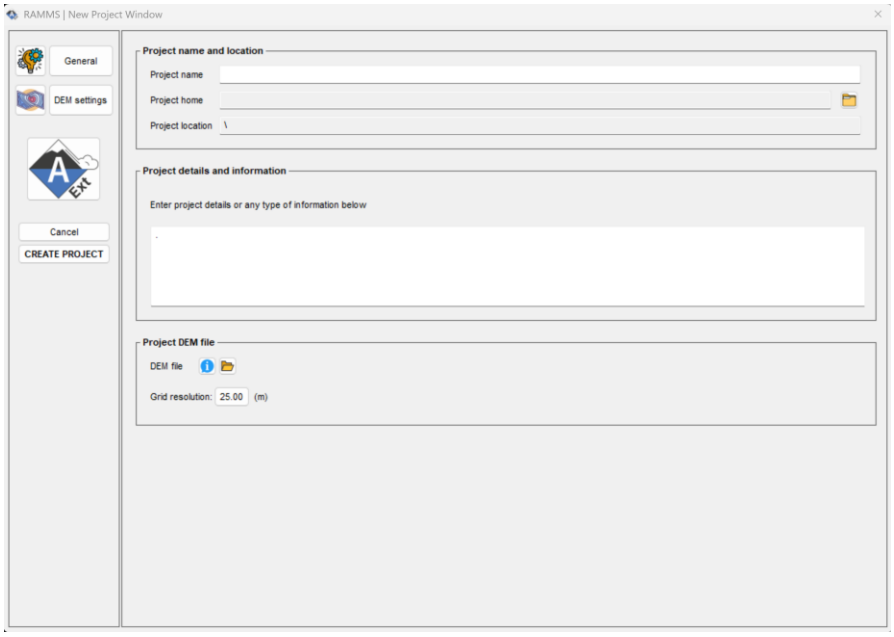


Figure 2.4: The RAMMS::Extended Create Project Menu. The menu is divided into three sections. In the upper section project name and home directory are defined. In the lower section the DEM is specified by the user. In the middle section project information can be inserted.

The **DEM Settings** menu is accessed from the **Create Project** menu (Figure 2.5) and serves two main purposes: **providing key spatial information**—such as the maximum **North, South, East, and West** coordinates—and **allowing users to clip the DEM** if needed.

Clipping the DEM can be done in one or two ways:

1. **Manual Coordinate Entry** – Users can manually define the clipping boundaries by entering exact **N, S, E, and W** coordinates. This is done by activating the corresponding icons in the **DEM Settings** menu, after which the GUI prompts the user to input the desired values.

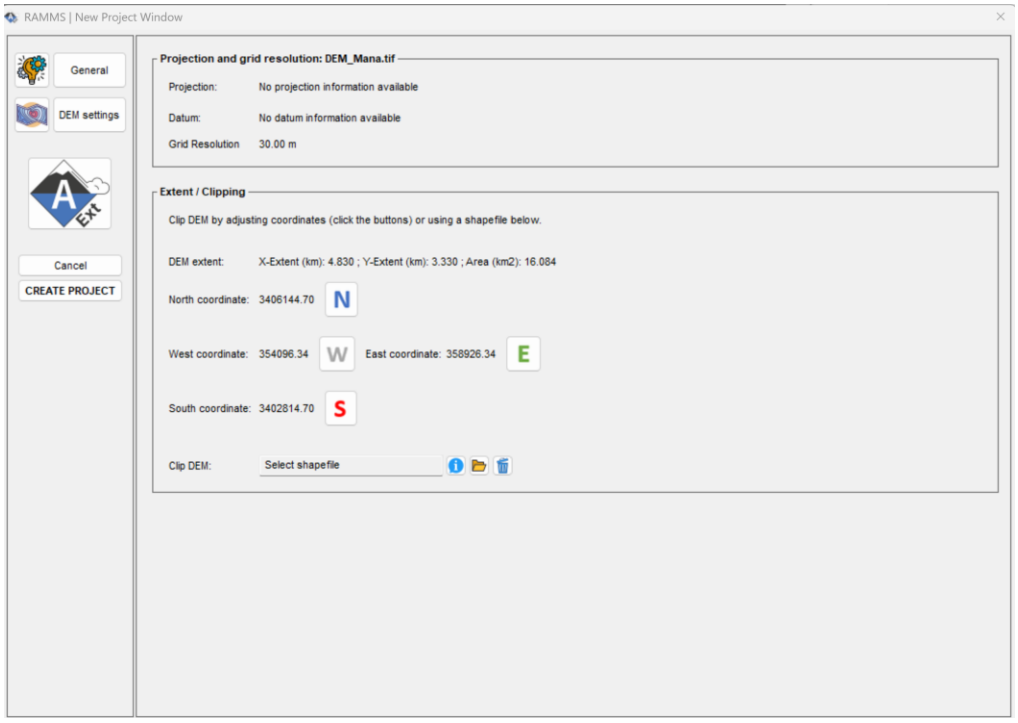


Figure 2.5: The DEM Settings menu. With this menu, the DEM can be reduced by clipping.

2. **Shapefile-Based Clipping** – Alternatively, users can clip the DEM using a **domain shapefile**. This method is often preferred, particularly when refining the computational domain from existing simulations and when a suitable domain file from a previous calculation is available. (Because the project is being created, often, a domain file does not exist.)

By offering these two clipping options, the **DEM Settings** menu provides flexibility in tailoring the DEM to the specific needs of the project.

2.4.4 Opening an Existing Input File

To access one of the input files of an existing project, the user must click on the **Open Input File** icon located on the **upper toolbar** of the **Graphical User Interface (GUI)**. This

icon is positioned between the **Create Project** and **Open Output File** menu options. Navigate to the folder of the project and select the input file of your choice.

Upon opening, the input file will load with the **Digital Elevation Model** (DEM), along with the most recently defined **map** or **orthophoto** overlay. The release shapefile remains unchanged from its last saved state, ensuring that all previously configured parameters and settings are preserved.

2.4.5 Opening an Existing Output File

To open one of the scenario **output files** of an existing project, the user clicks on the **Open Output File** icon, located on the **upper toolbar** of the **Graphical User Interface (GUI)**. This icon is positioned **next to the Open Input File icon**, making it easily accessible. Navigate to the folder of the project and select the output file of your choice.

When selected, the corresponding output file loads, displaying the **simulation results**. This includes **computed flow paths, deposition areas, velocity distributions, or other relevant output**, depending on the simulation settings. The loaded output file retains all previously saved results, allowing users to analyze, compare, or refine their simulations as needed.

2.5 Interacting with the DEM: 2D/3D, Zoom and Rotate

	**2D/3D Toggle**	Toggles between 2D and 3D visualization
	**Refresh Simulation Window**	Refreshes simulation window
	**Zoom 2D/3D View**	Shifts and zoom 2D/3D Visualizations
	**Rotate 3D View**	Opens an existing output file

RAMMS::Extended provides a range of intuitive tools for navigating and visualizing simulation results in both two-dimensional (2D) and three-dimensional (3D) environments. These tools are organized within the graphical user interface (GUI) to support efficient workflows and quick access during project analysis.

The **2D/3D toggle switch** and the **refresh icon** are positioned at the **bottom of the vertical action toolbar**, located along the **left-hand side** of the GUI. These controls allow users to easily switch between visualization modes and update the view as needed.

Additional navigation tools—such as **panning**, **zooming**, and **rotation**—are found along the **horizontal toolbar** at the **top of the GUI**. These tools are essential for exploring the simulation domain, adjusting the view to focus on specific terrain features, and inspecting the overall quality of the digital elevation model (DEM) and simulation output.

Together, these interface elements provide a flexible and user-friendly way to interact with both 2D and 3D data within **RAMMS::Extended**.

Tool	Description
2D/3D Toggle	Located on the left-hand action bar, this button **switches between 2D and 3D visualization modes**. The default view when a project is created or opened is **2D**. 2D views are particularly useful for **comparing simulation results over georeferenced maps and orthophotos**. The **3D view**, on the other hand, allows users to examine terrain features and assess the quality of the underlying Digital Elevation Model (DEM). The icon toggles automatically depending on the current mode, making it easy for users to identify and switch between views.
Refresh	Resets/refreshes simulation view.
Zoom 2D/3D View	This tool is activated by default upon opening a project. **It enables users to pan and zoom within the 2D or 3D domain**. To shift the DEM, users simply click and hold the left mouse button on the terrain, then drag to move in any direction. Zooming in and out is done via the mouse scroll wheel (roll bar).
Rotate 3D View	This function is only available when in **3D mode**. Once activated, three interactive on-screen rotation arcs appear representing the three rotation directions. Users can rotate the terrain model using the rotation arcs (see Figure 2.6). Clicking and holding the left mouse button on a specific arc allows for rotation around that axis. If no specific arc is selected, users can still rotate the entire domain freely by clicking and dragging anywhere on the terrain.

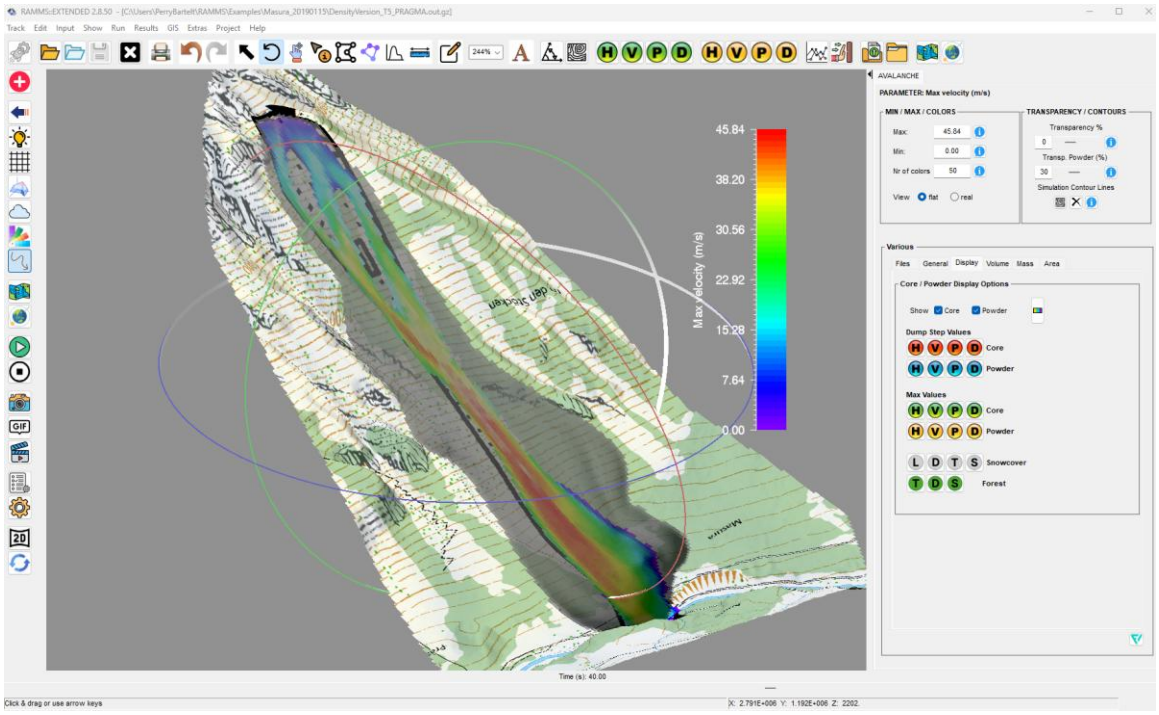


Figure 2.6: Rotation arcs appear when the rotate icon is selected. These can be activated by dragging the mouse in the direction of an arc.

2.6 Drawing Shapefiles / Inputting Shapefiles

	**Draw Shapefile**	Draw a shapefile to define a release area, a forest or a no-flux boundary
	**Add Shapefile**	Add a shapefile to the visualization

Shapefiles play a crucial role in defining key terrain features within a **RAMMS::Extended** project. They are used to delineate **avalanche release areas**, **forested terrain**, **no-flux boundary conditions**, and **regions with localized friction values**.

To create a shapefile, the user clicks on the **Shapefile Icon**, which is represented by a **polygon with five vertices**. Upon selection, the **mouse pointer automatically switches to shapefile mode**, allowing the user to outline a polygon.

Drawing a Shapefile Polygon

1. The **first left-click** is used to establish the **initial vertex** of the polygon. At this stage, no lines are visible, as the polygon's shape has not yet begun to form. It is only with the **second left-click** that a connecting line appears between the **first and second vertices**, signaling the start of the polygon's outline.
2. Since the **first vertex** serves as the foundation for the entire polygon, users should take special care to **note its location**. Keeping track of this initial point ensures a well-defined shape and a smooth closure of the polygon when completing the drawing process.
3. **Orientation & Placement**: The polygon should be drawn in a **clockwise direction** using the **left-click button** to place each vertex.
4. **Defining Vertices**: Users can define as many vertices as needed, but typically, fewer than **20 vertices** are sufficient for accurate modeling.
5. **Closing the Polygon**: To complete the shape, the **right-click button** is used to connect the first and last vertices. Note that this final right-click does **not** add an extra vertex; instead, it simply **closes the polygon and finalizes the calculation domain**.

By using shapefiles effectively, users can fine-tune model parameters and improve simulation accuracy, ensuring that the terrain characteristics are well-represented in the analysis.

2.7 Defining/Editing Release Areas

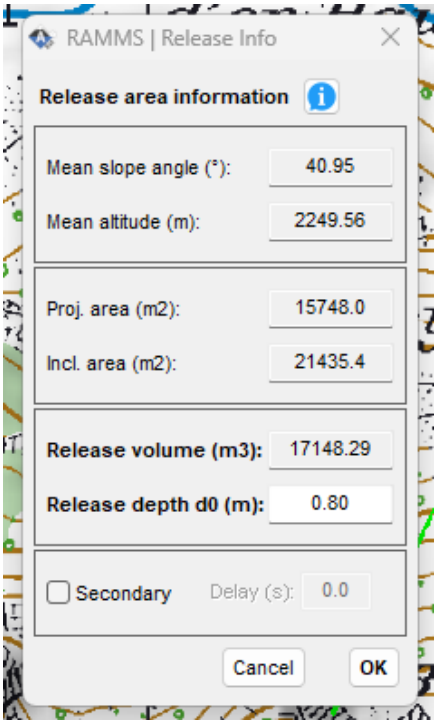
	**Defining/Editing Release Areas**	Sets the mouse into select mode for a release shapefile

To initialize a **RAMMS::Extended** simulation, users must define at least one **release area** by providing a **release shapefile** and assigning it a **release depth** (d_0). This process is carried out using the **View/Edit Release Area** tool, located on the **upper action bar** of the GUI.

Step-by-Step Process

1. **Select the Release Area Tool**
   - Click the **View/Edit Release Area** icon in the upper action bar.
2. **Choose a Release Shapefile**
   - Using the **left mouse button**, select a predefined **release shapefile**.
3. **Input the Release Depth (d_0)**
   - A menu will appear, prompting you to enter a **release depth** (d_0) in meters.
   - By default, when a shapefile is selected for the first time, the **release depth is set to 0.0 m**.

- As you enter a depth value, the **release volume field updates dynamically** based on the equation: $V = d_0 \cdot A$ where **A** is the area enclosed by the shapefile polygon.



4. **Review Shapefile Information**
   - The menu provides key details about the selected release area, including:
     - **Mean slope angle**
     - **Mean altitude**
     - **Total project area**
     - **Inclined (real) surface area**
5. **Define Primary and Secondary Releases**
   - **Primary releases** occur at $t = 0$ when the simulation starts.
   - **Secondary releases** can be specified by checking the **"Secondary"** box and setting a **delay time** (in seconds) relative to the primary release.
   - At least **one primary release** is required. If only a secondary release is defined, the simulation will immediately terminate because no mass is flowing at $t = 0$.
6. **Finalize and Confirm**
   - Click **OK** to confirm the release depth and settings.
   - Initially, selected shapefiles are displayed as **dashed red outlines**.
   - Once the **release depth is assigned**, the polygon will be **filled with a color** representing the **release depth gradient**.

Important Considerations

- The **friction parameters** used in the simulation depend on the **release volume** (V). For details on selecting appropriate values, refer to **Chapter 4**.
- Ensuring the correct definition of **primary and secondary releases** is critical to avoid simulation errors.

2.8 Slope Angles and Contour Lines

	**Show Slope Angles**	Draws slope angles
	**Add/Remove Contour Lines**	Adds/removes contour lines to visualization

RAMMS::Extended includes tools for visualizing terrain features that are essential for avalanche modeling and terrain assessment. Two such tools—**Show/Remove Slope Angles** and **Add/Remove Contour Lines**—are in the **upper toolbar**.

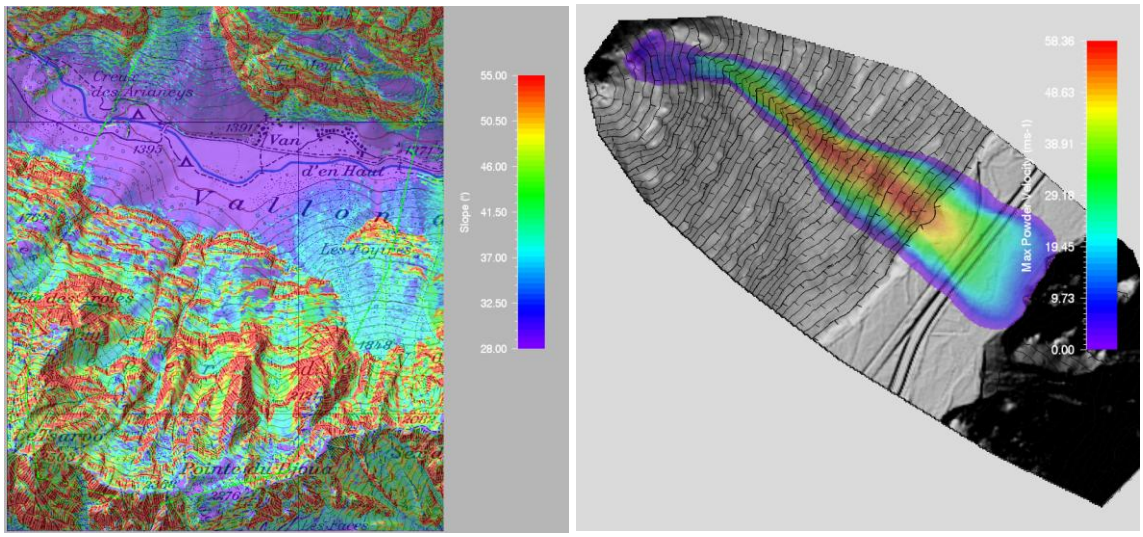


Figure 2.7: Left panel: Visualization of slope angles. Right panel: Visualization of contour lines using a hillshade image. Slope angle visualization is often used to delimit avalanche starting zones. The contour line feature is used when no maps are available, or to control the quality of the DEM.

Slope Angle Visualization

Clicking the **Show/Remove Slope Angles** icon activates a visualization of slope angles ranging from **28° to 55°** within the simulation domain (Figure 2.7). These slope values are

typical for **avalanche release zones**, making this tool particularly useful when delineating potential starting zones. To enhance visibility and accurately interpret these zones, users are encouraged to adjust the **transparency percentage slider** on the **right-hand side** of the GUI.

In addition, the **minimum and maximum slope angle thresholds** can be redefined to focus on specific slope ranges, helping users to better refine the display and isolate areas of interest. Clicking the slope angle icon again will remove the visualization from the view.

Contour Line Visualization

The **Add/Remove Contour Lines** icon allows users to overlay topographic contour lines onto the terrain (Figure 2.7). This feature is especially useful when working with **hillshade imagery**, particularly in cases where no georeferenced map is available. Contour lines help highlight **steep, rugged terrain** and are commonly used to assess surface features and elevation gradients.

A key application of the contour overlay is to **compare DEM-generated contours** with those found on **georeferenced topographic maps**. By visually examining the differences between these contour sets, users can quickly assess the **quality, resolution, and potential inaccuracies** in the digital elevation model. This comparison provides important insights into the reliability of the terrain data being used for simulation. Contour lines are removed by re-clicking the contour line icon.

2.9 Calculation Domain: Left click, Right click

	**Draw Calculation Domain**	Draw a shapefile around the calculation domain to speed-up calculations

The **Calculation Domain** icon, located on the upper menu bar, allows users to create a shapefile around the (supposed) avalanche track. This feature is designed to define a smaller calculation domain, significantly improving computation speed.

Creating a Calculation Domain:

When the icon is selected (Figure 2.8), the mouse pointer immediately switches to shapefile mode, enabling the user to outline a polygon around the avalanche track. The polygon should be drawn in a clockwise direction, using the **left-click button** to place each vertex. To complete the polygon, the **right-click button** is used to connect the first and last vertices. Note that the final right-click is not included as a vertex—it simply closes the calculation domain polygon.

Functionality and Behavior:

- Cells outside the polygon are excluded from calculations, effectively limiting the computational domain.

- The polygon acts as an open boundary, meaning any mass—whether from the avalanche core or powder cloud—exiting the domain is no longer considered in the simulation.

- Users should not worry if part of the avalanche extends beyond the domain, as mass flowing out is naturally removed from the calculation. The user can define a larger polygon in the next simulation if necessary.

- This feature is identical to the one available in **RAMMS::Avalanche** and serves primarily to reduce computation times.

By strategically defining a calculation domain, users can optimize performance.

2.10 Inserting Maps, Orthophotos and Hillshades

	**Insert Map**	Allows the user to place a georeferenced map over the DEM
	**Insert Orthophoto**	Allows the user to place a georeferenced orthophoto over the DEM

One of the most effective ways to visualize both input and output data in **RAMMS::Extended** is by overlaying a **georeferenced map or orthophoto** onto the calculation domain. These tools are invaluable for precisely **positioning release zones** and identifying **endangered structures** in runout areas.

TIFF (*.tif) Files: Tagged Image File Format (TIFF) is a widely used raster image format known for its high resolution and ability to store geospatial metadata.
World Files (*.tfw): A TFW file is a plain-text file that contains transformation parameters, allowing GIS software and **RAMMS::Extended** to properly position a raster image within a coordinate system.

Functionality and Workflow

Both the map and orthophoto tools function in the same way:

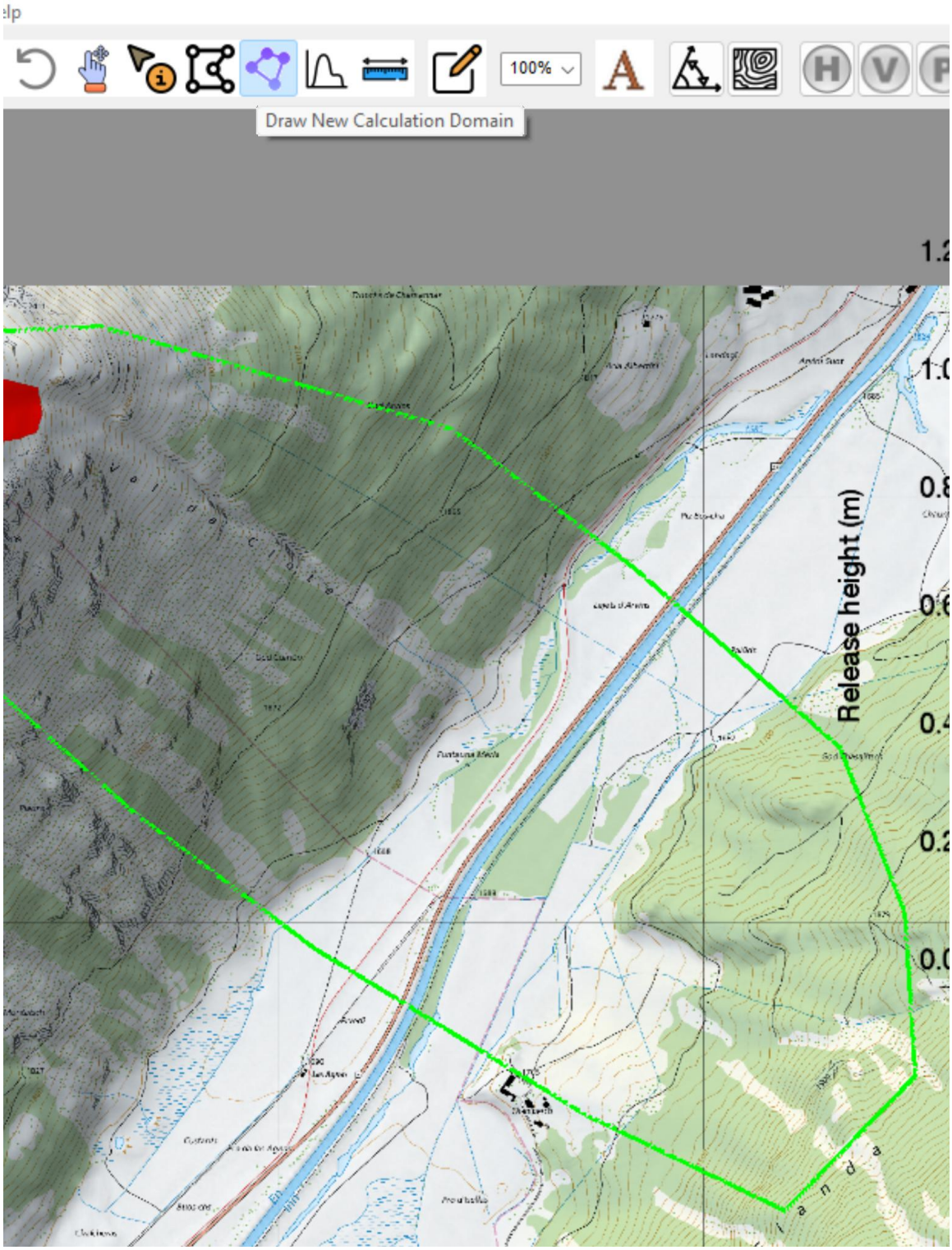
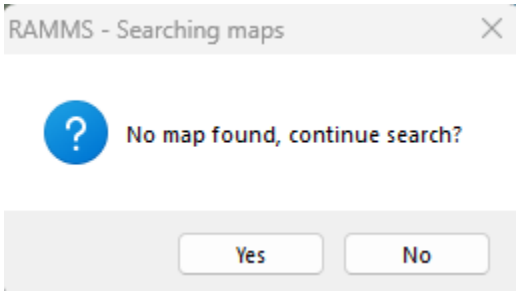


Figure 2.8: By clicking on the draw calculation domain icon, it is possible to define a smaller calculation domain, here shown in green. Selection of a smaller calculation domain improves computation speed. The vertices of the calculation domain are defined by a left click. To close the domain polygon, right click.

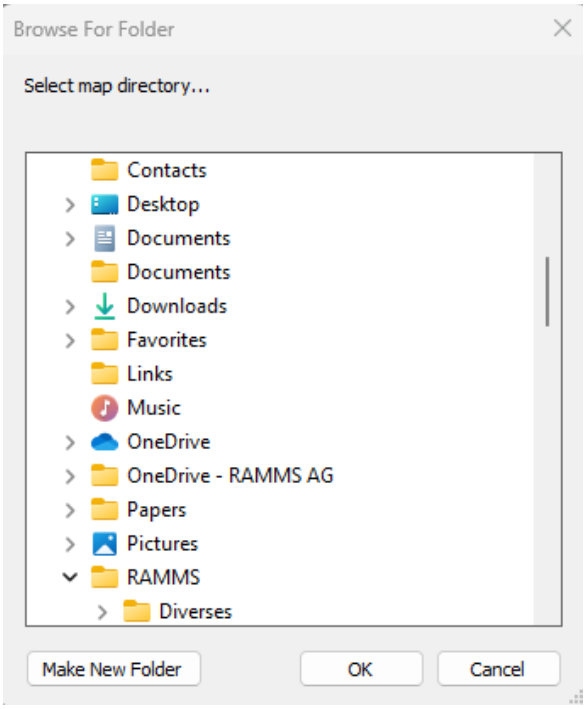
1. **File Search & Selection**

   - When the corresponding icons on the **top menu bar** are selected, the program **automatically searches** for a georeferenced map or orthophoto in the simulation subfile directory.

   - If no suitable file is found, the user is prompted to **search in another directory**. Selecting "Yes" opens a file selection dialog, allowing the user to navigate to the correct location.



   - Typically, maps and orthophotos are stored in a **separate subfile directory**, requiring manual selection during the initial setup.



2. **File Format Requirements**

   - The georeferenced images must be in **TIFF format** (*.tif).

- A corresponding **World File** (*.tfw) is required to provide spatial reference data, ensuring proper alignment within the calculation domain.

3. **Switching Between Map and Orthophoto**

   - Both a map and an orthophoto can be used simultaneously.
   - Once loaded, the left-side menu icons allow users to quickly toggle between the map and orthophoto for better visualization.
   - The top menu bar icons are used only for file selection, while the left-side menu icons facilitate switching between the two overlays.

Hillshade Images

In many cases, users may find themselves working in regions where **no georeferenced maps or orthophotos are available**. This lack of background data can make it difficult—if not impossible—to properly evaluate simulation results.

To address this challenge, **RAMMS::Extended** provides an intuitive solution: the ability to generate **hillshade images** directly within the graphical user interface (Figure 2.9Figure 2.9). **These images offer a shaded relief representation of the terrain, enhancing visual interpretation and providing essential spatial context.**

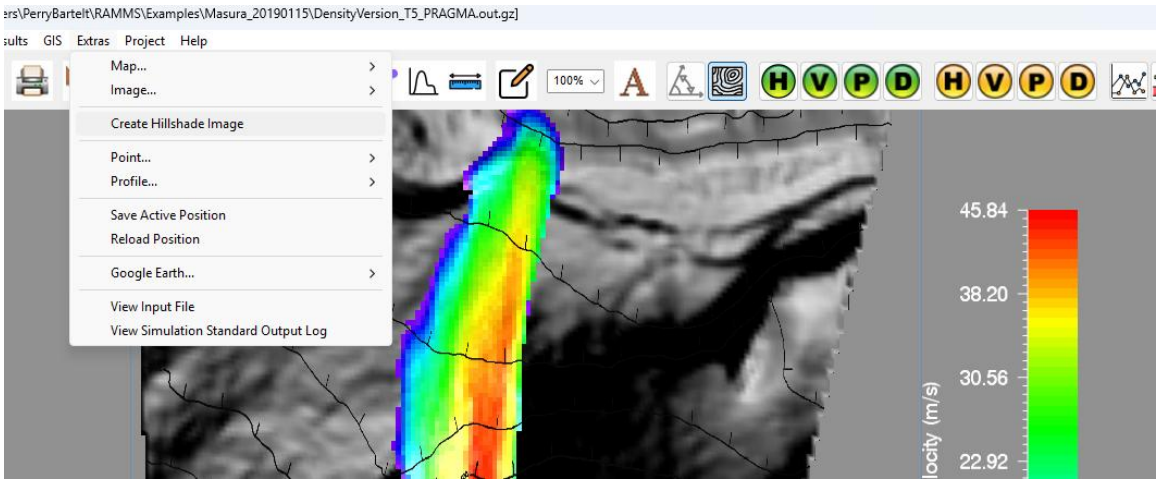


Figure 2.9: To create a Hillshade Image go to the Extras→Create Hillshade Image pull-down menu. Hillshade images are often used when a georeferenced map or image is not available.

To create a **hillshade image**:

1. Navigate to the **Extras** menu.

2. Select **Create Hillshade Image**.

3. **Specify a name and save location for the image**.

Once created, the hillshade image acts as a stand-in for a georeferenced map or orthophoto and can be used as a background layer during result visualization. The image can be loaded into the simulation the same way as a map or orthophoto. The hillshade, with the name given by the user, will appear in the table of maps or photos. If no maps/orthophotos are available, the hillshade image will be automatically loaded.

For even greater clarity, users are encouraged to activate the **contour line overlay** (using the contour line icon). This places elevation contours on top of the hillshade, further enhancing terrain readability and aiding interpretation, especially valuable when no other reference data is available.

2.11 Run Simulation and the Input Tab

2.11.1 General Tab

The first tab located on the left side of the **Run Simulation Window** is the **General Tab** (Figure 2.10). This tab serves three key functions:

1. **Scenario Name Input**
   The **Scenario Name** field allows users to specify a unique name for their simulation run. Both **input** and **output** files will use this name. As with all RAMMS modules, the scenario name should not contain spaces (even in directories) or **special characters unique to specific languages**. A well-structured naming convention helps distinguish between multiple simulation runs using different parameters.

2. **Additional Information & Digital Elevation Model (DEM)**
   In the middle section of the **General Tab**, there is an **Additional Information Field**, where users can input supplementary details relevant to the simulation. This section also displays crucial information about the **calculation domain** and the **digital elevation model (DEM)** being used.

3. **DEM Update**
   A standout feature of **RAMMS::Extended** is the ability to update the **digital elevation model**. This capability is particularly useful when the simulated avalanche deposits obstruct a flow channel, leading to channel outbreaks on the runout fan. In such cases, users **can integrate the avalanche deposits into the DEM, creating an updated terrain model**. (Creating an updated DEM is performed using the **Results** pull down menu and selecting **Miscellaneous** and then **Add Topographic Changes (Erosion/Deposition)** to DEM, see Results section and Figure 2.11Figure 2.11.

This new DEM is saved under a different file name (see Results section for details) and can be **loaded** by the **General Tab** for subsequent simulations, ensuring more realistic flow modeling.

4. **Defining Stopping Criteria**
   The third key function of the **General Tab** is to establish **stopping criteria** for the simulation. As RAMMS performs calculations, it continuously monitors the **momentary mass flux** (moving mass) and compares it to the **maximum mass flux** recorded during the simulation. This ratio is expressed as a percentage and is checked against the **predefined stopping criterion** defined in the **General Tab**.

   - **Default Stopping Threshold:** The standard threshold is 5%, meaning the simulation will stop when the moving mass flux falls below 5% of the maximum recorded flux. The simulation will then terminate with a "**LOW MASS FLUX**" message—indicating a natural conclusion rather than an abrupt termination as the simulation end time is reached.

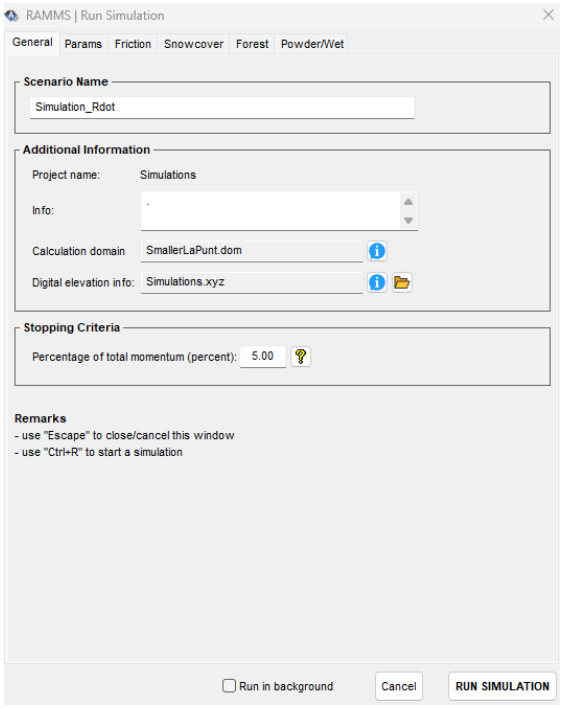


Figure 2.10: The General Tab. Used to input simulation name, additional information and stopping criteria.

   - **Adjusting for Different Avalanche Sizes:** For larger avalanches (>10,000 m^3), the 5% threshold is typically appropriate. However, smaller avalanches, containing less mass, may require a more lenient stopping criterion:

- **Moderate-sized avalanches (~2,000–10,000 m³)**: A 10% threshold is often preferable.

- **Tiny avalanches (<2,000 m³)**: A 20% threshold may be necessary to prevent avalanches from continuously moving and creeping forward with a small mass flux.

By fine-tuning the stopping criteria, users can ensure that RAMMS simulations run to completion, accurately reflecting the physical behavior of avalanches across different scales.

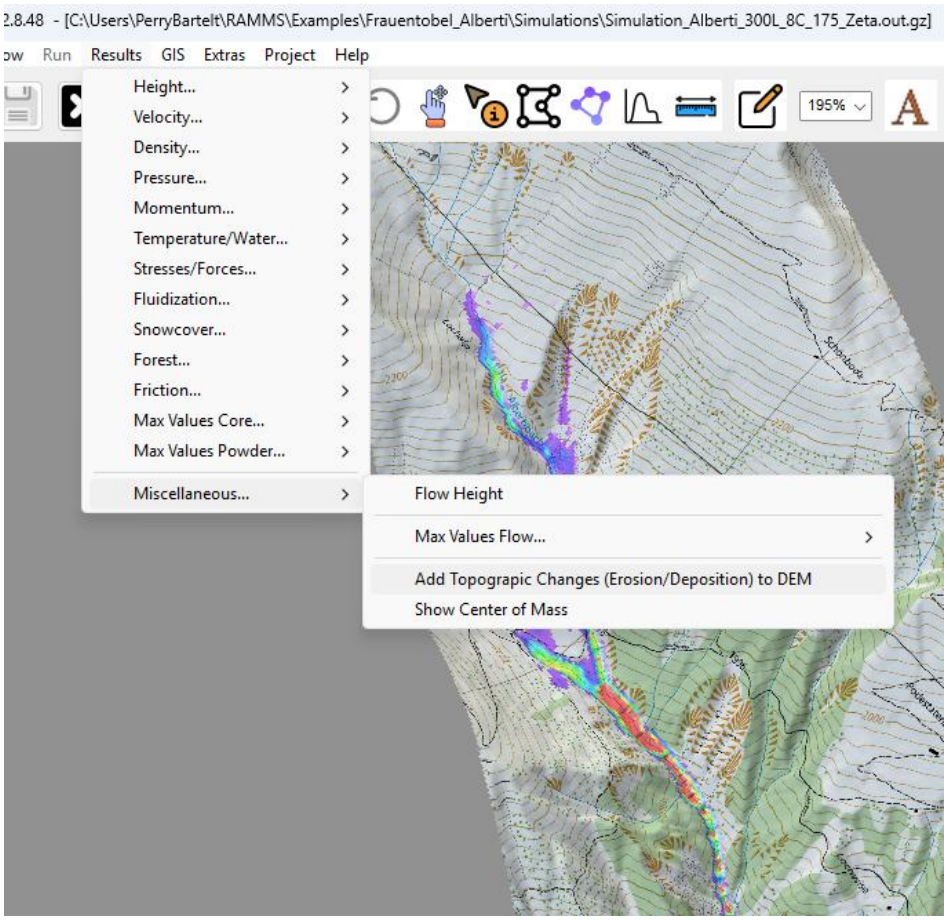


Figure 2.11: The DEM can be updated with avalanche deposits by activating the Results pull-down menu and selecting Miscellaneous and Add Topographic Changes (Erosion/Deposition) to DEM. Core heights should be selected and displayed as shown in the figure. In this case avalanche deposits of over 5m are blocking the flow channel. The DEM is given a name, which can be selected in the General Tab.

2.11.2 Params Tab

The **Params Tab** (Figure 2.12) in **RAMMS::Extended** serves three essential functions:

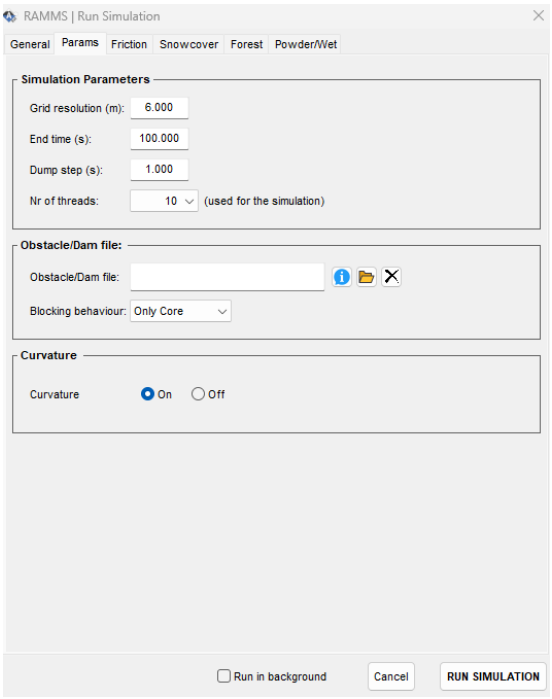


Figure 2.12: The Params Tab is used to input simulation parameters such as grid resolution, dump step, curvature and no-flux boundary conditions.

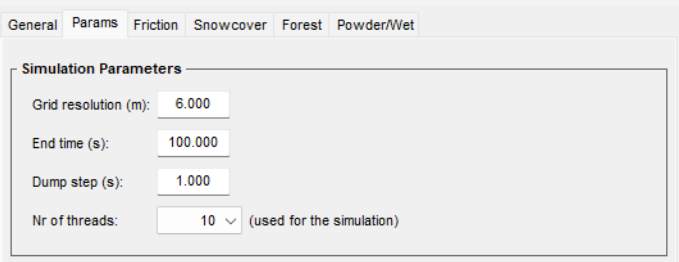
1. **Simulation Control Parameters**: It allows users to input key parameters such as **grid resolution**, **maximum simulation time**, **dump step interval**, and the number of **parallelization threads**.

2. **Boundary Conditions**: It is used to define **no-flux boundary conditions**, which are helpful in modelling dams, or steep mountain terrain and counterslopes.

3. **Curvature Effects**: Users can specify whether **curvature effects** (centripetal accelerations) should be incorporated into numerical calculations.

2.11.2.1Simulation Control Parameters

The upper section of the **Params Tab** contains fields for four **simulation control parameters**:



Grid Resolution: The **default grid resolution** for **RAMMS::Extended** calculations is **5m × 5m**. This resolution is optimal for most simulations, as it balances the representation of **terrain features** with the **damping effects of the snow cover**.

- **Standard Avalanches**: For avalanches with **release volumes greater than 5000m³**, the **default resolution (5m × 5m) is recommended**. Small (S), Medium (M), and Large (L) avalanches generally respond only to medium- and large-scale terrain features, making this resolution sufficient.

- **Tiny Avalanches (T)**: Avalanches with low mass and shallow flow heights may require finer terrain resolution. In such cases, a **2m × 2m resolution is recommended**. Tiny avalanches running on terrain without snowcover will certainly require higher terrain resolution.

- **Very Large Avalanches**: For high-volume avalanches, a 10m × 10m resolution may be appropriate. In fact, it is advisable to perform initial simulations at this resolution to define the calculation domain before refining further.

End Time: The **End Time** input defines the **maximum simulation duration in seconds**. For example, if **End Time** is set to 300s (default), the simulation will stop after 300 seconds, regardless of whether mass flux is still ongoing. The 300s represents avalanche time, no real calculation time (the amount of time the user waits for the calculation to end). Most avalanches reach the deposition zone **within 120 to 180 seconds**. A first approximation of 200s is typically recommended.

Dump Step Interval: This input value determines how frequently simulation results are saved.

- A **standard recommendation** is 5s intervals, which provides sufficient **temporal resolution while keeping output file sizes manageable**.

- For creating **videos** or detailed **time plots**, a 1s dump step interval may be preferable.

- If the **dump step interval** is set close to the **End Time**, only the initial and final states will be available. This cuts down on output size dramatically.

- **Caution**: Selecting an excessively high resolution (e.g., t < 0.5s) can lead to **unnecessarily large output files**.

Number of Parallelization Threads: The number of **parallelization threads** can be adjusted based on the user's available computational resources.

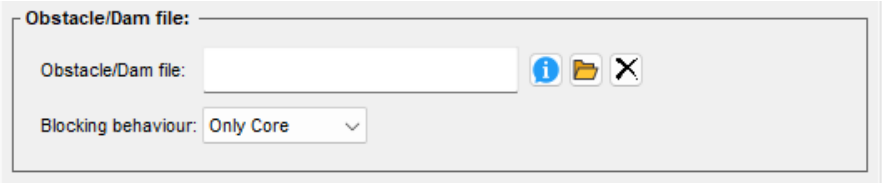
- **Selecting a large number** (32) can lead to substantial speed-ups, if the threads are available.

- If the user **wants to continue working** while the simulation is running, it is advisable to allocate **fewer threads than the total available on the system**.

- Selecting **too many threads** may fully utilize the CPU, slowing down other tasks on the computer.

By carefully configuring these parameters, users can optimize both performance and accuracy in RAMMS::Extended simulations.

2.11.2.2 No-flux Boundary Conditions – Obstacles / Dam File



In **RAMMS::Extended**, **no-flux boundary conditions** are defined in the Obstacles/Dams input mask within the **Params Tab**. **These boundaries prevent the avalanche core and cloud from entering designated areas and are specified using shapefiles** (Figure 2.13). These shapefiles can represent dams, large obstacles like buildings, or extreme terrain features such as counter-slopes or cliff faces.

A key feature of **RAMMS::Extended** is the ability to apply **no-flux conditions selectively**—either to the avalanche **core alone** or **to both the core and the powder cloud**. By default, the no-flux boundary applies only to the core, allowing the powder cloud to flow over obstacles (Figure 2.14). This is particularly useful for modeling **powder pressures behind dams** that stop the core flow.

Types of No-Flux Boundaries: Braking vs. Deflecting Dams

No-flux boundaries can serve two main purposes:

1. **Braking Dams**: Used to stop avalanche mass directly.

2. **Deflecting Dams**: Designed to smoothly redirect the avalanche flow.

A significant issue with modeling braking dams is the generation of **excessively high core flow heights**. When an avalanche core impacts a stopping dam, the snow undergoes extreme compaction, leading to densities higher than the co-volume density. Air trapped in the snowpack dissipates energy through snow compression with air expulsion, a process not well captured in depth-averaged models. As a result, simulated impacts can produce unrealistically large flow heights.

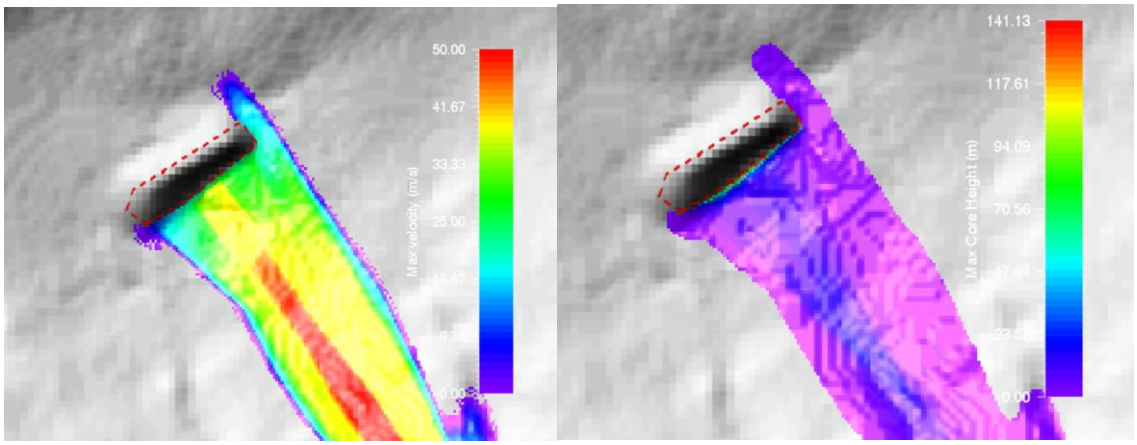


Figure 2.13: Using a shapefile (dashed red line) to define no-flux boundary conditions for the avalanche core. Left: The no-flux boundary prevents the avalanche from flowing into the shapefile. Right: The high-velocity impact results in unrealistic flow heights. Flow heights on the no-flux boundary are over 100m high. Care should be taken when using no-flux boundaries as stopping conditions under high-velocity impact.

To mitigate this issue, users may need to supplement **no-flux boundaries with localized friction shapefiles** placed before the dam or obstacle. These friction zones help dissipate energy more accurately, preventing exaggerated flow heights and their subsequent effects on powder cloud development. Without such measures, unrealistically large core and powder cloud heights can also be generated when braking dams are impacted (see Figure 2.13).

This issue does not arise when using no-flux boundaries as deflectors rather than stopping barriers.

Applying no-flux boundaries to the powder cloud as a stopping mechanism can also lead to unrealistic powder cloud heights. **To maintain realistic simulations, users should**

carefully distinguish between stopping and deflecting barriers when setting up no-flux boundaries.

2.11.2.3 Curvature

RAMMS::Extended incorporates curvature effects by adding centripetal accelerations to the hydrostatic normal forces, accounting for the influence of slope curvature on avalanche dynamics. Unlike **RAMMS::Avalanche**, where a smoothed DEM is used to approximate the effect of mountain snow cover—reducing extreme curvature variations—**RAMMS::Extended** directly calculates curvature from the DEM. **This results in more precise adjustments to the normal force, which can either increase or decrease depending on local terrain curvature.**

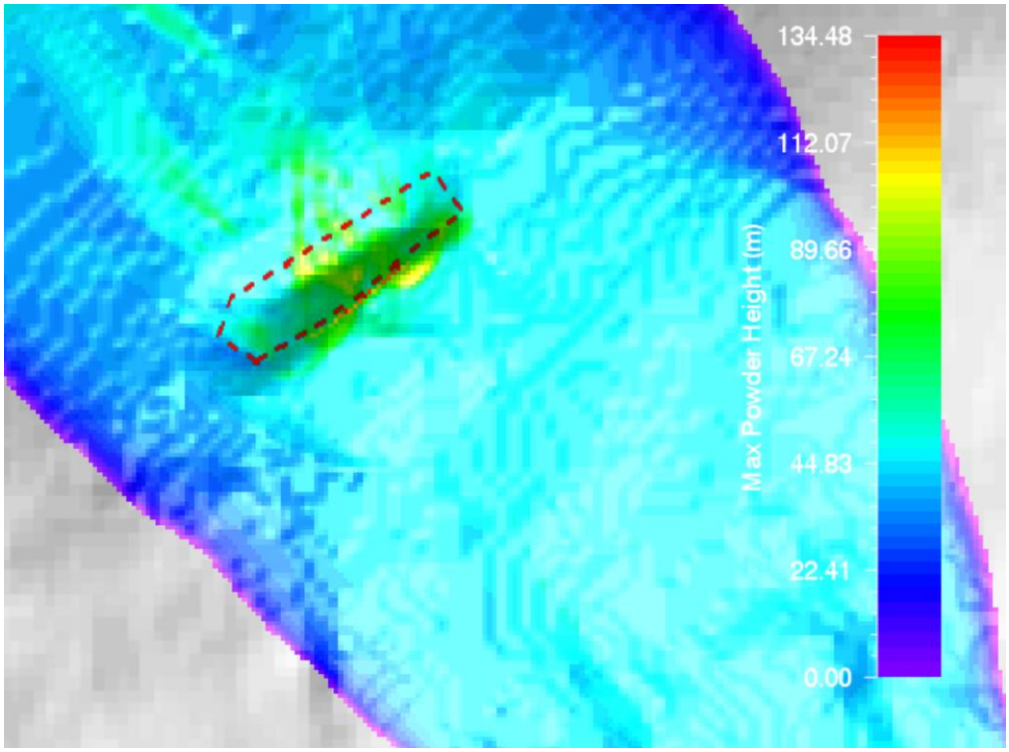


Figure 2.14: The core is stopped by the dam, but the powder cloud flows over the dam and is not affected by the no-flux boundary acting on the core. Powder cloud heights can be unrealistically high (in this case over 100m) as the core heights are unrealistically large.

Impact of Curvature in Gullies

While **curvature effects** enhance accuracy in many scenarios, they can introduce **unrealistic deceleration and reduced runout distances in steep, V-shaped gullies**.

In such terrains, the direct curvature calculation can generate excessive normal accelerations, leading to an unnatural loss of avalanche velocity. This occurs because

deep gullies often exhibit curvature singularities—sharp variations that are naturally smoothed by **accumulated snow cover** and **past avalanche deposits**.

Additionally, snow falling from steep gully walls often fills in these depressions, further altering the actual curvature of the terrain.

Curvature Settings: When to Use and When to Avoid

In **RAMMS::Extended**, users can choose to include or exclude curvature effects through the **Curvature Input Mask**:

- **Recommended ON** for Open-Slope Avalanches: For broad, open slopes, enabling curvature improves accuracy by accounting for natural slope variations.

- **Not Recommended for Steep V-Shaped Gullies**: In steep, confined gullies, excluding curvature often produces more realistic results, avoiding excessive deceleration due to artificially strong normal forces.

Since terrain and snow conditions vary, we **encourage users to experiment with curvature ON and OFF, especially in gully simulations, to determine the most realistic representation of avalanche motion**.

2.11.3 Friction Tab

One of the most **critical factors in avalanche dynamics calculations** is the precise modeling of flow resistance within the avalanche core (Figure 2.15). This resistance not only governs the motion and behavior of the **core** itself but also plays a pivotal role in shaping the **formation and evolution** of the **powder cloud**, the airborne component of an avalanche that significantly impacts its reach and destructive potential. Chapter 3 discusses the friction problem, from theory to input. Here, we provide a general overview of the Friction Tab.

In **RAMMS::Extended**, the **Friction Tab** is the dedicated tool for configuring these essential parameters.

1. **Defining Core Baseline Friction Values** $S_{\Phi}(\mu_0, \xi_0, N_0)$.
2. **Introducing Shapefiles for Localized Core Friction** $S_{\Phi}(\mu_0, \xi_0)$.

In **RAMMS::Extended**, baseline friction values (see definition Chapter 1) are automatically assigned based on two key factors:

- **Avalanche Return Period** – This determines the likelihood of an avalanche occurring within a given timeframe, influencing friction calibration.

- **Avalanche Size** – This is indirectly specified by the release area shapefile size and the fracture height (d_0).

A full discussion on return periods, friction values, and avalanche release volumes is provided in **Chapter 3**, **Section 4**. Here, we simply note that the user must manually select the avalanche return period, while the release volume is defined by the input shapefile and fracture height parameter.

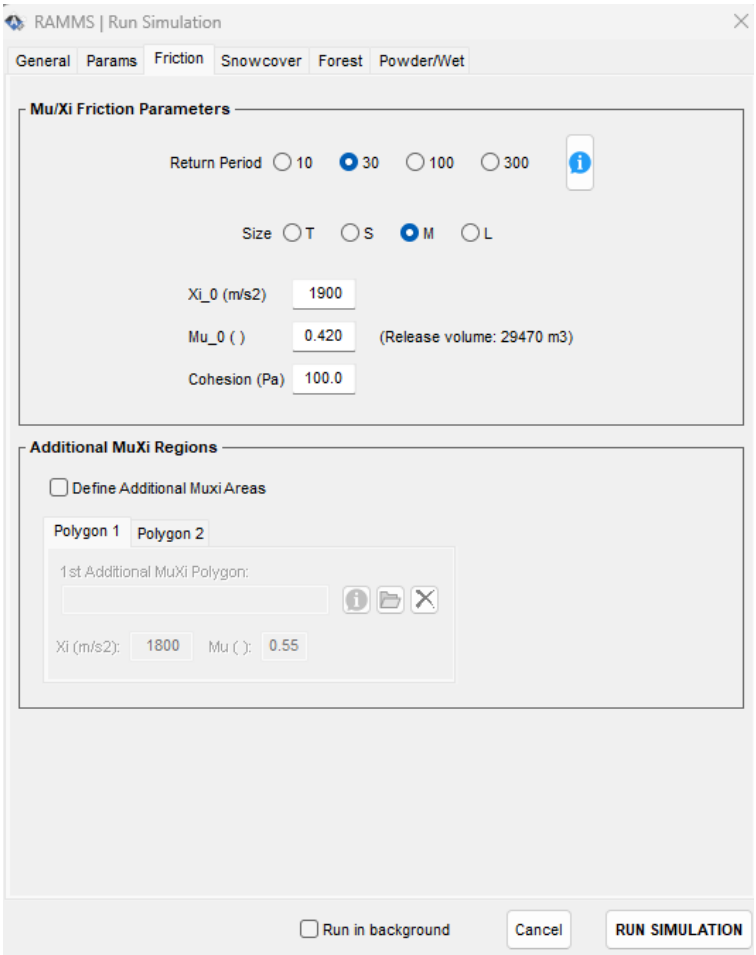


Figure 2.15: The Friction Tab contains two input panels. The upper panel allows the user to define a return period and avalanche size category. Based on the return period and size, the friction values are automatically selected, see Chapter 3 for details.

2.11.4 Snowcover Tab (Snowcover Entrainment)

The **Snowcover Tab**, found in the **Run Simulation panel**, provides structured control over how snowcover is represented spatially within the simulation domain (Figure 2.16). This is a core component of the **RAMMS::Extended** philosophy: entrainment and flow behavior depend critically on *where* snow is and **what kind of snow it is**. The snowcover is not defined by a single value—but by **field variables** that can vary with **elevation, slope angle, snow type, and terrain structure. All input is discussed in detail in Chapter 4, Snow Entrainment.**

The tab is organized into **four main sections**, each addressing different physical aspects of the snowcover: **Reference Altitude, Erosion Parameters**, **Densities and Release Information.**

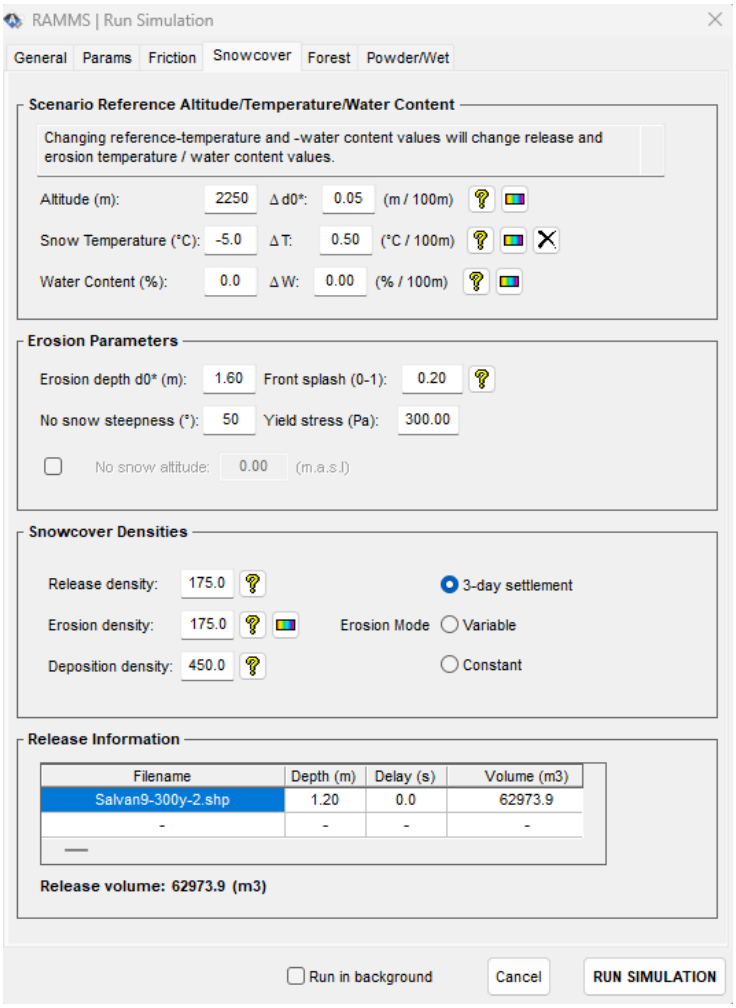


Figure 2.16: The Snowcover Tab contains four input panels: Reference Altitude, Erosion Parameters, Snowcover Densities and Release Information. See Chapter 4 for more details.

2.11.4.1 Reference Altitude

This section defines a **reference elevation** Z_0, from which vertical gradients in snow temperature, height, and water content are applied. This allows realistic snowpack conditions to be projected across a 3D mountain terrain, reflecting how snow properties change with altitude.

Input fields:

- **Reference Altitude (Z_0):** The baseline elevation for snowcover definition. Often the release elevation or a measurement station.
- **Snow Temperature** (T_Σ)**:** Mean snow temperature at Z_0. Not the surface or air temperature, but an internal average (e.g., –6 °C).
- **Water Content** (m_Σ): Liquid water content (in %) at the reference level.
- **Snow Height Gradient (Δh_Σ):** Defines the decrease in snow depth (m) per 100 m of vertical descent.
- **Temperature Gradient (ΔT_Σ:):** Temperature increase (°C) per 100 m elevation loss (e.g., 0.3 °C/100 m).
- **Water Content Gradient (Δm_Σ)**: Change in water content (%) per 100 m elevation drop.

Visualization tools: Each gradient input is linked to a **colorbar** that displays its spatial distribution across the terrain, providing intuitive feedback for snow property variation.

2.11.4.2 Erosion Parameters

This section controls how and how much snow is entrained during the simulation. **RAMMS::Extended** uses a spatially varying snowcover defined by the inputs in this tab to simulate realistic erosion behavior across complex terrain (Figure 2.17).

Input fields:

Erosion Depth (h_Σ): Snow depth on a 28° slope at Z_0. Often corresponds to the **3-day new snow depth** $h_\Sigma = d_0^*$:

- Adjusted for:
    - **Elevation gradient** (Δh_Σ)

- **Slope correction $f(\psi)$:** Reduces snow depth on steep slopes based on a stability function.

- **No Snow Steepness:** Critical slope angle (e.g., 60°) above which snow cannot accumulate. Used to mask out steep cliffs and rock faces in the snowcover.

- **Splash (γ_Γ):** Fraction of eroded snow that is splashed forward, moving at double the avalanche speed. Typically set to γ_Γ = 0.2 (20%).

- **Yield Stress (τ_Σ):** Minimum stress required for snow entrainment:

  - τ_Σ **=0 Pa**: **No resistance**; full entrainment

  - τ_Σ **=300 Pa**: **Standard value** for typical 3-day new snowcovers

  - τ_Σ =1000 Pa: **High resistance**; restricts lateral or secondary arm formation

Higher yield stress limits lateral entrainment and suppresses secondary flow arms, focusing flow within the main avalanche path.

2.11.4.3 Release, Erosion and Deposition Densities

This **section sets the initial density of snow in the release area** ϱ_0, the baseline **erosion density** ϱ_Σ and the **deposition (or co-volume) density**.

Input fields:

- **Release Density (ϱ_0): Average density in the release zone. Typical values range: Dry, cold snow: 150–175 kg/m³; Standard: 200 kg/m³; Wet, dense snow: ≥300 kg/m³.**

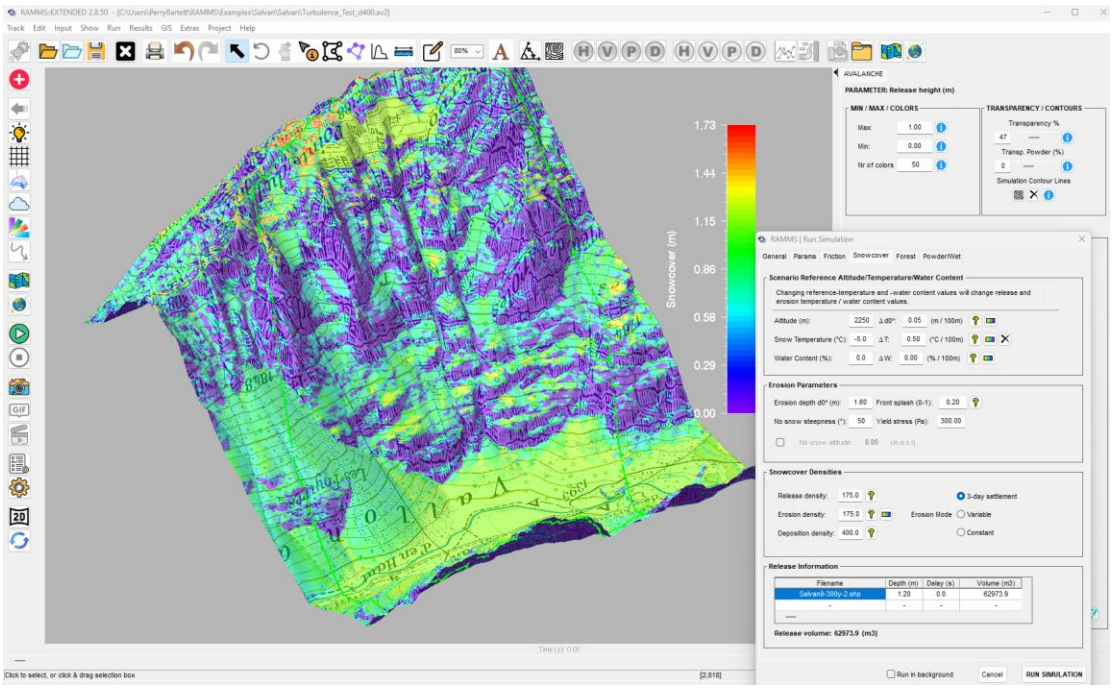


Figure 2.17: The Snowcover tab is used to generate an erodible snowcover across three-dimensional terrain. Snow depth and density are spatially distributed based on slope angle and temperature, using predefined baseline density values.

- **Baseline Erosion Density:** Users first select the baseline density ϱ_{Σ} of the erodible layer. This density distribution is then calculated using one of three methods:

    - **3-day Settlement: RAMMS::Extended** calculates the density of the erodible layer according to the formula of Kojima which is a function of overburden and temperature.

    - **Variable:** Adjusts baseline density according to slope angle, not temperature dependent.

    - **Constant: Prescribes constant snowcover density. Ideal for wet snow avalanches.**

- **Deposition Density**: Co-volume density $\hat{\rho}_{\Phi}$ of deposited snow, typically $\hat{\rho}_{\Phi} =$ 450 kg/m³. This is the density of the avalanche core when fully compacted and is used to calculate final flow height.

2.11.4.4 Summary of Release Information

To support the creation of **internally consistent hazard scenarios**, the Snowcover Tab repeats key **release zone information**—including **fracture depth** and **fracture volume**—originally defined in the release input. By displaying this data directly within the Snowcover interface, users can more easily calibrate the **amount and density of erodible snow** to match the release conditions. This tight coupling between release and entrainment parameters ensures that the snowcover is not only physically realistic but also **coherent across all stages of the avalanche event**, from initiation to deposition.

2.11.5 Forest Tab

The **Forest Input Tab** (Figure 2.18) allows users to define up to **three polygon-based forest shapefiles** and assign a range of **biophysical and snow interaction parameters** for each. This tab is essential for modeling **avalanche–forest interactions**, including **tree destruction**, **snow detrainment**, and **flow resistance** in vegetated terrain.

The interface is organized into **four functional sections**:

2.11.5.1Tree and Detrainment Parameters (Upper Section)

This section enables users to assign both **tree-specific** and **avalanche detrainment parameters** to each forest shapefile:

- **Tree parameters** include **species**, **height**, **diameter at breast height (DBH)**, and **breaking stress**.
- A key parameter in modeling snow retention is the **detrainment code (K_Ψ)**, which defines the **rate of avalanche snow detrainment** within forested areas. This code is based on **tree stem density** and **ground surface roughness**, and its values are defined in the Forest Detrainment Manual.
- Users can also set the **snow interception rate** during **three-day new snowfall events** to simulate forest canopy effects on snow accumulation.
- The choice of **tree species** (currently **Spruce**, **Pine**, **Larch**, and **Birch**) directly affects the **breaking stress**, with **Larch** and **Birch** providing the highest resistance to avalanche forces. (To add additional species, please contact the RAMMS development team.)

2.11.5.2 Snowcover Characteristics

This section manages **snow interception behavior** through a dropdown menu offering three options:

1. **No snowcover** – full interception by the forest canopy.
2. **Interception based on crown density** – controlled by the assigned **K-value**.
3. **No interception** – assumes a fully **erodible snow cover** under the forest canopy.

These settings help define the amount of snow available for entrainment in forested regions at the onset of the avalanche.

2.11.5.3 Avalanche Behavior After Tree Destruction

Once **RAMMS::Extended** determines that trees have been broken or uprooted, users can select one of two modeling responses:

1. **Reduce the K-value**, simulating decreased snow detainment capacity after tree destruction.
2. **Increase terrain friction**, representing enhanced **flow resistance and jamming** caused by fallen, overturned, or broken trees.

These options allow users to realistically simulate **post-destruction forest dynamics** and their influence on avalanche flow behavior. Detailed explanations of these processes are available in **Chapter 7** of the manual.

2.11.5.4 User Manuals and Documentation

The lower section of the tab provides direct access to **user manuals**, including resources on **forest parameterization**, **detrainment coding**, and **tree species definitions** to support accurate model configuration.

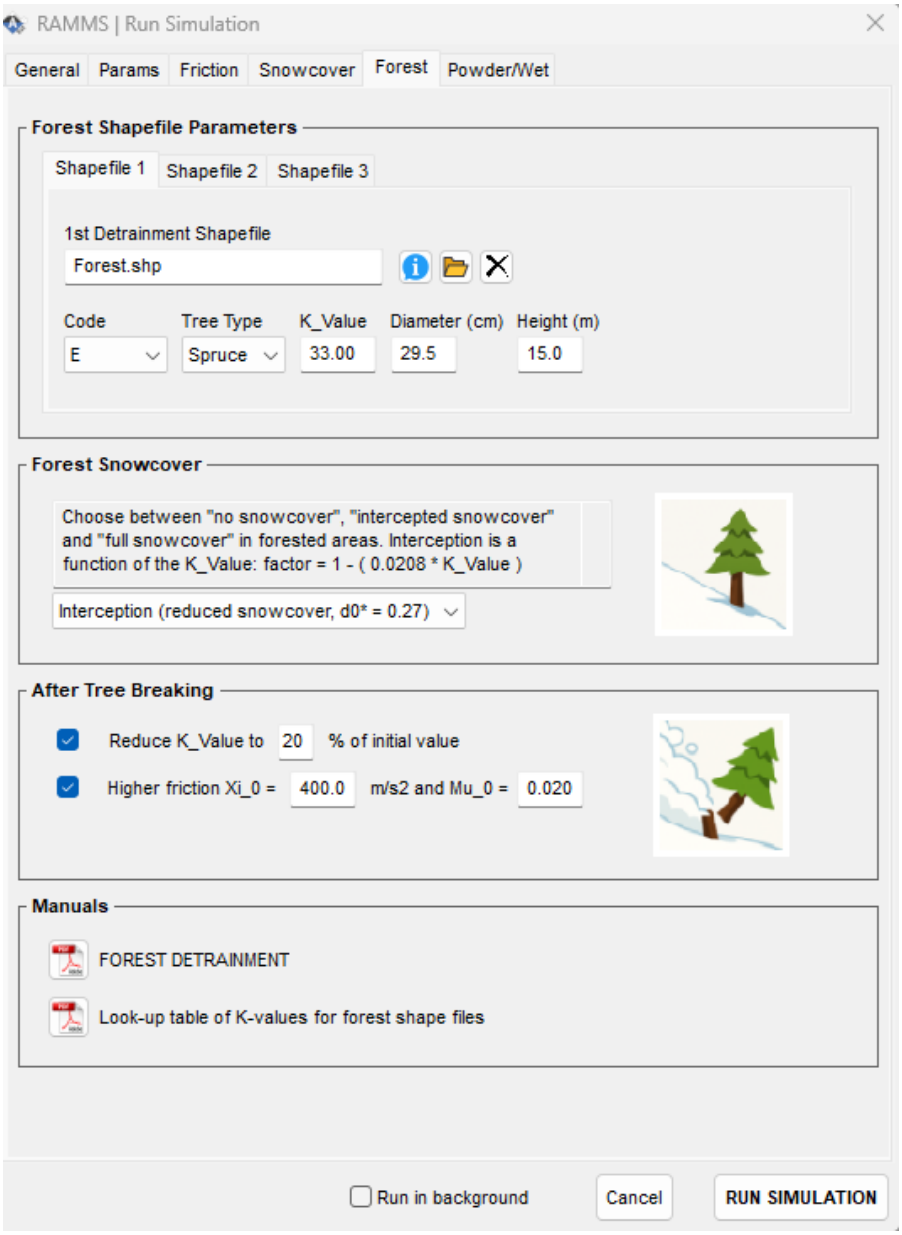


Figure 2.18: The Forest Input Tab allows users to define up to three polygon-based forest shapefiles and assign physical properties for each. The tab is organized into four sections: The upper section is used to assign tree parameters—such as species, height, diameter at breast height (DBH), and detrainment coefficient—to each shapefile. The second section defines the snowcover characteristics within the forested areas. The third section instructs the model how to simulate avalanche behavior after tree failure, including flow resistance and detrainment in destroyed forest. The bottom section provides access to user manuals and documentation for guidance on forest parameterization.

2.11.6 Powder/Wet Tab

The **Powder/Wet Tab** provides access to key parameters that steer the behavior of **powder-snow** and **wet-snow** avalanches in **RAMMS::Extended** (Figure 2.19). It is divided into several input regions, allowing the user to define both the **avalanche type** and the **physical processes** involved in the simulation. Much more information is found in Chapter 5 of this manual which concerns the powder cloud.

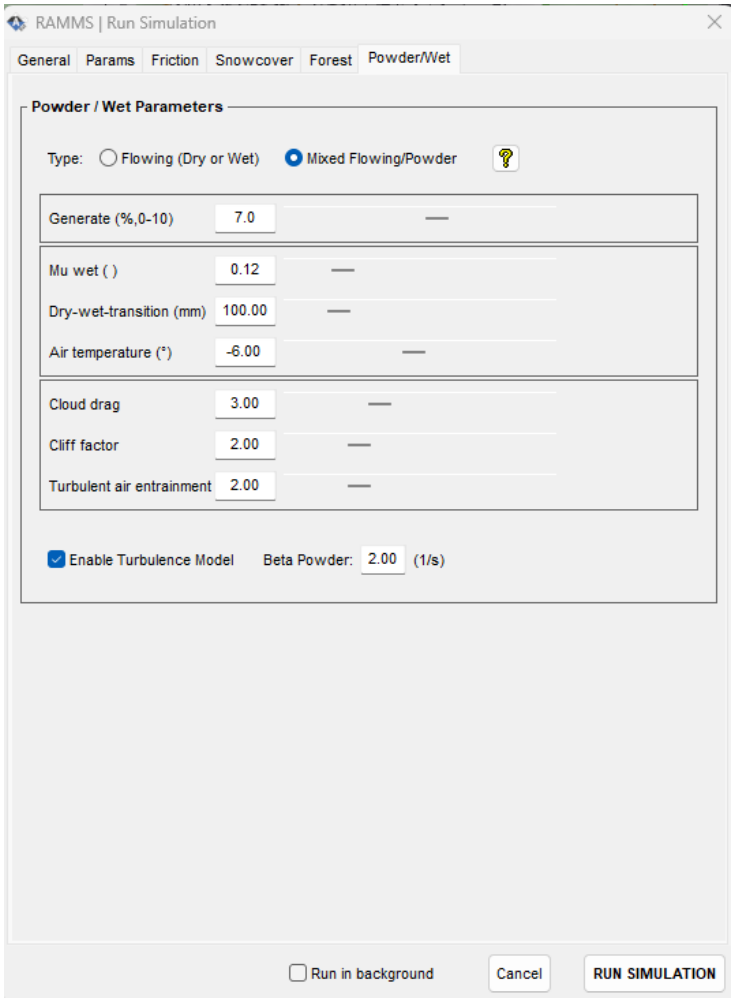


Figure 2.19: The Powder/Wet Tab is used to input values for wet snow and powder snow avalanches.

2.11.6.1Avalanche Type and Powder Cloud Activation

The upper section of the tab determines the **avalanche type** and whether the **powder cloud module** is active:

- **Flowing:** For dry or wet flow-only avalanches. No powder cloud is calculated.
- **Mixed Flowing/Powder:** Activates the powder cloud module for simulations that include suspension layers.

Note: For **wet snow avalanche simulations**, users should select **Flowing**, as these avalanches typically do not generate powder clouds.

2.11.6.2 Generate Parameter (α_Φ)

This parameter governs the **degree of dispersion and fluidization** in the avalanche core. It reflects how loosely the snow grains are packed and how readily they transition into a more fluid-like state. The input value is **in percentage (%).**

- For **wet snow avalanches**, select $\alpha_\Phi \leq$ **0.05** (5%)
- For **powder-snow avalanches**, it is recommended to use a **constant** $\alpha_\Phi =$ **0.07** (7%)

Lower values produce **plug-like flows**, suitable for wet or dense snow; higher values produce **dispersive flows**, suitable for dry, turbulent avalanches. The parameter α_Φ controls the dispersion of the avalanche core and therefore the strength of the powder cloud formation.

2.11.6.3 Wet Snow Parameters

When modeling wet-snow avalanches, the following parameters are important:

- **Mu_wet:** Defines the **basal friction coefficient** μ_w when snow grains are lubricated by a thin film of meltwater. This value is similar to the **RAMMS::Avalanche** parameter set. For wet avalanches $0.12 \leq \mu_w \leq 0.2$.
- **Dry_Wet_Transition:** Specifies the **water content threshold** (in mm) at which the model transitions from dry to wet flow behavior.
    - A typical value is **10 mm**
    - Lower values (e.g., **1 mm**) result in a **sharper, more sudden transition** from dry to wet flow regimes
- **Air Temperature (T_Λ):** The **ambient air temperature** is also defined in this tab and influences the **formation of wet-snow avalanches**:
    - **Higher temperatures** promote melting and the generation of wet, plug-like avalanches.
    - **Colder temperatures** support powder-snow avalanche behavior.

- Unless actual air temperatures have been measured, **it is recommended to use mean snow temperatures,** operating on the assumption of **a thermal equilibrium** between air and snow temperatures $T_\Lambda = T_\Sigma$**.**

These inputs allow the simulation to represent the **continuous evolution** of snow behavior as meltwater content increases, impacting runout and flow dynamics.

2.11.6.4 Powder Cloud Parameters

The final section contains input fields specific to **powder-snow avalanche** dynamics:

- **Cloud Drag (δ_Π):** Controls the overall resistance acting on the powder cloud, combining both laminar and turbulent drag forces into a single input.
  - **Influences:** Cloud velocity and flow attenuation.
  - **Typical value: δ_Π = 3.0** for medium to large powder avalanches.
  - Drag coefficients scale linearly with δ_Π.
- **Turbulence Decay (β_Π):** Specifies the rate at which turbulent energy dissipates in the cloud.
  - **Required only when the Turbulence Model is enabled.**
  - **Low β_Π** → Longer turbulence lifetime, resulting in taller clouds.
  - **High β_Π**→ Rapid decay, leading to a transition to laminar flow.
  - **Recommended range: β_Π = 1.0–2.0 /s**.
- **Cliff Factor (λ_Π):** Enhances the mass and momentum transfer from the avalanche core to the powder cloud when avalanches descend steep slopes (cliffs).
  - Active at slopes **>35°**, peaking at **55°**.
  - Higher values (λ_Π **=5.0)** increase cloud formation during cliff drops.
  - **Recommended range: λ_Π = 2.0–5.0**.
- **Turbulent Air Entrainment (ψ_Π):** Determines the rate of air entrainment into the powder cloud due to turbulence.
  - Affects cloud height, density, and momentum.
  - Strong air entrainment occurs at the cloud front; minimal in the wake.
  - **Default value: ψ_Π = 2.0**; usable range: **1.0–6.0**.

2.11.6.5 Summary

The **Powder/Wet Tab** is integral for defining avalanche types and tailoring simulations to specific snow conditions. **Most user-defined model parameters** are input in the **Powder/Wet Tab,** see Table 2.1.

Cloud Parameter	What it does
Generate Parameter (α_{Φ})	Controls the particle dispersion in the avalanche core, and ultimately the amount of mass, volume, momentum and turbulent energy in the cloud formation.
Cloud Drag (δ_{Π})	Controls the overall drag resistance acting on the powder cloud, combining both laminar and turbulent drag forces into a single input.
Turbulence Decay (β_{Π})	Specifies the rate at which turbulent energy dissipates in the cloud.
Cliff Factor (λ_{Π})	Enhances the mass and momentum transfer from the avalanche core to the powder cloud when avalanches descend steep slopes (cliffs).
Turbulent Air Entrainment (ψ_{Π})	Determines the rate of air entrainment into the powder cloud due to turbulence.

Table 2.1: Powder cloud parameters.

See Chapter 5 for details of the model parameters. By configuring parameters related to avalanche type, dispersion, wet-snow behavior, air temperature, and powder cloud dynamics, users can model a wide range of avalanche scenarios within **RAMMS::Extended**. Each parameter influences the evolution, runout, and impact of the powder cloud. Collectively, they enable **RAMMS::Extended** to simulate the transition from turbulent flow to extended propagation, capturing the dynamics of powder-snow avalanches across complex terrain.

2.12 Output

2.12.1 Quick Access Buttons for Core and Cloud Results

The **RAMMS::Extended** graphical user interface (GUI) offers a variety of intuitive tools for analyzing and visualizing simulation output. A central feature of this interface is a dedicated set of **eight display buttons**, conveniently located in the **center of the upper toolbar** (Figure 2.20). These buttons provide rapid access to key result metrics from your simulation.

Display Buttons Overview

The display buttons are color-coded and organized into two groups:

- **Green buttons**: Represent output for the **avalanche core**.
- **Yellow buttons**: Correspond to the **powder cloud**.

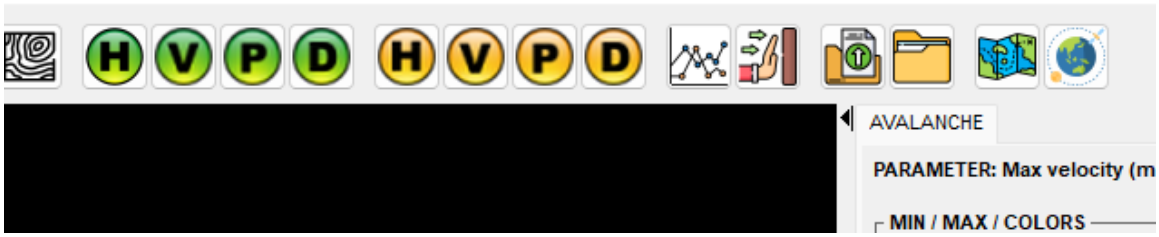


Figure 2.20: The eight Quick Access Buttons allow for the rapid display of maximum simulation outputs, including flow height (H), velocity (V), impact pressure (P), and flow density (D). Green buttons correspond to the avalanche core, while yellow buttons display results for the powder cloud.

Each group includes four buttons, allowing users to instantly display the following maximum values:

- **Flow height (H)**
- **Flow velocity (V)**
- **Impact pressure (P)**
- **Flow density (D)**

These buttons are specifically designed for **quick visualization of maximum values** computed during the simulation. While these maximum values are static and cannot be animated, users can enhance the visualization by overlaying **contour lines** to better interpret spatial variations across the model domain.

To facilitate further analysis or use in external applications, **RAMMS::Extended** includes a variety of **export tools (see section 2.18)**. At any point, users can swiftly export the

displayed max value data to **GeoTIFF** or **ASCII grid (.asc) files**, ensuring seamless integration with GIS platforms or post-processing software.

2.12.2 Display Tab

Located on the **right-hand side** of the **RAMMS::Extended GUI**, the **Display Tab** is the third tab in the interface, following the Files and General tabs (Figure 2.21). As one of the most frequently used components of the interface, the Display Tab provides quick access buttons for exploring nearly all simulation output.

The tab is organized into four vertically stacked regions, each designed for specific types of interaction and data control.

2.12.2.1 Core/Cloud Toggle & Colorbar Settings

Located at the top of the **Display Tab**, users can:

- **Activate or deactivate** the core and/or powder cloud result layers using dedicated check boxes.
- Adjust the **colorbar text color** by clicking directly on the colorbar icon. This feature is particularly useful for optimizing contrast, especially when the background color of the GUI is changed.

2.12.2.2 Dump Step Values

This section features **two horizontal rows of buttons**, each labeled **H**, **V**, **P**, and **D**, representing:

- **H** – Flow Height
- **V** – Flow Velocity
- **P** – Impact Pressure
- **D** – Flow Density

The **top row (red buttons)** displays values for the **avalanche core** at the specific time step selected via the **time slider**. The **bottom row (blue buttons)** performs the same function for the **powder cloud**. These controls are essential for switching the GUI into **time-step mode**, enabling users to initiate **animations** or conduct **frame-by-frame analysis**. They are particularly useful for closely examining the temporal evolution of key variables, providing deeper insight into the dynamics of both the core and the powder cloud over time (Figure 2.22).

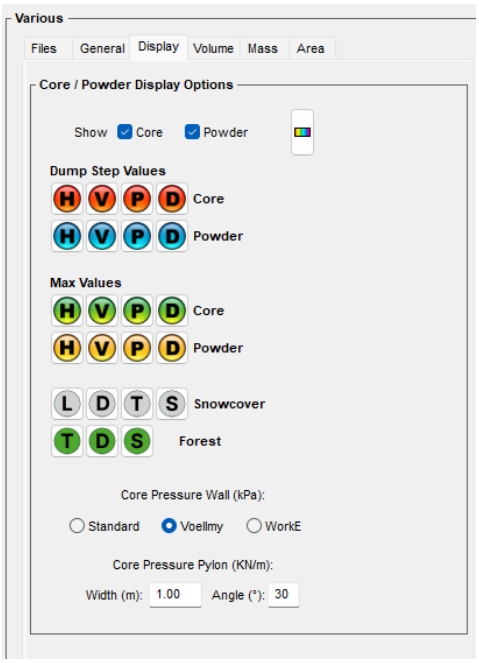


Figure 2.21: The Display tab is one of the most used parts of the RAMMS::Extended interface. It can be used to display both dump-step values using the time slider, as well as max values. The Display tab is also used to quickly display snowcover and forest destruction. It is also used to control impact pressures on walls and pylons.

2.12.2.3 Maximum Values (Core and Cloud)

This region also features **two additional rows of buttons** labeled **H**, **V**, **P**, and **D**, used to display the **maximum values** of each parameter over the entire simulation for the **core (green buttons)** and **cloud (yellow buttons).** These are the same button defined in the tool bar at the top of the GUI.

As previously defined:

- **H** – Maximum flow height
- **V** – Maximum velocity
- **P** – Maximum impact pressure
- **D** – Flow density

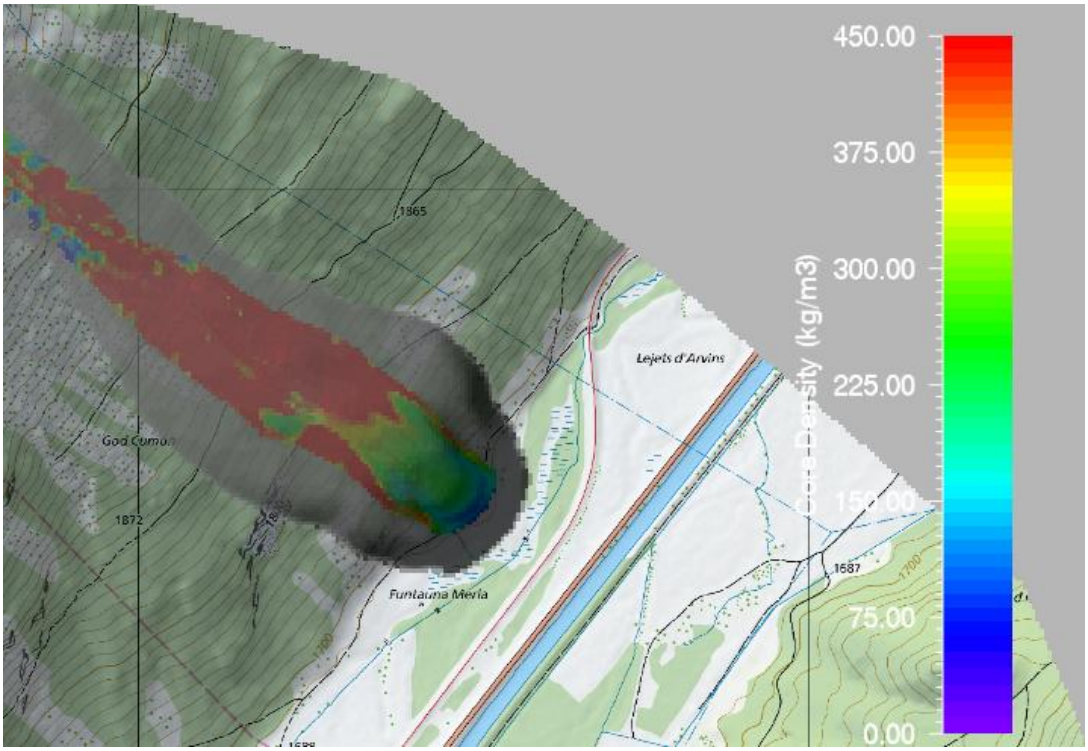


Figure 2.22: The red "D" button displays the avalanche core density at the selected time step *t*. This visualization is particularly useful for analyzing the internal structure of the core, from the low-density saltation front at the leading edge to the progressive densification toward the tail of the flow.

While the **green "D" density button** appears alongside the other maximum value controls, it functions differently: rather than displaying the **maximum flow density**—which would simply correspond to the **depositional density** at rest—it instead shows the **minimum flow density** encountered during the simulation. This provides valuable insight into the **fluidization of the avalanche front**, making it possible to identify and analyze **low-density regions** such as **splashing or saltating fronts**.

These buttons are especially useful when examining **powder avalanche impact pressures and flow densities** (Figure 2.23), where fluidized fronts can cause high-speed, low-density impacts. in conjunction with **contour plots**, which provide clearer spatial context and highlight gradients within the flow.

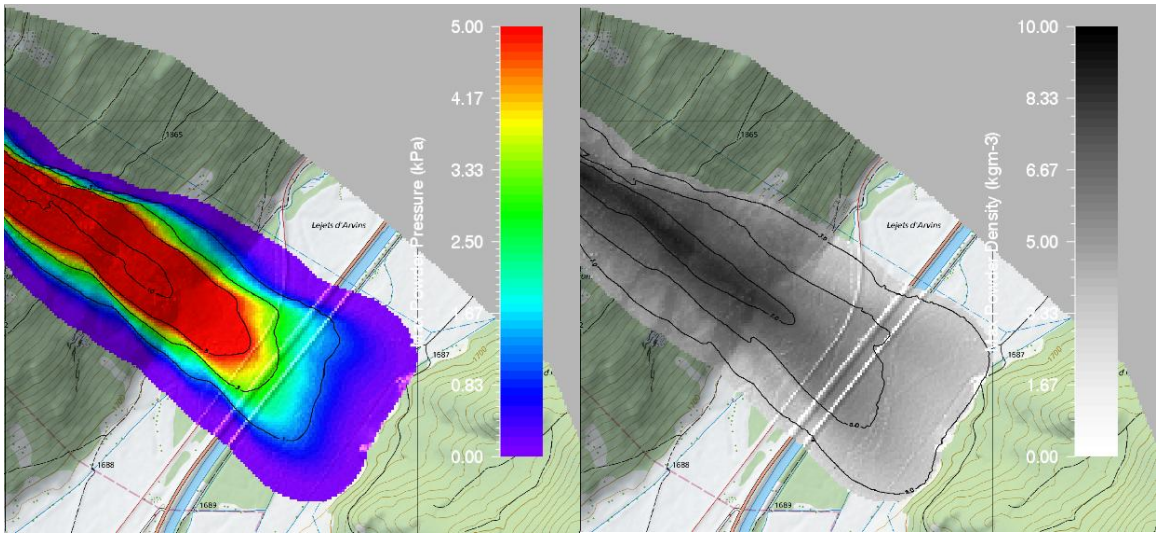


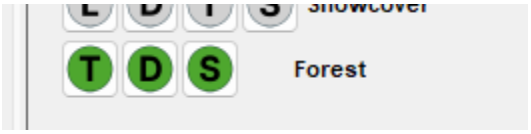
Figure 2.23: Maximum powder pressures (max 5kPa) and maximum powder cloud densities (max 10kg/m3). Do display the maximum densities RAMMS::Extended employs a white to black colorbar. These visualizations are created with the yellow P and D buttons.

2.12.2.4 Forest Interaction and Destruction

At the **bottom of the Display Tab**, **RAMMS::Extended** provides **three green buttons** for visualizing forest-related outputs derived from avalanche–forest interactions. These tools offer valuable insight into how the avalanche impacts vegetated terrain, especially in areas where forest dynamics play a critical role in flow resistance and snow detrainment.

The three forest output buttons are labeled:

- **T** – Tree Destruction
- **D** – Snow Detrainment in Forests
- **S** – Maximum Tree Bending Stress



T – Tree Destruction

The **T button** highlights **forest regions** that have been **damaged or destroyed** by either the **avalanche core**, the **powder cloud**, or both (Figure 2.24). These regions correspond to **forest polygons** defined in the **Forest Input Tab** (see **Chapter 7** for a detailed explanation of forest modeling).

Each tree polygon is characterized by:

- **Tree species**
- **Tree height**
- **Diameter at breast height (DBH)**

- **Breaking stress**

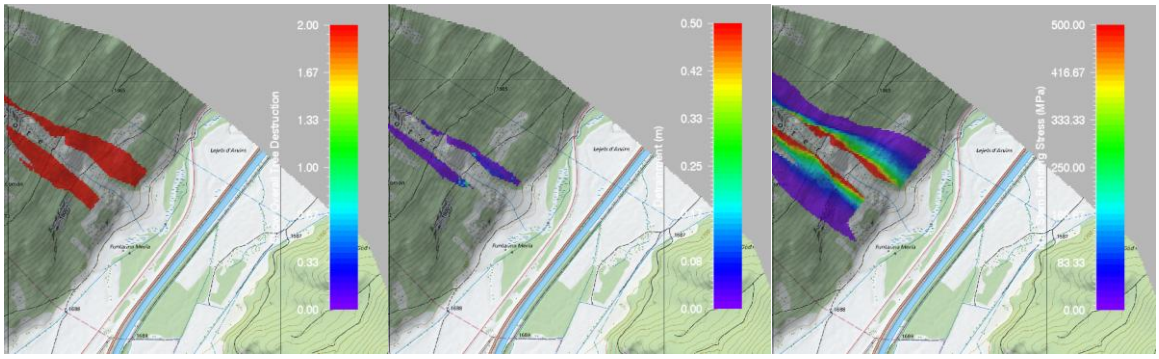


Figure 2.24: Forest Interaction Visualization using the three Green Buttons in the Display Tab. Left panel: Tree destruction displayed using the T button, indicating areas impacted by the avalanche core, powder cloud, or both. Middle panel: Snow detrainment visualized with the D button, showing the average snow accumulation within forested regions. Right panel: Maximum tree bending stress displayed using the S button, useful for assessing structural loading on vegetation. All results are confined to the forest polygon shapefile defined in the Forest Tab.

RAMMS::Extended calculates the **bending and overturning moments** exerted on the trees by both avalanche components. The destruction is then visualized using a **coded color scale**:

- **1** – Trees destroyed by the **avalanche core**
- **2** – Trees destroyed by the **powder cloud**
- **3** – Trees destroyed by **both** components

In many cases, tree failure is primarily caused by the **powder cloud**, which generates larger **bending moments** due to longer **torque arms**—the result of high-impact forces applied at greater heights along the tree stem.

D – Snow Detrainment

The **D button** displays the amount of **avalanche snow detrained** (in meters) within forested areas (Figure 2.24). This value represents the **average snow height** that has settled and accumulated around trees due to flow resistance. It's important to note that **localized snow pileups** at individual trees may exceed this mean value significantly, especially where the avalanche flow encounters dense vegetation.

S – Maximum Bending Stress

The **S button** visualizes the **maximum bending stress** (Figure 2.24) experienced by trees across the forested polygons. This output can be directly compared to the **tree species-specific breaking stress** defined in the forest input parameters. It provides a useful tool for **validating forest input data**—helping users determine whether the assigned **stem**

diameters and **mechanical properties** are **realistically calibrated** or potentially **too strong or weak** for the given avalanche event.

2.12.2.5 Snowcover Visualization Tools

In the **lower section** of the **Display Tab**, users have access to four specialized buttons designed to visualize various aspects of the **erodible snow layer** and its evolution during the simulation (Figure 2.25

). These tools are invaluable for understanding **snow entrainment**, **thermal conditions**, and **surface changes** resulting from avalanche activity.

The four snowcover display buttons are labeled as follows:

- **L** – Erodible Snow Layer Height
- **D** – Erodible Layer Snow Density
- **T** – Snow Temperature
- **S** – Surface Difference

L – Erodible Snow Layer Height

The **L button** provides a frame-by-frame visualization of the **height of the erodible snow layer** throughout the simulation. This allows users to pinpoint **when** and **where** snow is being **entrained** into the avalanche. In essence, it shows the **instantaneous snow depth** available for erosion at each time step, offering a dynamic picture of how the flow interacts with the snowpack.

D – Erodible Layer Snow Density

The **D button** displays the **initial density** of the erodible snow layer—i.e., the density present **at the onset of the avalanche**. Unlike the height information, this value is static and does not change over time. Instead, it provides essential background data for understanding **mass transfer** from snow entrainment.

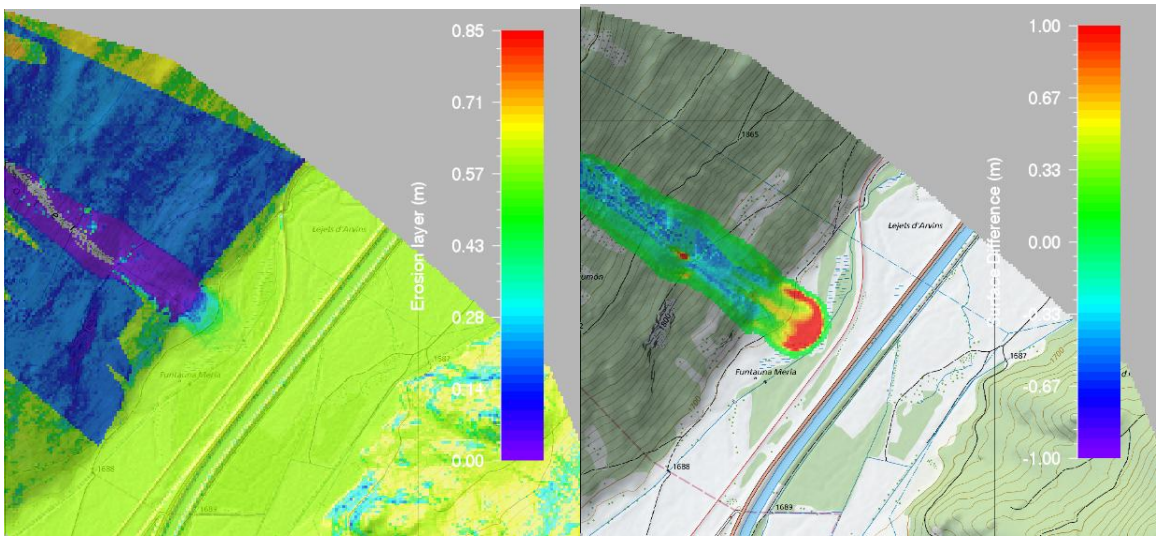


Figure 2.25: Snowcover and Entrainment Visualization Buttons. The left panel displays the erodible snow layer as the avalanche progresses into the runout zone. Areas shown in deep blue indicate zones of snow interception by forest cover, where snow is initially absent or reduced. The right panel illustrates the surface difference output. Here, red regions represent areas of snow deposition (positive height change), while blue regions highlight zones of snow entrainment (negative height change), where the avalanche has removed snow.

T – Snow Temperature

The **T button** visualizes the **temperature distribution** of the snow cover, typically revealing a gradient from **warmer, lower elevations** to **colder, higher elevations**. This information is crucial for assessing **snowpack stability**, as well as the **potential for melting** effects during the avalanche event.

S – Surface Difference

Perhaps the most powerful tool in this section, the **S button** calculates the **difference between the flow height and the depth of entrained snow**. This effectively reveals the **top surface elevation of the avalanche**. It is especially useful when comparing simulation output with **real-world drone-based surface measurements** taken before and after an avalanche. Negative values in this display indicate **erosion** (i.e., snow removed), while positive values show **deposition zones** where snow has accumulated.

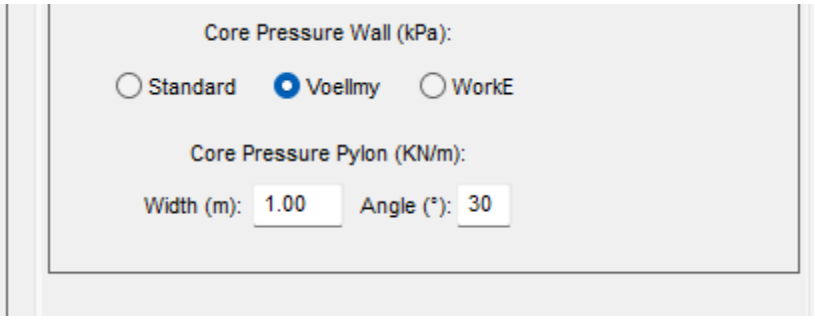
2.12.2.6 Impact Pressures: Walls and Pylons

At the bottom of the **Display Tab**, **RAMMS::Extended** offers detailed control over the calculation and visualization of **core impact pressures**, a crucial parameter for hazard mapping and infrastructure design. A key feature of the system is its ability to distinguish between **core impact on solid structures** (such as retaining walls) and **core impact on slender obstacles** (such as trees and pylons), enhancing the accuracy of hazard

assessments. For further details, see **Chapter 6**, which also includes recommendations for selecting the appropriate model.

Users can choose from three impact pressure calculation methods, controlled via intuitive **radio buttons**:

- **Standard**: The default method for general scenarios.
- **Voellmy**: A pressure model that incorporates snow compaction effects.
- **WorkE**: An energy-based approach to estimate impact pressures with enhanced precision.



RAMMS::Extended provides flexible options for calculating forces on pylons, offering three distinct methods: **Standard**, **Flat**, and **Round Obstacles**. Unlike conventional calculations that determine pressures (expressed in kN/m²), these methods compute **forces per unit height** (kN/m), which is particularly relevant for slender, column-like structures.

The **Flat** and **Round Obstacle** methods introduce additional precision by requiring the user to specify the **width of the obstacle** and the **deflection angle of the deadzone** parameter closely linked to the **internal friction angle** of the flowing snow. These refinements ensure an accurate representation of the interactions between avalanche flow and slender structures. For comprehensive explanations of these methods and guidance on their application, refer to **Chapter 6**.

2.12.3 Volume Tab (Volume Balance)

In **RAMMS::Extended**, users can compare simulated **avalanche volumes** with measured deposition volumes from field surveys, ensuring a more precise validation of model results. This can be achieved using the **Volume Tab**, located within the input panel on the right-hand side of the GUI (**Figure 2.26**). Volumes are displayed in m^3. The **Volume Tab** is positioned between the **Display** and **Mass** tabs.

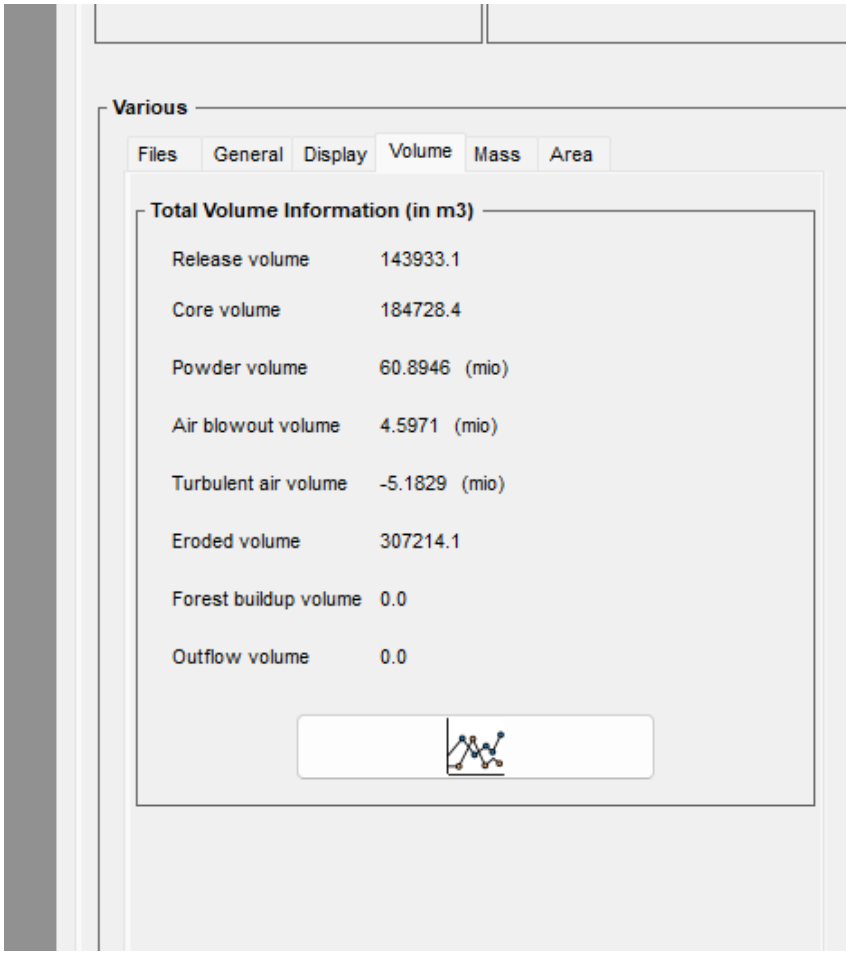


Figure 2.26: The Volume Tab displays numerical volume data corresponding to the current position of the time slider. As the time slider is adjusted, the displayed values update to reflect the selected time step. Additionally, the Plot button allows users to generate a time-series graph of the volume data, providing a visual representation of volume changes over time.

Using the Volume Tab

The **Volume Tab** provides real-time numerical volume data based on the **current position of the time slider**. As the slider is adjusted, the displayed values dynamically update to reflect the corresponding time step. This allows users to analyze how avalanche volume evolves throughout the simulation.

Additionally, clicking the **Plot button** generates a time-series graph of the volume data, offering a visual representation of total release volume, core volume, and entrained snow volume over time (Figure 2.27).

Interpreting the Volume Plot

- The **core volume** often decreases in the avalanche runout zone as the core densifies. The end value of the core volume often is compared to the measured

core volumes. The **density of the deposited volume** is taken to be the **co-volume density** (450kg/m^3).

- The point of **maximum core volume** often corresponds to the **strongest powder cloud formation**. At this point, the core volume is the largest, indicating a large particle dispersion and air intake, particularly at the avalanche front. Even with entrainment (mass growth), the avalanche volume often decreases.

- The **powder cloud volume** is represented in blue but must be multiplied by 1,000 to obtain the correct volume. The powder cloud volume often **overestimates measurements** because we never let the powder cloud volume decrease (no settlement). The cloud volume always grows, unless the cloud exists the simulation domain.

- If the **forest option** is enabled, the plot provides a numerical assessment of how much **mass is intercepted and stopped by the forest**, helping quantify the mitigating effects of vegetation on avalanche dynamics.

By leveraging these tools, users can effectively compare simulation outputs with real-world measurements, refining their understanding of avalanche behavior, deposition patterns, and environmental interactions.

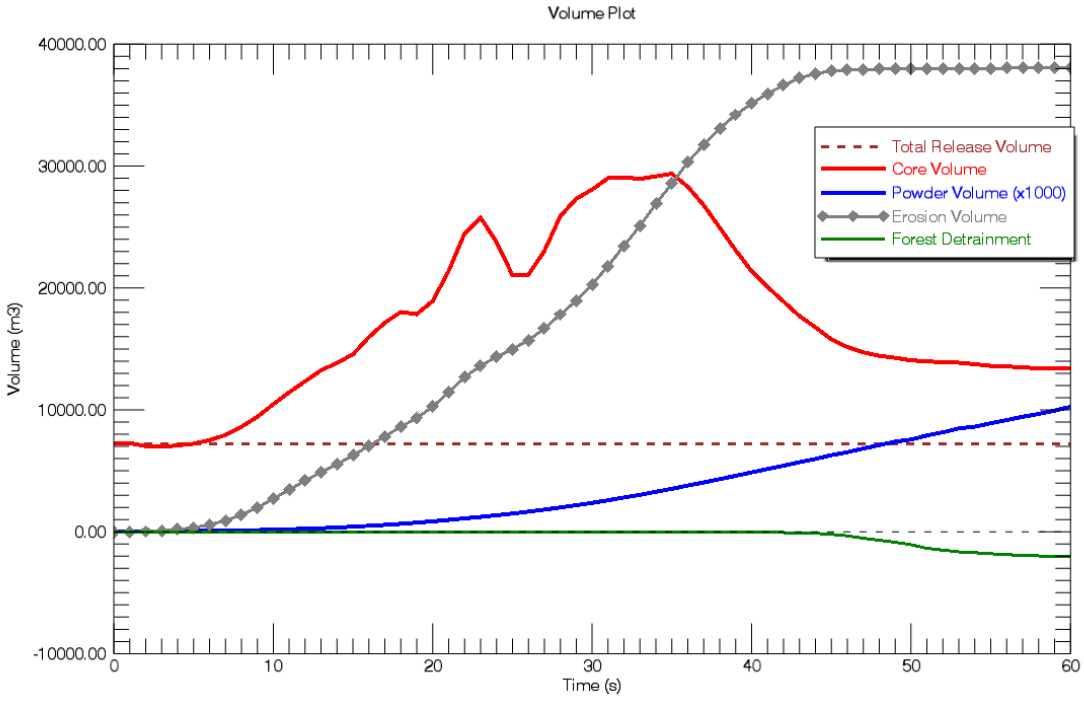


Figure 2.27: Volume Balance Plot – This figure illustrates the evolution of total release volume (dashed red line), core volume (thick red line), and entrained snow volume (gray line with diamonds) over time. Forest detrainment in green. The core volume decreases(!) as it densifies in the avalanche runout zone.

2.12.4 Mass Tab (Mass Balance)

Similar to the **Volume Tab**, the **Mass Tab** allows users to examine the **numerical evolution of mass in an avalanche and visualize its progression over time**, see Figure 2.28 and Figure 2.29. Mass is displayed in tons. However, unlike volume data—which can be challenging to interpret due to variations in avalanche density—**mass data provides a more straightforward analysis since it typically follows a strictly increasing trend**. A decrease in mass occurs only when the avalanche flows beyond the calculation domain.

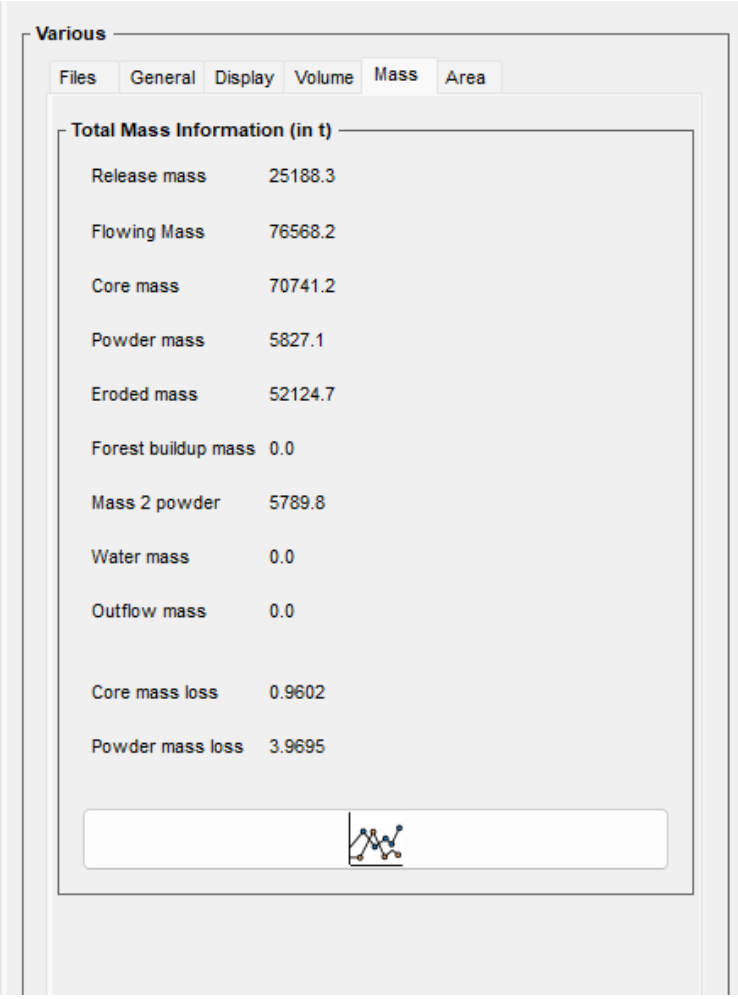


Figure 2.28: The Mass Tab presents numerical mass data (in tons) corresponding to the current position of the time slider. As the slider is adjusted, the values update dynamically to display the values at the selected time step. Users can also generate a time-series graph of mass and stopping data using the Plot button, offering a visual representation of mass and moving mass (expressed as a percentage of the maximum moving mass) over time.

Mass data serves **two critical functions**. First, it quantifies the amount of **moving mass relative to the maximum mass flux**, a key parameter for determining when to halt the avalanche simulation, see Figure 2.29, upper panel. The simulation stops when the mass

flow falls below a predefined percentage of the total. This percentage is defined in the **General Tab** of the input, see Section 2.6.1. Second, mass data is instrumental in assessing **growth factors**. For instance, a growth factor of 2 signifies a doubling of avalanche mass, rather than a mere increase in volume. Additionally, mass data is used to analyze the distribution of mass, such as identifying the portion suspended in the powder cloud versus the core.

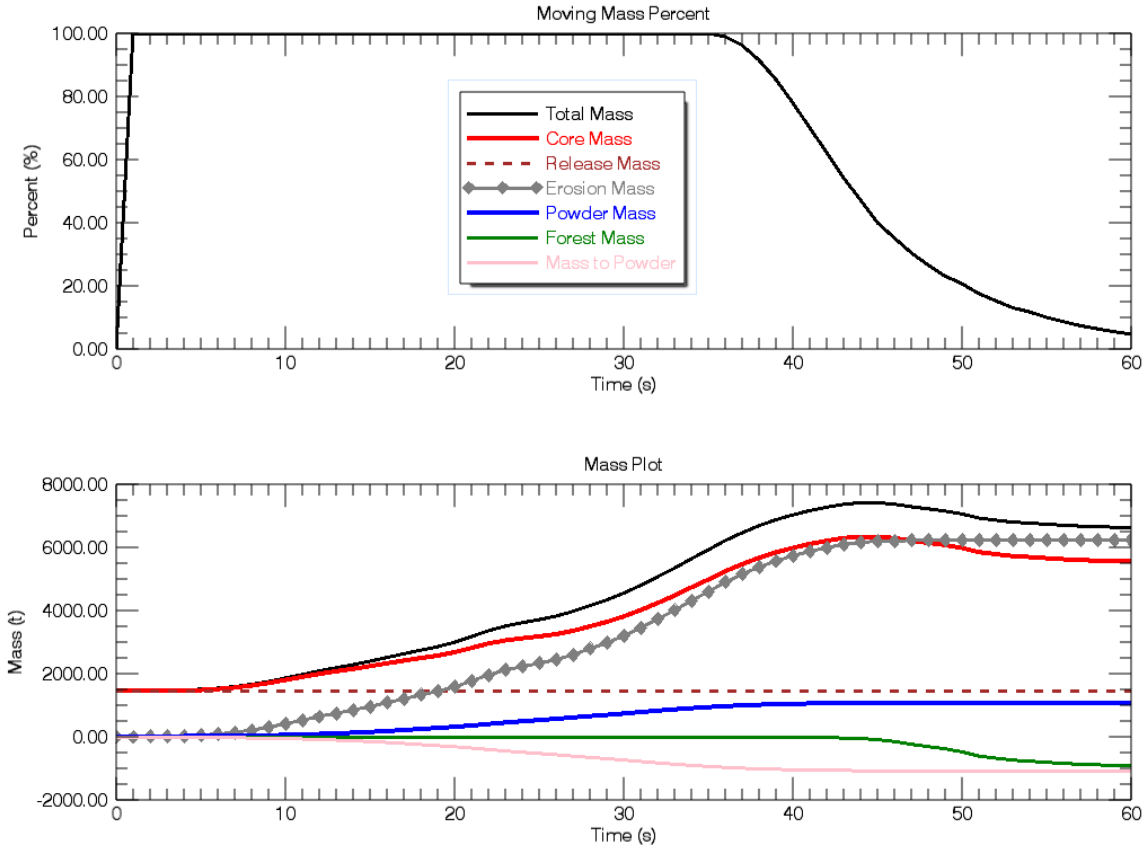


Figure 2.29: Mass Balance Plot – The top panel illustrates the moving mass over time as a percentage of the maximum total mass, which varies due to snow entrainment. The bottom panel shows the avalanche's mass evolution over time. Unlike volume plots (e.g., core volume), mass evolution usually increases. It can decrease when mass exits the calculation domain or is detrained in the forest. The upper black line represents the total avalanche mass, while the red line indicates the core mass, the blue line the powder mass. The mass detrained in the forest is shown in green. The entrained mass is shown in gray with diamond symbols. Detrained forest mass and the mass from the core to the cloud are negative, as they are removed from the core.

2.13Displaying the Powder Cloud in RAMMS::Extended

	**Cloud Visualization**	Clicking this icon causes the cloud visualization to appear/disappear

A particularly valuable feature of the **RAMMS::Extended** GUI is its ability to **visualize the powder cloud** dynamically. By default, the cloud is displayed as a **semi-transparent gray overlay** around the core, which can show core **height, velocity, pressure, or density**. This visualization is especially useful for analyzing the key differences between the core flow and the powder cloud, such as **variations in inundation area** and overall dynamics.

Toggling the Powder Cloud Visualization

The powder cloud display can be easily **activated** or **deactivated** using the **Cloud Icon** in the GUI.

- **Cloud ON**: By default, the transparent cloud is displayed over the core simulation results.
- **Cloud OFF**: Clicking the Cloud Icon removes the cloud overlay, allowing a clearer view of the core data alone.
- **Reactivate the Cloud**: Simply click the icon again to restore the cloud visualization.

This feature allows users to toggle between different visualization modes, helping to isolate specific elements of the simulation for better analysis.

Adjusting Cloud Transparency

The **TRANSPARENCY/CONTOURS** panel, located in the **upper right-hand corner** of the GUI, provides precise control over the powder cloud's transparency:

- **0% Transparency (Fully Opaque)**: The cloud becomes solid, obscuring all simulation data beneath it.
- **100% Transparency (Invisible)**: The cloud disappears completely—users sometimes mistakenly set this value and wonder why the cloud is no longer visible!

Enhancing Cloud Visualization with Contour Lines

For a more detailed representation, users can select the **Cloud Velocity** or **Cloud Pressure** options and **enable contour lines** (see Figure 2.30). This visualization technique is particularly effective in highlighting energy centers where velocity and pressure are at their highest.

By using these display features, users can gain deeper insights into the behavior and structure of the powder cloud, leading to a more comprehensive understanding of their simulation results.

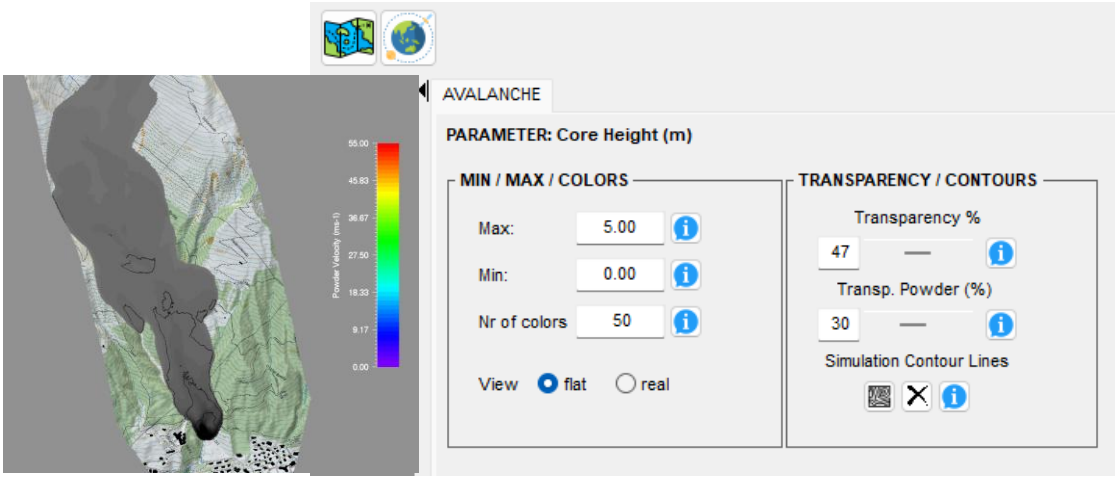


Figure 2.30: The Transp. Powder (%) slider is positioned in the upper right corner of the GUI, just below the Input Map and Orthophoto icons. This slider controls the transparency of the powder cloud, with 0% transparency rendering the cloud fully opaque. Additionally, users can overlay (velocity) contour lines on the cloud surface, providing a clear visualization of velocity gradients within the powder cloud.

2.14 Advanced Output Analysis in RAMMS::Extended

RAMMS::Extended employs a physics-based approach to simulate snow avalanche behavior. To compute the primary state variables that describe avalanche dynamics—namely **flow height, velocity, and impact pressure** of the **core and cloud** —the model also calculates a broad set of secondary variables. These include avalanche temperature (internal energy), random kinetic energy (also referred to as granular temperature), and water content, among others.

While the **Quick Access Keys** are designed for efficient visualization of the **primary variables and boundary conditions** (such as snow cover and forest data), they do not provide direct access to most secondary state variables.

However, expert users can explore these additional outputs—such as dynamic friction parameters and other internally computed quantities—through the **Results Tab** (Figure 2.31). This interface offers a comprehensive suite of tools for visualizing, plotting, and analyzing the full range of simulation data.

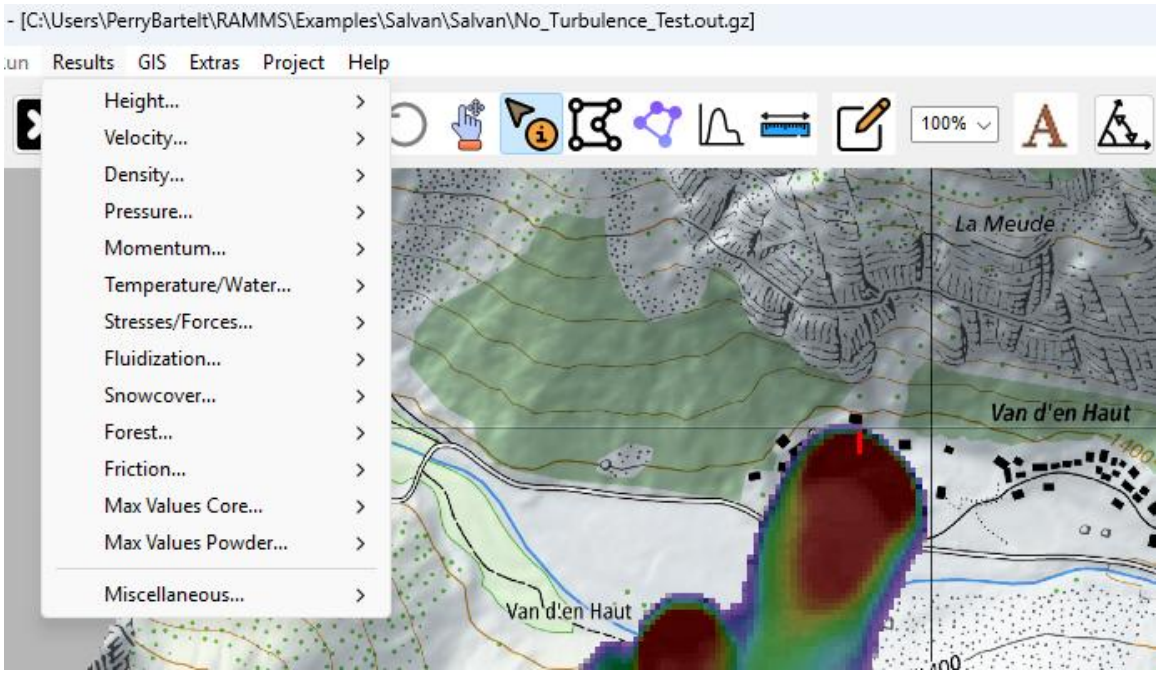


Figure 2.31: Via the Results Tab, the user can visualize many additional simulation results.

A complete list of the secondary variables available for inspection can be found in the tables below.

2.14.1.1 Flow and Deposition Heights

Heights	**Core Height**	Flow height of the avalanche core h_{Φ}
	Slope Angle vs. Deposition Plot	Provides a method to compare field measurements to simulation output.
	Powder Height	Flow height of the powder cloud h_{Π}
	Powder Deposition	The height of the powder cloud on the ground when settled to a density of 200kg/m^3

2.14.2 Velocities

Velocities	**Core Velocity**	Flow velocity of the avalanche core $\|\|\vec{u}_\Phi\|\|$
	Powder Velocity	Flow velocity of the powder cloud $\|\|\vec{u}_\Pi\|\|$

2.14.3 Densities

Densities	**Core Density**	Flow velocity of the avalanche core ρ_Φ
	Powder Density	Flow velocity of the powder cloud ρ_Π

2.14.4 Pressure

Pressures	**Empirical Pressure**	Empirical pressure formula $C_d = 2$ $$p = \frac{1}{2}\rho_\Phi C_d \|\vec{u}_\Phi\|^2$$
	Voellmy Pressure	p_Ξ $$= \rho_\Omega g \left[\sqrt{\left(\frac{q}{2}\right)^2 + H\left(\frac{p_\Lambda}{\rho_\Omega}\right)} - \frac{q}{2}\right]$$ with $$= \frac{p_\Lambda}{\rho_\Phi} - \left(\frac{H}{2}\right)\left[1 + \frac{\rho_\Phi}{\rho_\Omega}\right]$$
	Work-Energy Pressure	$$p_\Xi = \rho_\Phi \frac{h_\Phi}{h_\Xi}\left[1 - \frac{\rho_\Phi h_\Phi}{\rho_\Xi h_\Xi}\right]^{-1} V_\Phi^2$$

	Powder Pressure	$p_{\Pi} = \frac{1}{2}\rho_{\Pi}(V_{\Pi} + V_{\Pi}^{T})^2$ Laminar and turbulent parts

2.14.5 Momentum

Momentum	**Core Momentum**	Flow velocity of the avalanche core $h_{\Phi}\|\|\vec{u}_{\Phi}\|\|$
	Powder Momentum	Flow velocity of the powder cloud $h_{\Pi}\|\|\vec{u}_{\Pi}\|\|$

2.14.6 Temperature/Water

Temperature/Water	**Temperature**	Core temperature T_{Φ}
	Water Content	Meltwater in core m_{Φ}
	Meltwater Production	Meltwater production provides the total meltwater produced at a specific point on the avalanche track. Not a variable that moves with avalanche, fixed to track.

2.14.7 Stresses/Forces

Stresses/Forces	**Shear Stress**	S_{Φ}
	Normal Stress Core	N_{Φ}

	Shear Stress Powder	S_{Π}

2.14.8 Fluidization/Turbulence

Fluidization/Turbulence	**Core RKE**	$h_{\Phi}R_{\Phi}$
	Powder Turbulence	$h_{\Pi}R_{\Pi}$

2.14.9 Snowcover

Snowcover	**Erodible Layer**	Layer height over time (momentary height of layer) $h_{\Sigma}(t)$
	Layer Density	Density ϱ_{Ξ} Static, does not change with time (NO PLOT)
	Layer Temperature	Temperature T_{Σ} (NO PLOT)
	Surface Difference	Adds the core height of the avalanche to the momentary height of the erodible layer. Provides the changing height of the specific point

2.14.10 Forest

	Detrainment Height	Provides forest detrainment heights over time

Forest	**Stem Bending Stress**	Provides calculated bending stress over time at specific location in forest
	Max Bending Stress	Maximum bending stress at specific location in forest (NO PLOT)
	Tree Destruction	Tree destruction value over time. Time at which tree is destroyed

2.14.11 Friction Parameters

Friction	**Coulomb Friction**	Dynamic friction $\mu(t)$
	Turbulent Friction	Dynamic friction $\xi(t)$

2.14.12 Max Values Core

The **Display** tab, located on the right-hand side of the **RAMMS::Extended** graphical user interface (GUI), provides users with a streamlined way to visualize key maximum values derived from the simulation. By default, the interface allows for the display of four primary maximum variables for both the **core** and **cloud** components of the avalanche:

- Maximum flow height
- Maximum flow velocity
- Maximum impact pressure
- Minimum flow density

However, the menu also enables access to a broader set of maximum-value outputs for the **core**, offering deeper insight into avalanche behavior. These include:

- Maximum core pressure wall
- Maximum core pressure pylon
- Maximum core temperature
- Maximum core water content

- Maximum core meltwater content

Among these, the most informative and frequently utilized plots—particularly for expert analysis—are:

- **Maximum core temperature**
- **Maximum core water content**
- **Maximum core meltwater**

These variables provide critical insights into the thermodynamic and hydrologic processes within the avalanche core and are especially valuable when assessing melting, phase transitions, and water-driven mobility. (See Figure 2.32 and Figure 2.33)

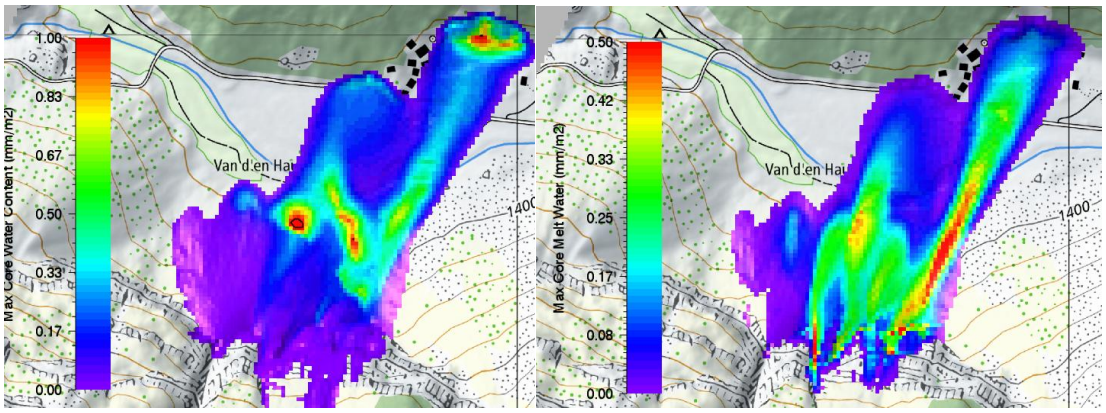


Figure 2.32: Right Panel: Max Water content in the avalanche. Meltwater is transported with the flow. Left Panel: Max Meltwater production. Values are less than 1mm/m², indicating a thin water film on the snow particles.

2.14.13 Max Values Cloud

The **Display** tab also supports visualization of key **maximum powder flow** values, giving users a quick overview of the powder cloud's dynamics. The four primary variables available for display are:

- Maximum powder cloud height
- Maximum powder cloud velocity
- Maximum powder cloud pressure
- Maximum powder cloud density

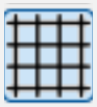
For more detailed analysis, the **Results** tab extends this functionality by offering additional powder-related maximum values, including:

- **Maximum powder cloud height** where **pressure exceeds 1 kPa**
- **Maximum powder cloud momentum**

These advanced variables are particularly useful when assessing the impact potential and dynamic behavior of the powder cloud. The pressure-filtered height, for example,

highlights zones of high-energy, high-impact flow, while the momentum output offers insight into the cloud's capacity to entrain and transport snow or interact with terrain and structures.

2.15 Show Simulation, Mesh, Lights and Colorbar

	**Show/Hide Simulation**	Toggles the visibility of simulation results in revealing the underlying map or orthophoto.
	**Show/Hide Colorbar**	Adds or removes the colorbar from the display for cleaner visual outputs.
	**Show/Hide Mesh**	Displays the computational mesh used in the simulation to assess DEM resolution and element quality.
	**Show/Hide Lights**	Displays a light source to create lighted surfaces with shadows.

RAMMS::Extended offers three intuitive visualization tools that enhance the interpretation of simulation outputs: **Show Simulation**, **Show Colorbar**, and **Show Mesh**.

- ### Show Simulation
  This toggle allows users to **quickly display or hide the simulation results**. It is particularly useful for revealing the underlying orthophoto or base map, making it easier to **identify terrain features, infrastructure, or other areas impacted by the avalanche**. While this tool provides a fast way to assess inundated zones, similar insights can also be obtained by adjusting the **transparency of the simulation layer**. As such, this feature is used occasionally and mostly for rapid visual inspection.

- ### Show Colorbar
  This option controls the **visibility of the colorbar** linked to the simulation data. It is especially helpful when creating clean, presentation-ready screenshots without a colorbar. Users often disable the colorbar to produce **uncluttered visualizations**, especially when the color scale is already known or explained elsewhere in the documentation.

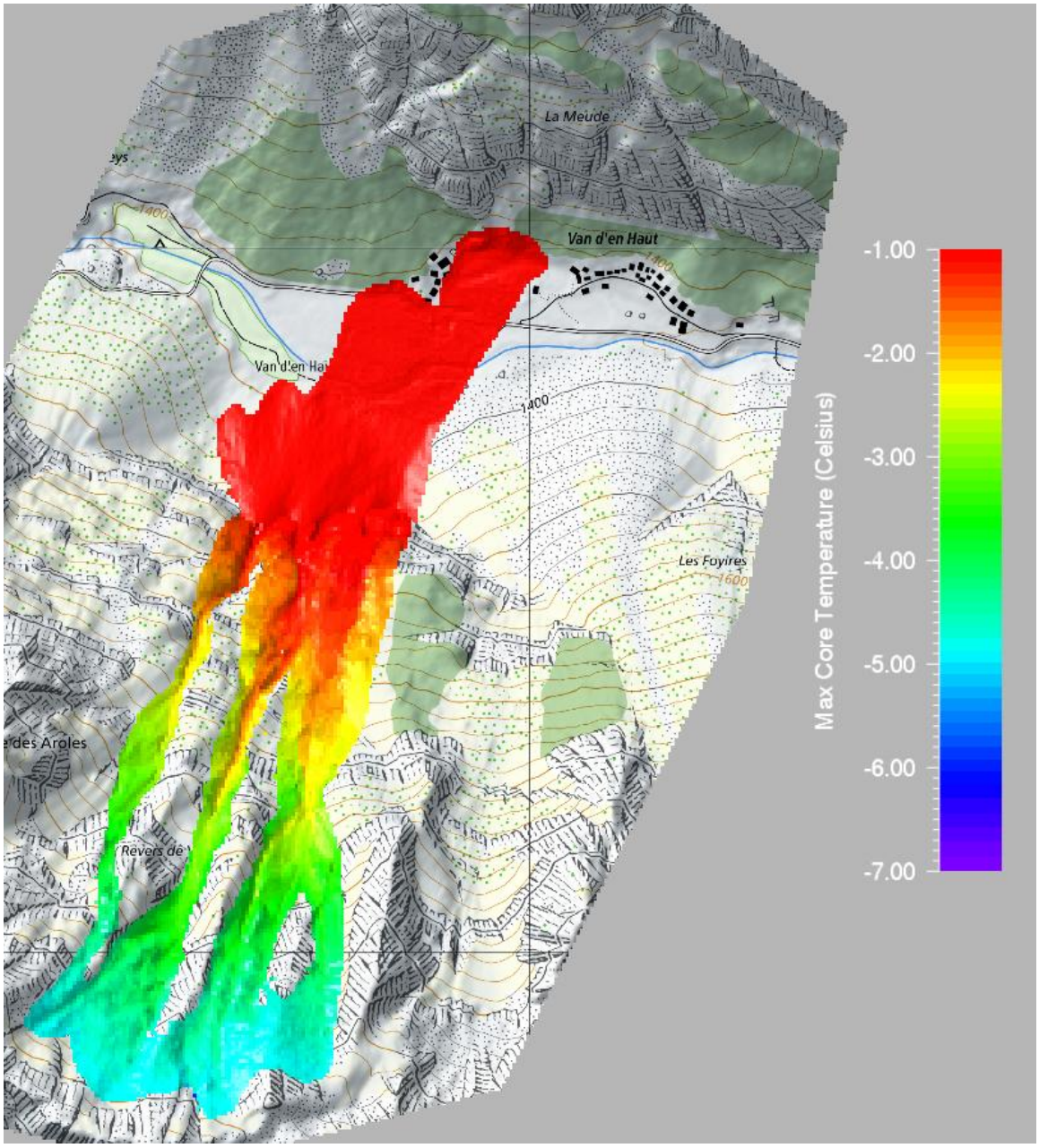


Figure 2.33: Maximum Core Temperature, as displayed in the Results menu, provides the most effective visualization of the avalanche's thermal evolution. In this example, the avalanche initiates at a temperature of –7 °C and progressively warms to approximately –1 °C during its descent.

- ## Show Mesh

  This feature displays the **computational mesh** used during the simulation, providing insight into the **underlying numerical grid** (Figure 2.34). It helps users assess the resolution of the Digital Elevation Model (DEM), identify element density,

and pinpoint potential problem areas that may lead to numerical instabilities. Mesh visualization is a **valuable diagnostic tool during model setup** and evaluation.

- ## Display Light Source

  This feature enables **a directional light source to illuminate the terrain**, creating **dynamic lighting and shadow effects across the surface**. It enhances the three-dimensional appearance of the digital elevation model (DEM) by emphasizing **terrain roughness**, **slope angles**, and **morphological details**. The interplay of light and shadow helps users better perceive subtle topographic features that may be difficult to distinguish in flat or uniform shading. This is particularly useful when analyzing flow behavior in steep terrain or complex topographies, as the lighting dramatically improves **spatial orientation and depth perception** during both 2D and 3D visualization.

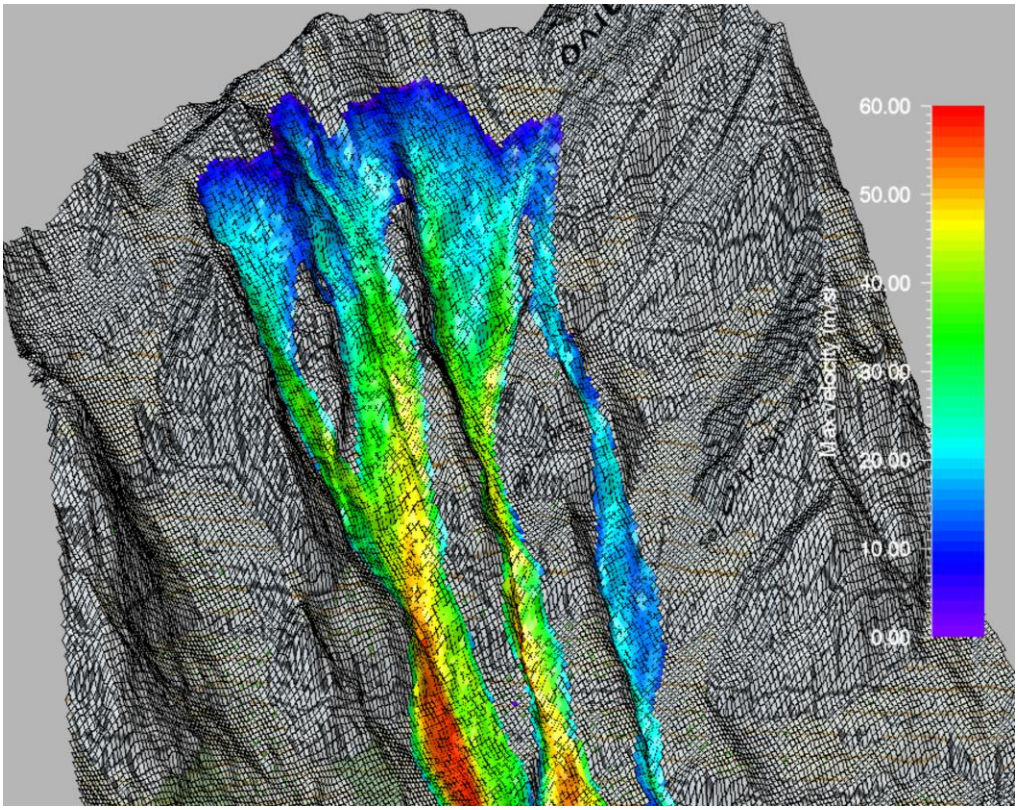


Figure 2.34: Display of simulation output and numerical grid enabled via Show Mesh.

2.16Profile and Point Info Plots

	**Create Line Profile**	Begins left click, left click ... right click process to define line profile
	**Select Info Point**	Selects info point to find max value or create time plot of simulation data

2.16.1 Creating Profile Line Plots

Step 1: Select the Desired Simulation Output
Begin by enabling the visualization of the specific simulation variable you want to analyze — such as **core flow heights**, **maximum core velocities**, or **maximum powder pressures** (Figure 2.35).

Step 2: Draw the Profile Line
Activate the **Create Profile Plot** tool, then define the profile by left clicking to place a sequence of vertices.

- You can start the line at any point: either from the top of the avalanche path (e.g., release area) down to the runout zone, or vice versa.
- The **first left-click** defines the beginning of the profile and marks the origin point $s = 0$, where s is the distance along the line.
- Continue placing the line using additional left-clicks.
- Conclude the profile by performing a **right-click**. The final segment ends at the **last left-click**; the right-click itself is only used to complete the action.

⚠️ **Important:** Once you start drawing the profile, do **not** zoom, pan, or click other interface elements until you complete the profile with a right-click — any interruptions may cancel the process.

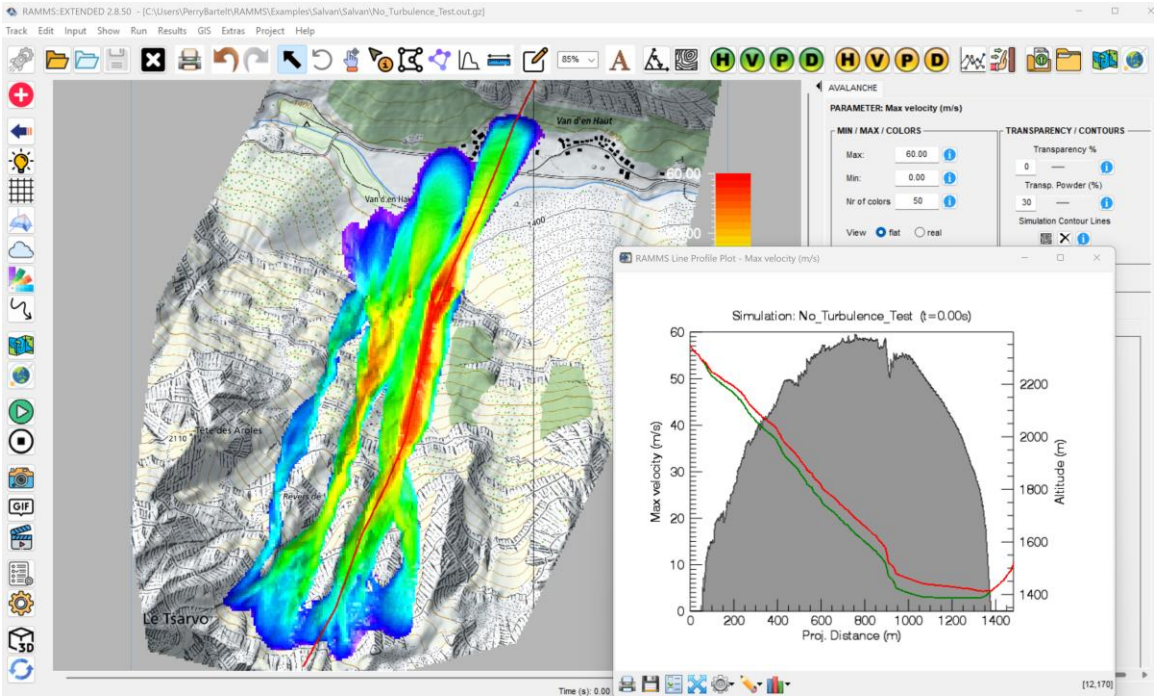


Figure 2.35: A longitudinal profile line has been created on a maximum velocity map. The inset shows the resulting profile plot: the green line represents the terrain elevation along the profile, the red line shows scaled maximum velocity values projected onto the elevation profile, and the gray plot displays the actual maximum velocity values corresponding directly to the y-axis.

Step 3: Analyze the Profile Plot
After finishing the line, a profile plot is automatically generated. This includes:

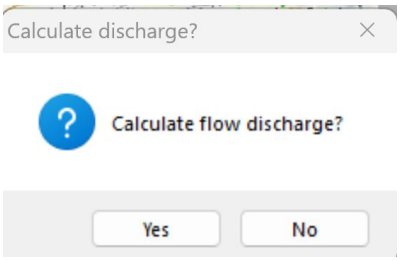
- The **elevation profile** of the terrain along the defined line.
- Two data representations:
    - A **terrain-following plot** showing values along the surface elevation (red line).
    - A **gray-colored plot** along the bottom axis, where y-axis values correspond precisely to the selected simulation variable.

The **lower-axis plot is preferred for accurate data interpretation**, while the surface plot helps visualize the spatial flow structure — such as velocity or flow depth along the avalanche body from front to tail.

Persistent Profile Line
Once defined, the **profile line remains visible** even if the user switches the displayed simulation data. For example, a profile initially created while viewing maximum core velocities can be instantly reused to generate new plots for maximum powder pressures, core heights, or other variables — enabling fast, multi-variable analysis along the same path.

Along-Path and Cross-Path Profiles
Profiles can be drawn:

- **Along the avalanche path** to study the evolution of flow dynamics from initiation to runout, or
- **Across the path** to create cross-sectional views of the avalanche body.



When the line profile has been completed (right mouse click), the user will be asked if the **Flow Discharge** should be calculated (Figure 2.36). Cross-path profiles are especially valuable for calculating **discharge across a section** — for example:

- Estimating inflow into a lake or reservoir,
- Assessing maximum impact pressures on roads, railway lines, or infrastructure that intersect the avalanche path (Figure 2.37).

If yes is selected the flow discharge will be calculated and displayed on the profile plot.

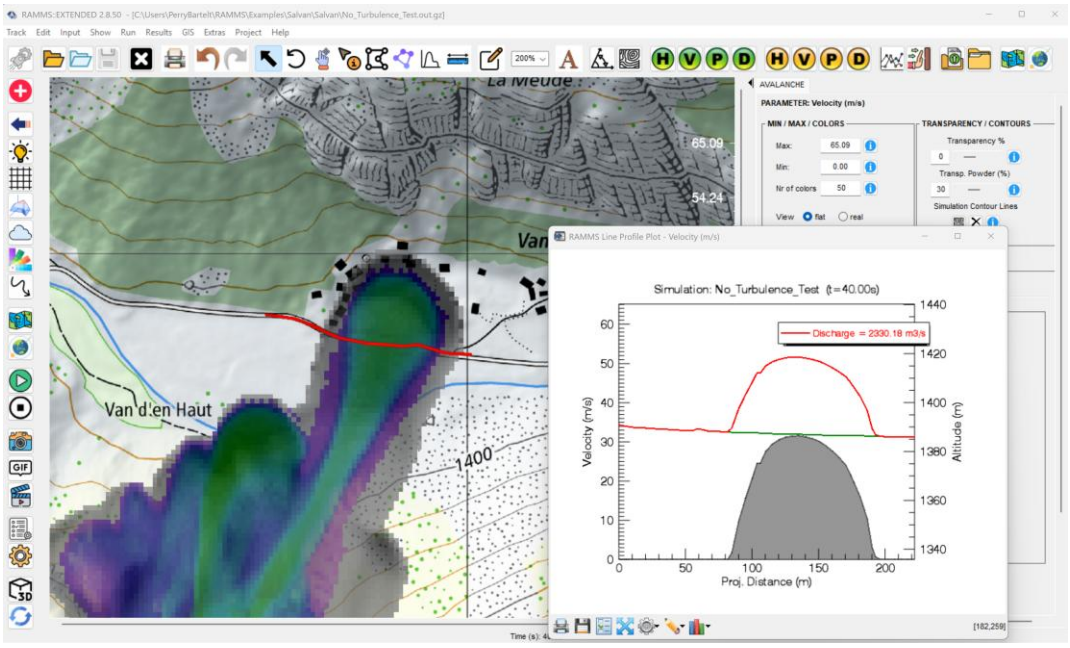


Figure 2.36: A profile has been drawn along a road to calculate the discharge of the core into the settlement and across the road. The inset shows the resulting profile plot when the avalanche crosses the road. The flow discharge is calculated. The dump-step is indicated in the profile plot.

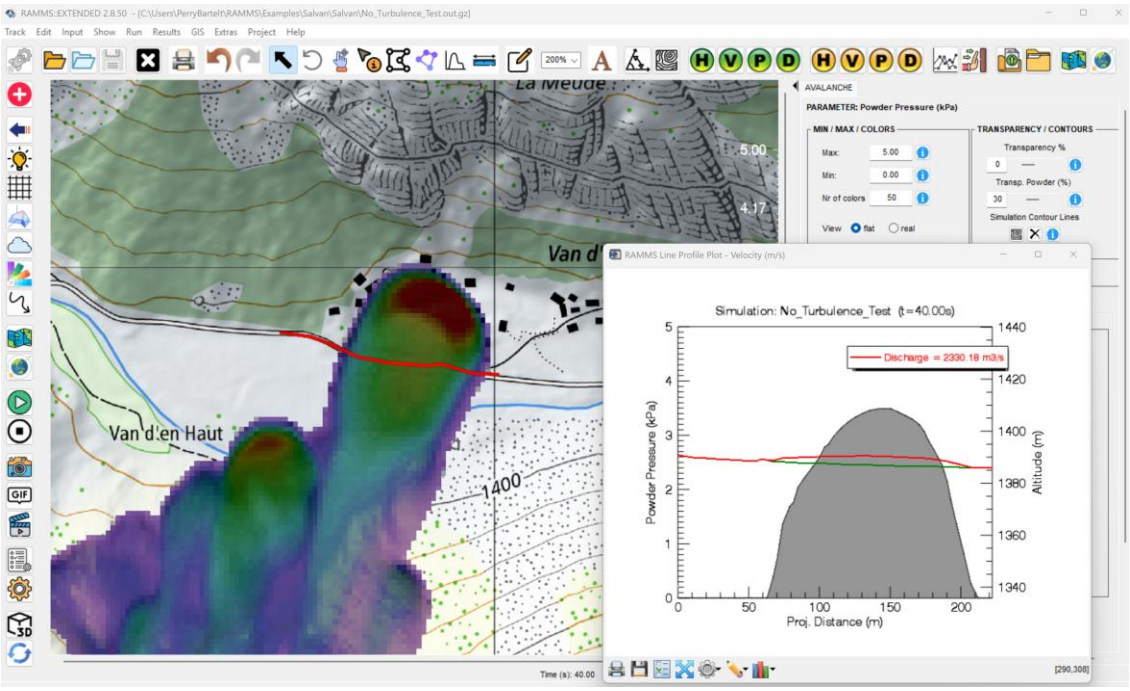


Figure 2.37: The same profile is used to calculate the powder pressure along the road. The discharge option can only be used when time-step data is depicted. The discharge will not be calculated when displaying maximum values, even if the user answered the calculate discharge question with yes.

2.16.2 Managing Profile Line Visibility and Appearance

When creating profile plots, it's common for users to want to remove the profile line from the visualization — especially when preparing clean screenshots or presentation material. Each time a profile is created, a shapefile named **profile.shp** is automatically generated and listed under the **Files Tab** (see Figure 2.38).

By default, the **blue checkbox** next to profile.shp is **enabled**, indicating that the profile is currently visible in the viewer. To hide the profile line, simply **click the checkbox** to uncheck it — this will immediately remove the profile from the display without deleting the data.

The **Files Tab** also allows users to customize the appearance of the profile line:

- **Right-click** on profile.shp to open a context menu.
- From here, you can adjust both the **color** and **line width** of the profile.
- By default, the profile line is displayed in **red** with a **4-point line width**.

This flexibility ensures that profile lines can be styled for visual clarity or hidden entirely when they are no longer needed in the active view.

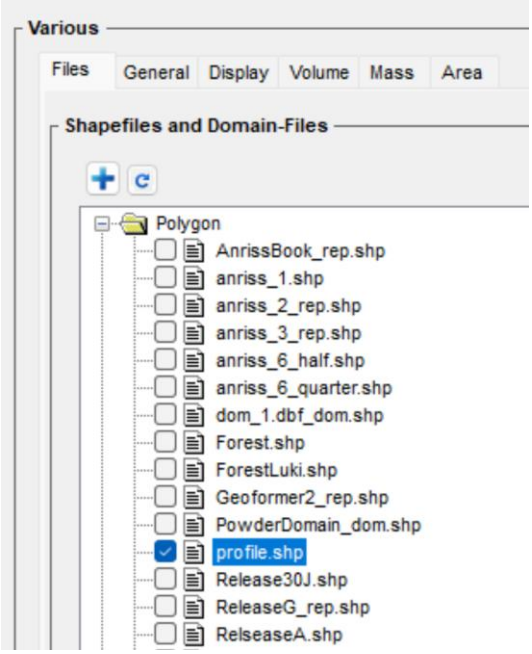


Figure 2.38: The profile.shp file in the Files Tab is used to manage the line profile—either remove from the visualization or to change color or width.

2.16.2.1 Saving and Reusing Profile Lines Across Simulations

In many cases, users may wish to reuse the same profile line across multiple simulations — for example, to compare results under varying flow parameters or boundary conditions. Reusing a consistent profile allows simulation results to be superimposed and analyzed along a shared reference line, making comparative plots more effective and meaningful.

To save a defined profile for future use (Figure 2.39):

1. Navigate to **Extras → Profile → Save Line Profile Points**.
2. Enter a name for the profile when prompted.

After saving, the new profile will appear in the **Files Tab**. If it doesn't appear immediately, **refresh the display**. To visualize the saved profile in the simulation view, simply **check the enable box** next to the profile name.

This feature provides a powerful way to ensure consistency and efficiency when conducting multi-scenario analyses using **RAMMS::Extended**.

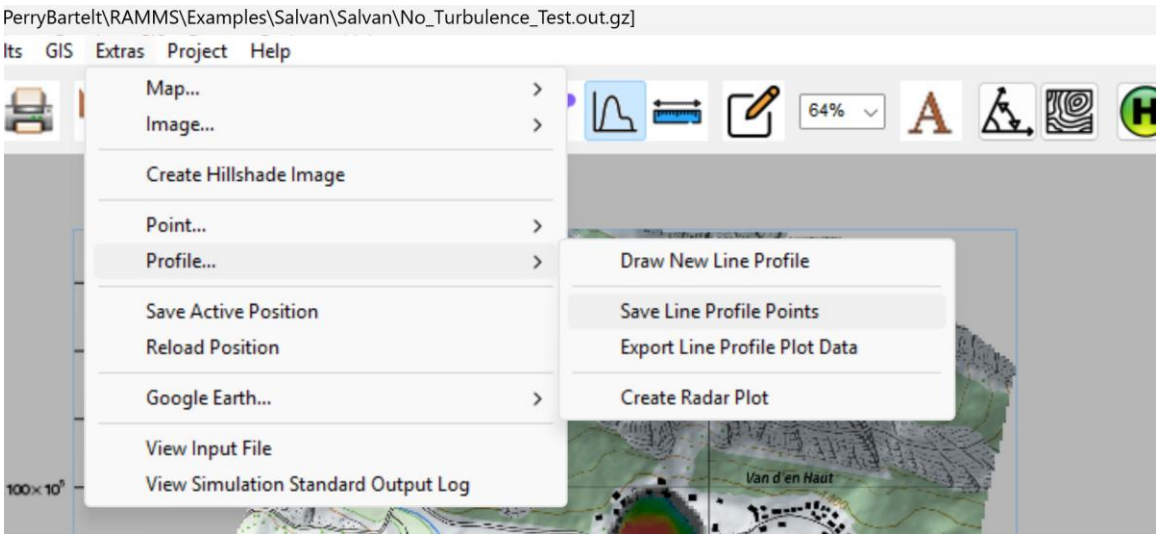


Figure 2.39: Saving a line profile using the Extras → Profile → Save Line Profile Points.

2.16.2.2 Loading a Saved Profile into a Simulation

To load a previously saved profile line into a simulation, users have two convenient options:

1. **Using the Profile Line Tool**
   Click the **Profile Line** icon to activate the tool. Instead of drawing a new profile using the left and right mouse buttons, simply **press the middle mouse button** (scroll wheel). This action opens a prompt allowing you to **select and load a saved profile** from the available files.
2. **Via the Files Tab**
   Alternatively, you can load a saved profile by navigating to the **Files Tab** and **activating the profile name** (e.g., MyProfile.shp) by checking the corresponding box. This will immediately display the profile in the viewer.

2.16.3 Point Information Tool

The **Plot Point Info** tool allows users to extract detailed simulation data at a specific location within the calculation domain (Figure 2.40). This tool works with both **maximum value outputs** and **time-dependent data**, making it a versatile option for localized analysis.

When the **Plot Point Info** icon is selected, the user can click directly on a point within the simulation area. Depending on the type of data currently visualized:

- A **data window** will appear showing the **maximum value** at that location (e.g., maximum flow height, pressure, or velocity), **or**
- A **time-series plot** will be displayed showing how the selected variable evolves at that point over time (see Figure 2.40).

The plotted data can include:

- **Core and powder flow heights**
- **Velocities**
- **Dynamic and static pressures**
- **Densities** — if density data are selected, the plot will include both the **density** and the corresponding **core or powder height**, depending on the type chosen (Figure 2.42).
- **Any output parameter** in the **Results menu**

2.16.3.1 Selecting a Location

Most commonly, the user simply **clicks a point** in the area of interest using the mouse. However, for precision analysis or comparative studies, it is also possible to **manually enter exact X-Y coordinates** of the desired location (Figure 2.41). This is particularly useful when consistent point tracking across simulations is required. Note: manual input assumes a solid understanding of the DEM's coordinate system, which can be obtained using a GIS application or directly within RAMMS.

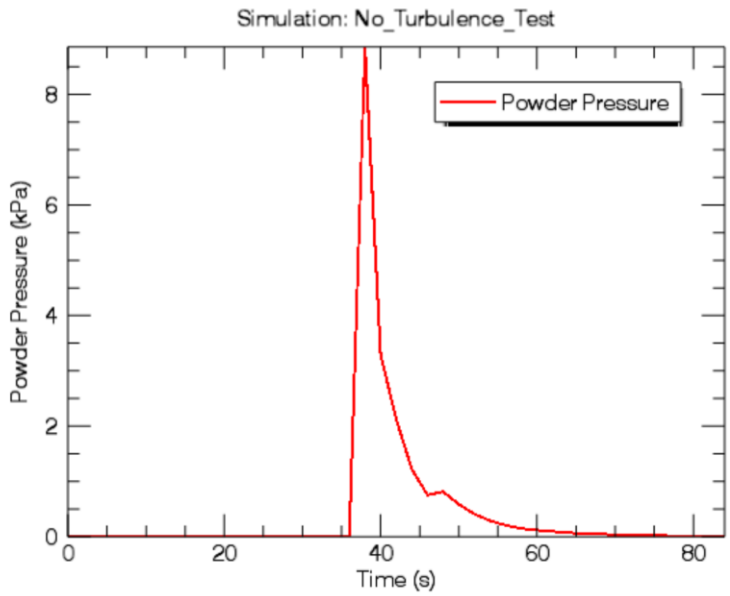


Figure 2.40: A line plot of a time-dependent value, in this case powder pressure.

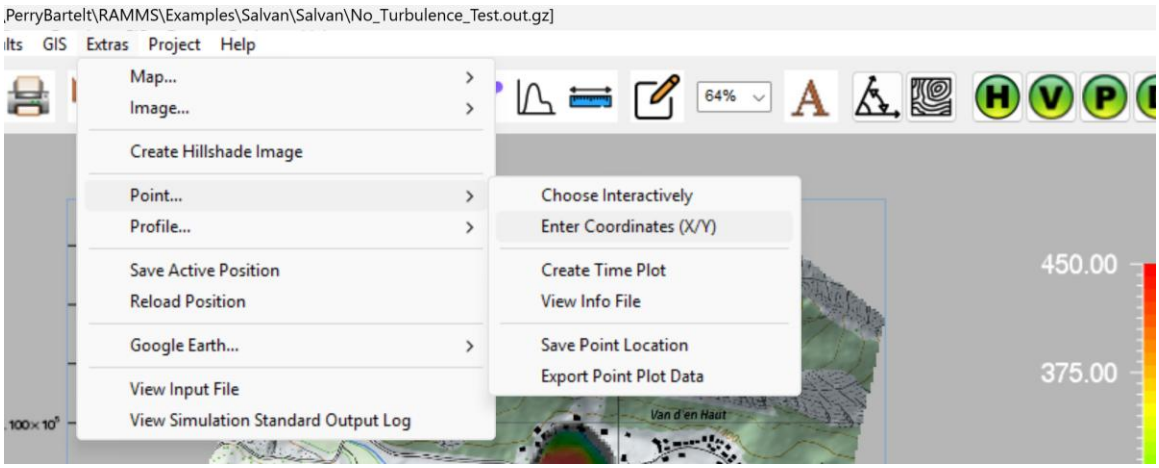


Figure 2.41: Defining a location using Extras → Point → Enter Coordinates (X/Y). Other options are View Info File, Save Point Location and Export Point Plot Data.

2.16.3.2 Accessing Detailed Point Data: The Info File

In addition to interactive plots, **RAMMS::Extended** provides a more detailed numerical summary at any point in the simulation domain through the **Info File**. To access it, navigate to: Extras **Point... → View Info File**

This feature allows users to retrieve a comprehensive snapshot of simulation data at a selected location. The **Info File** (see Figure 2.43) includes the following key details:

- The **(X, Y, Z)** coordinates of the selected point
- **Maximum core height**
- **Maximum core velocity**
- **Maximum core pressure**
- **Maximum powder pressure**

What sets the Info File apart is that it not only records the **maximum values** but also includes the **exact time** at which each of these values occurs. This is critical because different flow variables — such as core height, pressure, and velocity — typically reach their peak values at **different times** during the simulation. For example, the moment of maximum core height may not coincide with the moment of peak impact pressure.

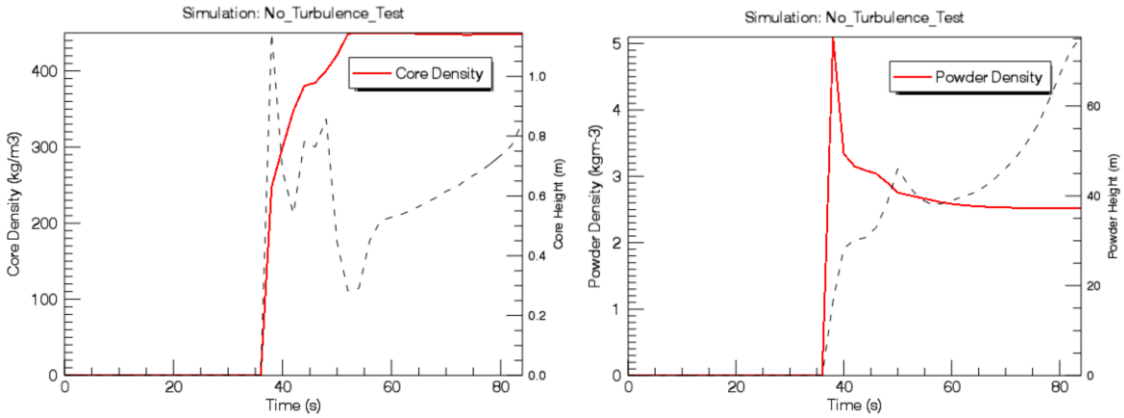


Figure 2.42: The density plots for the core and the cloud contain both the flow height and the flow density.

This temporal insight is particularly important when interpreting impact pressures, as **maximum impact does not necessarily occur when flow depth is at its highest**. The Info File, therefore, provides essential data for accurately analyzing the sequence and intensity of dynamic loading at specific locations.

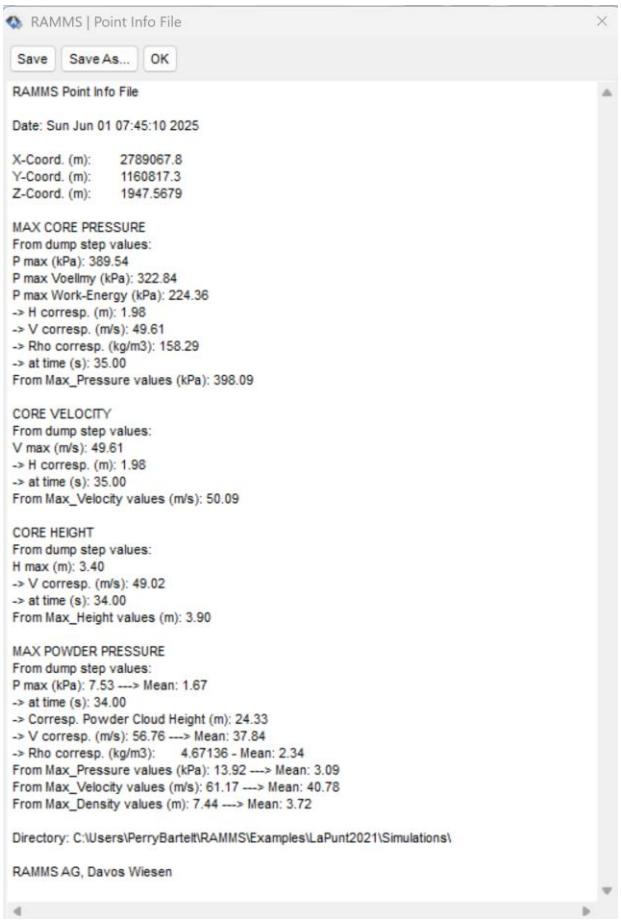


Figure 2.43: The point info file contains flow information (pressure, height, velocity, powder) about a specific point selected by the user.

2.16.3.3 Saving and Reusing Point Locations

For comparative studies or documentation purposes, it is often necessary to consistently analyze simulation data at a **specific geographic location** — such as the site of a measurement station, a damaged building, or a critical infrastructure point. **RAMMS::Extended** allows users to save such locations for reuse across multiple simulations.

Saving a Point Location
To save a point of interest:

1. Define the location, typically by entering **exact (X, Y) coordinates**, especially when the point corresponds to a real-world reference.
2. Go to **Extras → Point... → Save Point Location**.
3. Enter a file name. The point will be saved as a **.txt file**.

Unlike profile lines, the point locations saved **do not appear in the Files Tab**.

Loading a Saved Point
To reintroduce a saved point into a simulation:

1. Click the **Point Info** icon to activate the tool.
2. Instead of selecting a point with the mouse, press the **middle mouse button (scroll wheel)**.
3. A file dialog will open, prompting you to load a saved **point .txt file**.

Once loaded, the selected point will behave like a manually chosen point — enabling access to time-series plots or maximum values at that location, depending on the current simulation output being displayed.

This feature ensures that consistent spatial references can be maintained across scenarios, which is essential when comparing different flow conditions, evaluating mitigation designs, or validating model results against field data.

2.16.3.4 Visualization of the Selected Point

- In **2D view**, a **red vertical line** appears at the selected point.
- In **3D view**, the point is marked by a **downward-pointing arrow**, making it especially useful for presentations and video exports — offering a clear visual reference as the avalanche approaches or flows past the selected location (Figure 2.44).

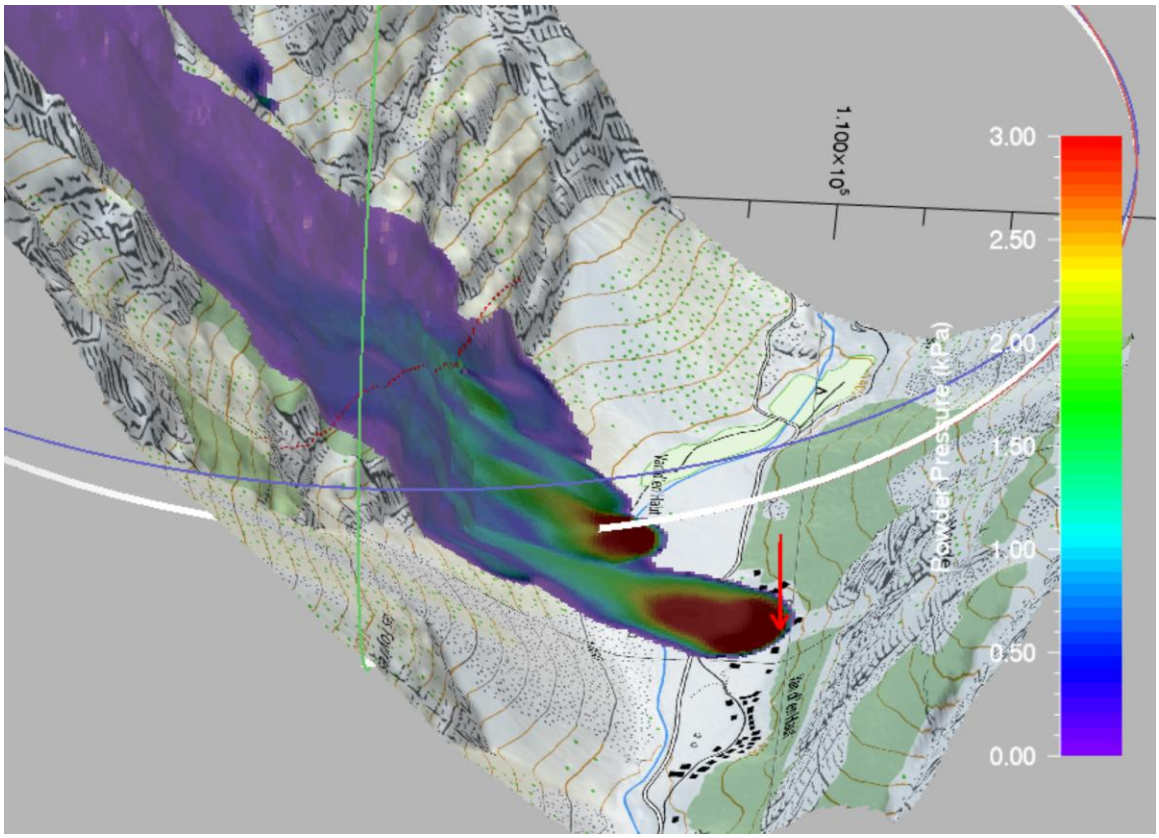


Figure 2.44: Visualizing a point in three dimensions. The point becomes an arrow.

2.17 Images, Animations and Making Videos

2.17.1 Graphic Images and Gif Animations

	**Create PNG Image**	Creates and stores a *.png file of the current image in the interface
	**Create Gif Animation**	Creates and stores a GIF animation file with name *.gif.

The RAMMS interface offers two powerful visualization tools: the ability to generate **static images** and **GIF animations**. These features allow users to capture and share simulation results with clarity and precision.

To access these options, use the Image or GIF workflow icons located in the left vertical toolbar, just below the Run and Stop buttons on the graphical user interface (GUI).

- **Creating an Image**: Selecting the Image icon prompts the user to specify the file name and storage location for the exported .png image.

- **Generating a GIF Animation**: Similarly, clicking the GIF icon opens a dialog where users can define the name and save location of the animation file.

Before exporting an image or GIF, it is crucial to **configure the visualization settings** to reflect the desired output. This includes:

- Switching between 2D and 3D visualization modes using the 2D/3D icon

- Adjusting the background color for optimal contrast

- Scaling numerical results to enhance readability

By fine-tuning these settings beforehand, users can ensure that their exported images and animations effectively communicate key aspects of the simulation.

The static image and GIF animations options can also be selected using the **Export** feature in the pull-down **Track** menu (see Figure 2.45).

2.17.2 Exporting Results as a Video (FFmpeg format)

To create a **high-quality video (FFmpeg) of your simulation results**, the most effective method is to use the **Export** option located in the **Track** pull-down menu. Select the Video (FFmpeg) option (Figure 2.45).

Before selecting this option, ensure that the interface is set to display the desired results.

A common workflow for exporting videos involves configuring the visualization mode to a **three-dimensional view**, adjusting core height scaling, and positioning the time slider at t = 0. Additionally, users must decide what elements should be animated—whether to include all scaling parameters, display the powder cloud, or omit it entirely. Fine-tuning background colors and the results scale are also an essential step in achieving optimal clarity.

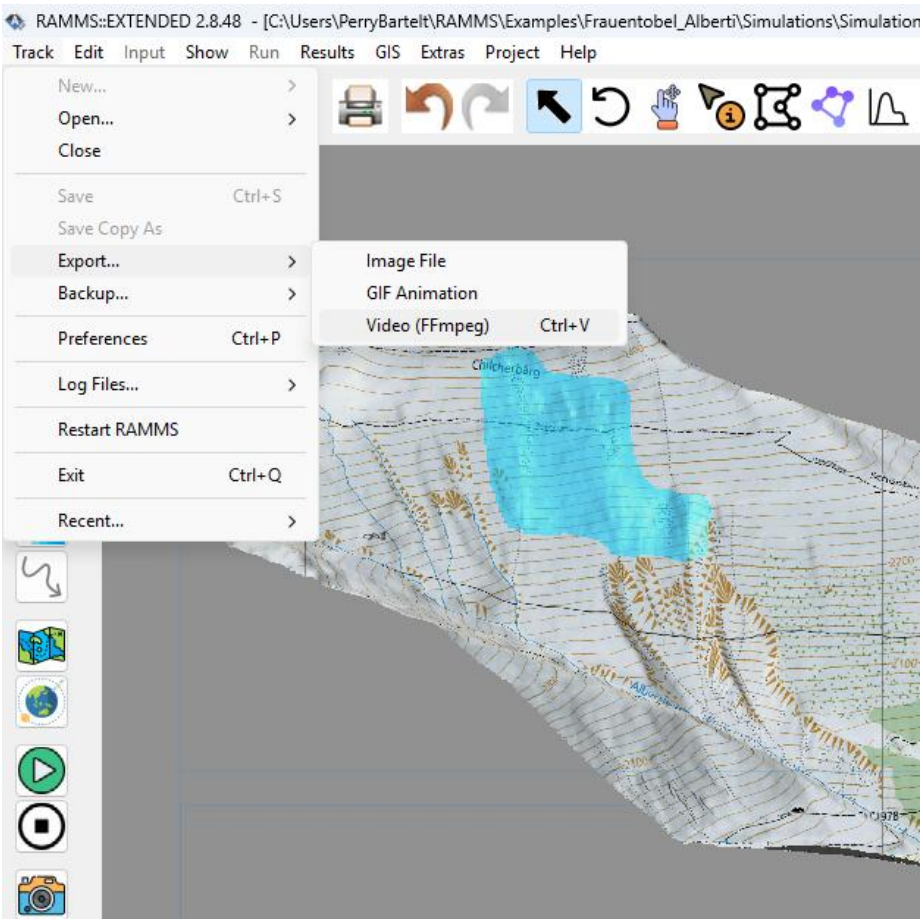


Figure 2.45: Start making a video using the Export and Video option in the Track pull-down menu. Static image files, and GIF animations can also be created with this option.

Upon selecting the **Export/Video** option, a dialog menu will prompt the user to specify which dump steps should be used to generate the video (Figure 2.46Figure 2.46). For best results, it is recommended to output simulation data at 1-second or 2-second intervals, as this provides the flexibility needed to create smooth and realistic animations.

The **Frame rate** and **Quality** settings allow users to control the speed and resolution of the video. While higher quality ensures finer details, it also increases file size. Once the desired settings are configured, clicking OK will prompt the user to define the **start** and **end** times (steps) of the video. When creating close-up views of specific locations, it is advisable to set the start time to the moment when the simulated avalanche enters the visualization. Selecting the **final time** step can also be useful, as this allows the powder cloud to dissipate, revealing a clear representation of the core simulation results.

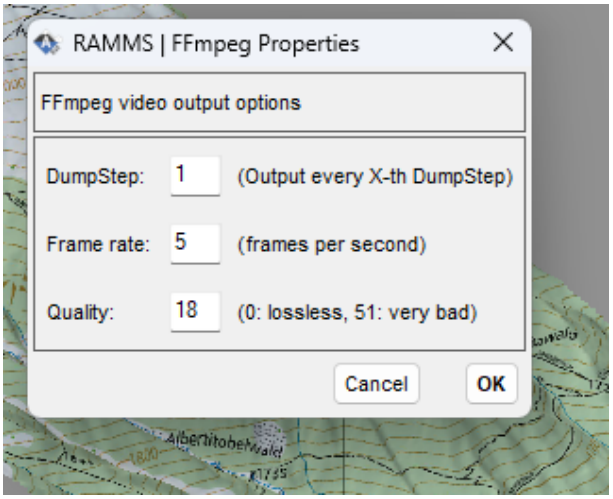


Figure 2.46: Menu to set the video options. The default values create videos with reasonable quality and size.

2.18Exporting Numerical Results

2.18.1 Quick Export as Raster Data

	**Export Result as Raster**	Exports spatial data (e.g max velocity, max pressure) to file

Exporting simulation results to GeoTIFF- or ASCII- (.asc) files is a common and essential step for users of **RAMMS::Extended**. This functionality enables seamless integration with **Geographic Information Systems (GIS),** allowing users to visualize and analyze simulation outputs—such as maximum velocity, flow height, or pressure—for inclusion in professional reports or further scientific research.

Applications and Benefits

- **Professional Reporting**: Engineering offices frequently use exported data to generate high-quality visualizations, often customized with client-specific color schemes, for inclusion in technical documentation and project reports.

- **Scientific Research:** Researchers benefit from the ability to perform custom analyses on the exported datasets using specialized tools outside of RAMMS.

- **GIS Integration:** The GeoTIFF - and ASCII- grid format is widely supported by GIS software, making it easy to overlay simulation results on real-world topographic maps and infrastructure layouts.

To export Simulation Data:

1. **Display the Desired Result:** Before exporting, users must first ensure that the relevant simulation layer (e.g., maximum velocity, flow height) is visible in the main viewer.

2. **Initiate Export:** Clicking the *Export to Raster* icon opens a dialog window prompting the user to specify a file name.

3. **Default File Naming:** By default, **RAMMS::Extended** will automatically generate a file name based on the simulation name and the selected data layer.

4. **Output Format:** The default file format is **GeoTIFF**. An alternative is the **ASCII** format: The exported file contains two-dimensional gridded data in standard ASCII format. A sample header format is illustrated in Figure 2.47.

```
ncols 417
nrows 346
xllcorner 2788004.00
yllcorner 1159330.00
cellsize 6.0000000
NODATA_value -9999
     -9999.000     -9999.000     -9999.000     -9999.000     -9999.000     -9999.000     -9999.000     -9999.000     -9999.000
.000     -9999.000     -9999.000     -9999.000     -9999.000     -9999.000     -9999.000     -9999.000     -9999.000     -999
9999.000     -9999.000     -9999.000     -9999.000     -9999.000     -9999.000     -9999.000     -9999.000     -9999.000
-9999.000     -9999.000     -9999.000     -9999.000     -9999.000     -9999.000     -9999.000     -9999.000     -9999.000
-9999.000     -9999.000     -9999.000     -9999.000     -9999.000     -9999.000     -9999.000     -9999.000     -9999.000
-9999.000     -9999.000     -9999.000     -9999.000     -9999.000     -9999.000     -9999.000     -9999.000     -9999.000
```

Figure 2.47: Header of ASCII grid file. Note the definition of NODATA=-9999

Notes

- **NODATA Values:** The specific numeric value used to represent *NODATA* is clearly indicated in the header of the output .asc file.

- The user will be informed when the file has been successfully exported.

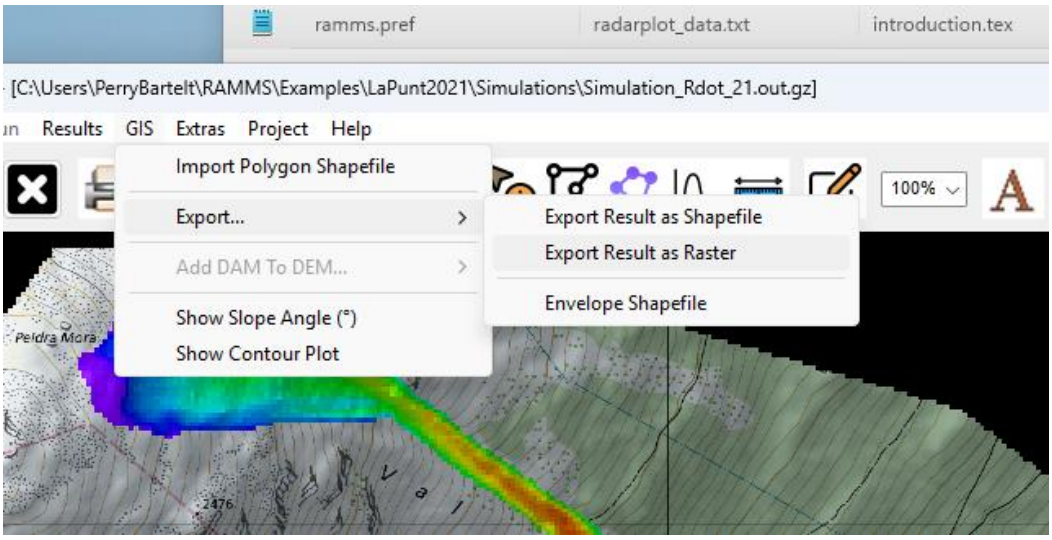


Figure 2.48: Users can navigate to *GIS → Export → Export Result as Raster* to initiate the export of simulation data.

In addition to the quick-access export icon, **RAMMS::Extended** offers an alternative method for exporting simulation results through the main menu interface. Users can navigate to **GIS → Export → Export Result as Raster** to initiate the export process (Figure 2.48).

Functionally, this menu option performs the same operation as the quick-export button. Upon selection, the user is prompted to enter a file name for the **GeoTIFF** or **ASCII** raster file that will store the simulation output. This provides a more structured workflow for users who prefer navigating via the menu bar or when working through documented export procedures.

2.18.2 Exporting Line Profile Data as ASCII text

To export simulation data along a defined line profile, the user must follow a specific sequence of steps. First, a **line profile** must be created within the simulation domain. Next, the user should **visualize the desired result** (e.g., velocity, flow height, pressure) along this line. Only after the line profile data has been successfully plotted on the screen can the export function be accessed.

To export the data:

1. Navigate to **Extras → Profile → Export Line Profile Plot Data** (Figure 2.49).
2. A dialog will appear, prompting the user to manually enter a file name for the export.

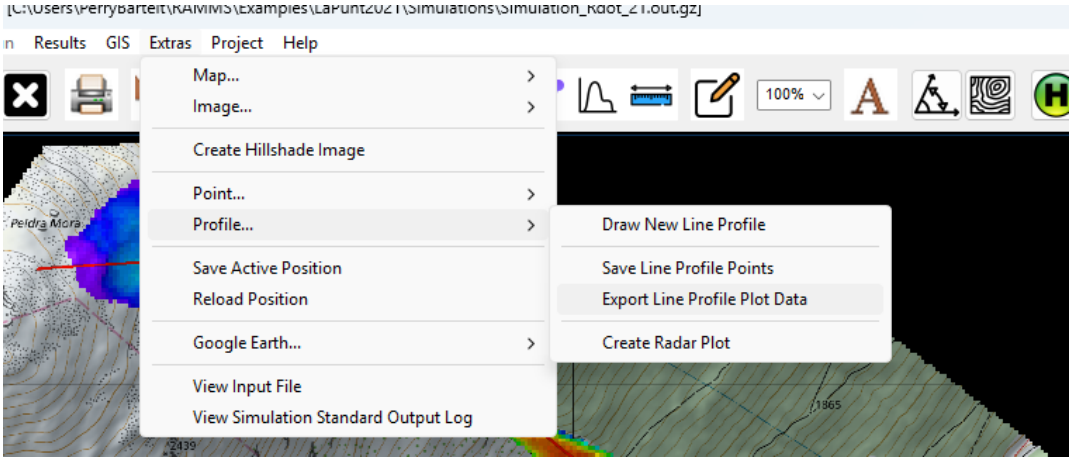


Figure 2.49: To export line profile data select: Extras → Profile → Export Line Profile Plot Data.

Unlike the grid-based raster export, which automatically suggests a default file name based on the simulation and data layer, the **line profile export requires the user to provide a custom name**. The data is saved as a plain text file with a .txt extension. The format of the *.txt file is shown in Figure 2.50.

```
RAMMS Profile Data

1st column: Proj. distance (m)
2nd column: Altitude (m a.s.l.)
3rd column: Max velocity (m/s)
4rd column: Mu min ()

     0.000000      2562.53     0.000000     0.000000
      5.00976      2561.00     0.000000     0.000000
      11.0215      2557.93     0.000000     0.000000
      17.0332      2554.65     0.000000     0.000000
      23.0449      2551.83     0.000000     0.000000
```

Figure 2.50: Format of Profile Line export file.

This feature is particularly useful for analyzing cross-sectional behavior along specific flow paths, enabling detailed post-processing in external tools such as Excel, MATLAB, or Python.

2.18.3 Exporting Point Plot Data as ASCII text

To export **point-specific simulation data**, the following steps are required:

1. **Select a Point of Interest:** Begin by activating the **Info Point tool** and clicking on a specific location within the simulation domain.

2. **Display Time-Varying Data:** Ensure that a **dynamic result** is selected—such as **core height vs. time** or **powder pressure vs. time**. It is important to note that

maximum values (e.g., max velocity or max pressure) cannot be exported in this format, as they are not time dependent.

3. **Visualize the Time Plot:** Once the time-dependent data for the selected point is displayed as a graph on-screen, proceed to export.

4. **Initiate Export: Navigate to Extras → Point... → Export Point Plot Data**. A dialog will appear asking the user to create or select a file name—ideally ending with a .txt extension.

The exported text file begins with a header (Figure 2.51), which contains:

- The **coordinates of the selected point.**
- The **units and descriptions** of the two data columns:
    - The first column typically represents time (in seconds).
    - The second column contains the corresponding simulation values (e.g., core height in meters, pressure in Pascals, etc.).

```
RAMMS Time Plot Data

Point coordinates (m):        2789378.3       1160606.4       1747.0799
1st column: Time (s)
2nd column: Core Height (m)

     0.000000     0.000000
      1.00000     0.000000
      2.00000     0.000000
```

Figure 2.51: Format of Point Plot export file.

2.19 Arrows

	**Draw Velocity Arrows**	Overlays velocity vectors (arrows) on colored spatial simulation results.

While velocity visualizations in **RAMMS::Extended** effectively display the magnitude or maximum value of the avalanche's velocity—whether for the core or the powder cloud—they do not inherently convey the **direction of flow**. Yet, understanding the flow direction at specific locations is often critical for interpretation and hazard assessment.

To address this, **RAMMS::Extended** offers the ability to overlay **velocity arrows** on top of simulation results (Figure 2.52). These arrows, superimposed on the velocity plot, provide

directional vectors that complement the color-coded velocity magnitudes, enabling users to simultaneously assess both speed and flow direction.

2.19.1.1Activating Velocity Arrows

By default, arrow overlays are displayed **for the avalanche core**, which is the most common use case. These overlays can be toggled on and off directly from the GUI.

Arrows	**GUI PREFERENCES FILE**
ARROW_MODE = 0 **Core** ARROW_MODE= 1 **Cloud**	`ARROW_MODE  0`

These ARROW preferences can be changed in the PREFERENCES file to optimize the display of velocity arrows	`ARROW_COLOR green`
	`ARROW_FACTOR 1`
	`ARROW_HEADSIZE 2`
	`ARROW_LENGTHSCALE 4`
	`ARROW_THICK 2`

Table 2.2: Visualization of arrows can be optimized by changing these options in the RAMMS::Extended GUI PREFERENCES FILE.

2.19.2 Arrow Scaling and Visibility

Properly **scaling the velocity arrows can be challenging**. If arrows appear too dense or visually cluttered, especially when zoomed out—they may resemble a black, tangled mass. **To improve visibility** and **readability**:

- **Zoom in** on a specific region of interest. This naturally enlarges the arrows, making their direction more discernible.
- Alternatively, adjust the **arrow scaling parameters** in the Preferences File to fine-tune their size across the entire visualization. (See Table 2.2)

Users who wish to visualize the **powder cloud** velocity vectors must modify the application settings. This is done by editing the **Preferences File**:

1. Click the **Edit GUI Preferences** icon located in the vertical toolbar on the left-hand side of the GUI.
2. Locate the setting labeled ARROW MODE.
3. Change its value from 0 (core) to 1 (powder cloud).
4. Save the changes and **restart the GUI** for the new settings to take effect.

By combining color-coded velocity plots with directional vectors, users gain a richer and more informative view of avalanche dynamics—allowing for more accurate interpretation of simulation outcomes.

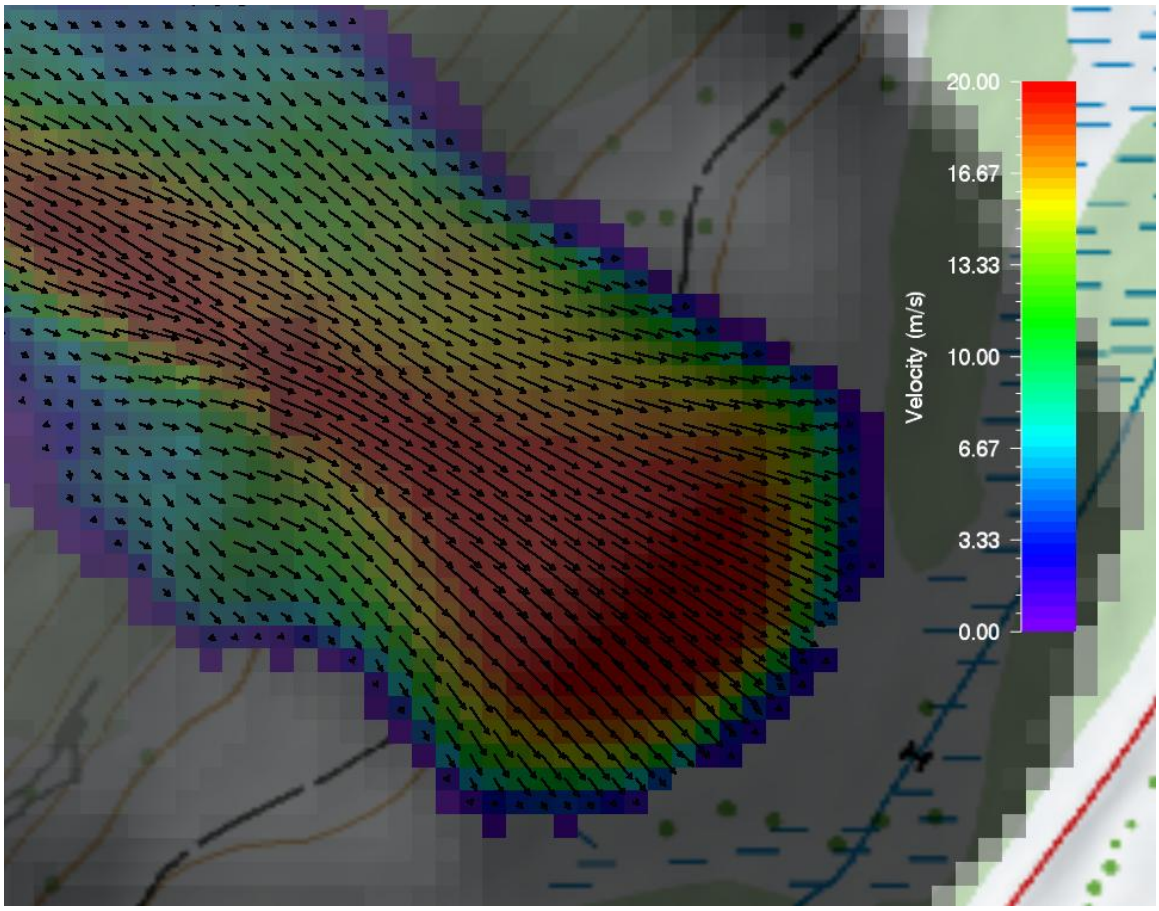


Figure 2.52: Visualization of avalanche core velocity combining color-coded grid cells with overlaid velocity vectors. The color grid represents the magnitude of the core flow velocity, while the superimposed arrows indicate the corresponding flow direction, providing a comprehensive view of avalanche dynamics.

2.20 Radar Plots: Leading Edge Velocity

One of the most common **misconceptions in avalanche dynamics** is the interpretation of "**velocity**." While the term may seem straightforward, it can refer to different measurements depending on the context.

2.20.1 Core Velocity vs. Leading Edge Velocity

In numerical simulations, we typically define **velocity** as the core velocity $(\vec{u}_{\Phi})$ or cloud velocity $(\vec{u}_{\Pi})$ — values associated with a fixed computational cell in an **Eulerian reference frame**. This velocity represents the internal motion of the avalanche at a specific point as it passes through the computational grid over time. In **RAMMS::Extended** internal velocities

are depicted when selecting the **visualization of maximum velocity** or making **point** or **profile plots.** It is important to note that avalanche internal velocities can be measured directly in the field using instruments such as light gauges mounted on masts, such as those at Vallée de la Sionne (the signals are rarely easy to interpret).

However, when discussing avalanche motion, we often refer to the **leading-edge velocity** — the speed at which the front of the avalanche propagates downslope. Unlike core velocity, the leading-edge velocity is a **Lagrangian measurement**, meaning it tracks the movement of a specific feature of the avalanche rather than the flow at a fixed point.

Several techniques exist for determining **leading edge velocity**:

- **Camera Arrays & Video Analysis**: By positioning fixed cameras along a path, the motion of the avalanche front can be tracked by georeferencing the front location at known time intervals. Usually, however, the velocity is simply estimated using video recordings taking advantage of prominent terrain features that can be easily located on maps.

- **Radar Systems:** Radars operate by sending and receiving signals reflected by the avalanche core and/or powder cloud. By knowing the difference between the departure and arrival time it is also possible to track the leading edge of the avalanche.

2.20.2 Visualizing Leading Edge Motion

In **RAMMS::Extended**, **leading edge velocity** can be determined using the **Create Radar Plot** option, allowing direct comparison with radar measurements. To generate these plots, users must utilize the **Profile Plot** option and select "**Create Radar Plot**" from the **Extras** pull-down menu (see Figure 2.53).

The process involves three main steps: **selecting the profile**, **visualizing core heights**, and **generating the radar plot**.

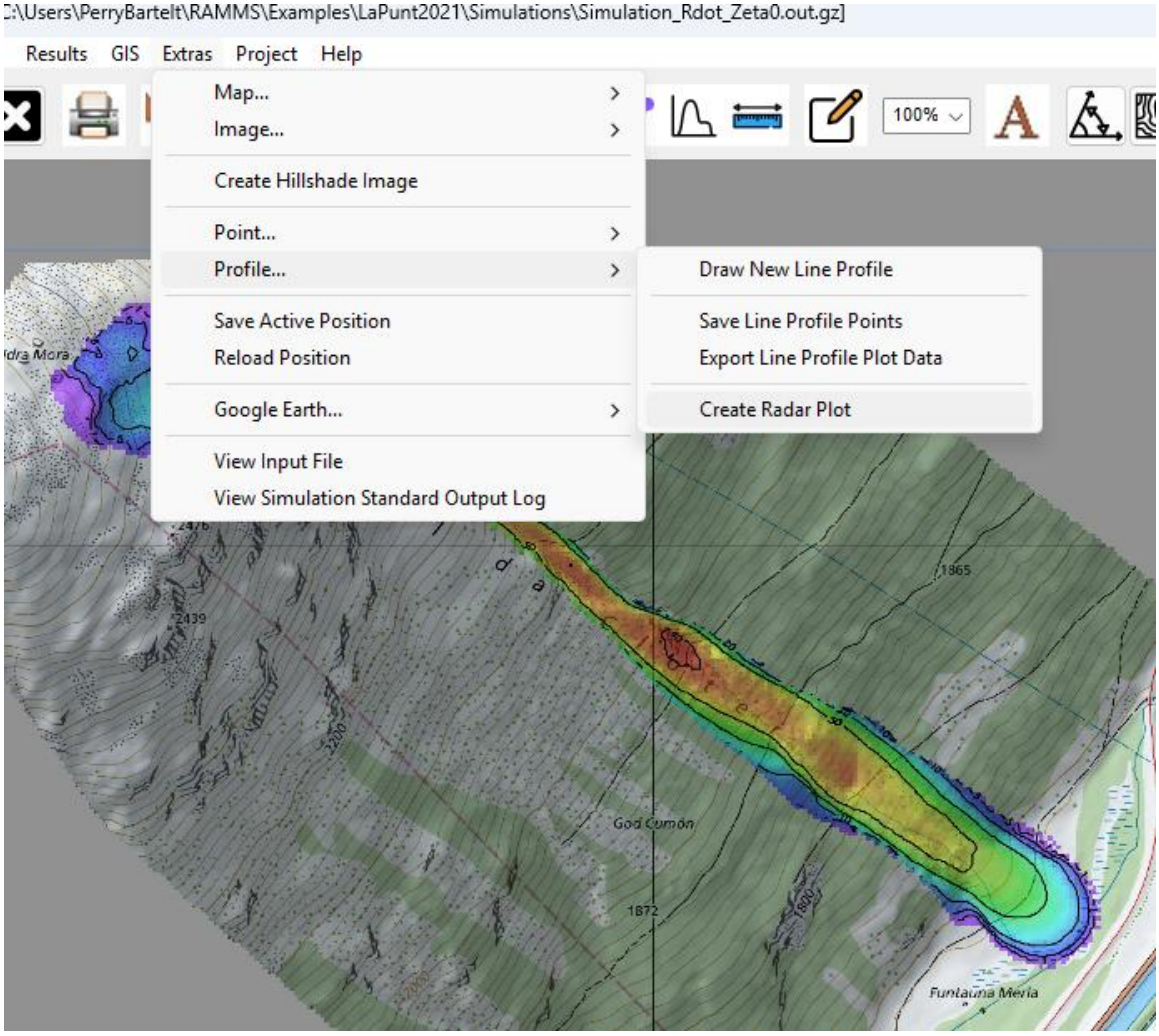


Figure 2.53: The Radar Plot option is found under the Extras pull down menu under Profile.

Step 1: Selecting the Profile

1. Begin by enabling **the visualization of maximum core velocities** in the simulation.
2. First create a **line profile by**
    - Navigate to **Extras** → **Profile...** → **Draw New Line Profile.**
    - **Left click** to define the path of the line profile.
    - Start **above the release zone** and work down the mountain.
    - The final point (last left-click) should be at the **radar location**.

- Ideally, place the profile along the path of maximum velocities (see Figure 2.54) to ensure maximum leading edge velocities.

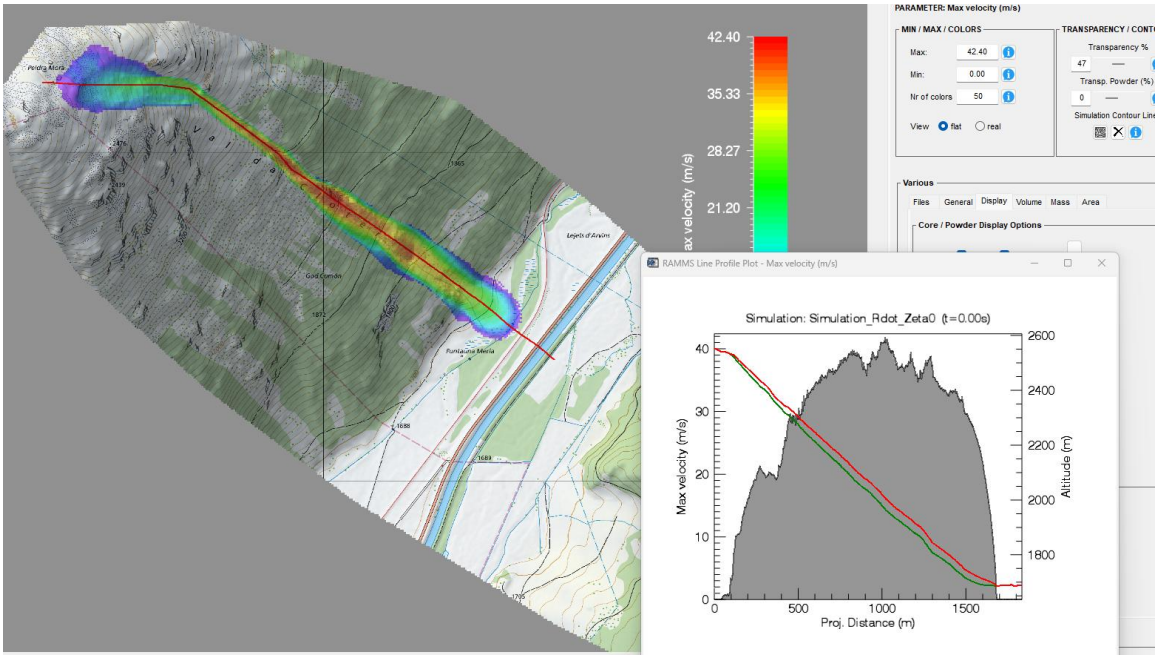


Figure 2.54: A profile has been creating starting avove the release and moving down the mountain. In this case *x*=0 is located an an elevation of 2550m. The profile plot depicts the Eulerian internal velociteies. The avalanche reaches 40m/s.

Step 2: Visualizing Core Heights Over Time

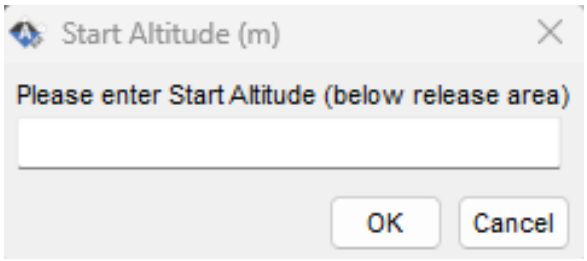
1. Activate the **core height visualization** by clicking the **red "H" button** in the **Display** tab.

2. Set the simulation time to t = 0 to ensure that the avalanche release zone is visible (see Figure 2.54).

Step 3: Generating the Radar Plot

1. Navigate to **Extras** → **Profile** and select "**Create Radar Plot**."

2. A menu will appear, prompting the user to enter a start altitude below the release zone.



- This altitude will be identified on the profile and used to track the arrival time of the leading edge.

3. **The system will then process the entire profile, recording:**
    - **X-Axis**: The time when the avalanche front reaches each point.
    - **Y-Axis**: The distance of each point from the radar location (final point on the profile).
4. The resulting **radar plot** (Figure 2.55) displays distance vs. time, where the slope of the plotted line represents the **leading-edge velocity** of the avalanche.

By following this structured approach, users can create accurate radar plots to compare simulated avalanche velocities with real-world radar measurements, enhancing both validation and analysis of avalanche dynamics.

Once a radar plot is generated, its **numerical values are automatically saved** in the project's subdirectory. The data is stored in a file named **radarplot_data.txt**, which contains two key parameters:

- **Time (s)**: The timestamp for each recorded position.
- **Distance (m**): The distance of the avalanche front from the radar location (the last point on the profile).

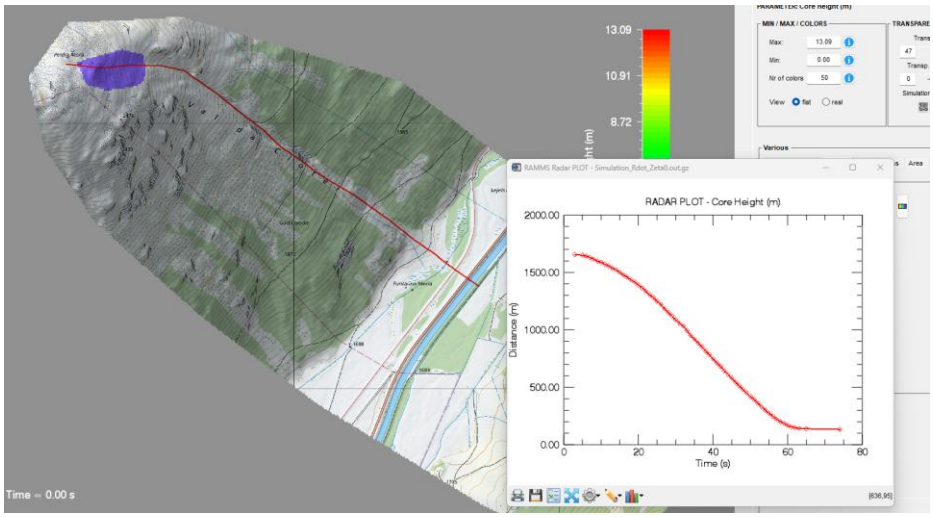


Figure 2.55: Radar Plot. The X-axis of the plot represents time, the Y-axis the distance from the radar. The curve provides the location of the leading edge over time, referenced to the radar location. The slope of the line is the leading-edge velocity.

Example: Calculating Leading-Edge Velocity

By examining the **radarplot_data.txt** file (Figure 2.56), we can extract an example data point:

- At t = 33s, the avalanche front is 994.857m from the radar.
- At t = 34s, the front has moved to 952.303m from the radar.
- The difference in distance: 994.857m - 952.303m = 42.554m.
- Since this movement occurs over 1 second, the leading-edge velocity is calculated as:

$$\mathrm{v} = \frac{\Delta\mathrm{distance}}{\Delta\mathrm{time}} = \frac{42.554\mathrm{m}}{1\mathrm{s}} = 42.554\ \mathrm{m/s}$$

This simulated **leading edge velocity** (42.554 m/s) aligns well with the maximum core velocities obtained from the **Eulerian velocity plot** (Figure 2.54), demonstrating consistency between different velocity analysis methods.

By analyzing the radar plot data, users can validate simulated avalanche dynamics against real-world radar observations, improving the accuracy and reliability of avalanche modeling in **RAMMS::Extended**.

```
Date: 10/Mar/25
File: Simulation_Rdot_Zeta0.out.gz
Parameter: Core Height (m)
Parameter: Time (s), Distance (m)
      3.00000      1655.28
      5.00000      1649.38
      6.00000      1643.00
      7.00000      1629.63
      8.00000      1616.99
      9.00000      1597.64
      10.0000      1590.94
      11.0000      1571.27
      12.0000      1557.66
      13.0000      1537.60
      14.0000      1524.01
      15.0000      1501.73
      16.0000      1476.36
      17.0000      1460.47
      18.0000      1434.66
      19.0000      1416.78
      20.0000      1391.07
      21.0000      1365.79
      22.0000      1337.47
```

Figure 2.56: An example of the *radarplot_data.txt*. Time vs Distance from the radar location.

2.21Open File Browser

	**Open Project Directory**	Quick open of File Browser

In many cases, a project may contain **multiple input files**, making it essential for the user to have a clear overview of all available files within the project directory. Additionally, **RAMMS::Extended** automatically generates various output files—including **profile, plot, volume, and mass data**—which the user may need to access for reviewing specific numerical details.

To streamline this process, **RAMMS::Extended** provides a dedicated **file browser** accessible via a convenient icon. When selected, this tool **opens the project's home directory directly within the GUI**, eliminating the need to navigate external file explorers. This allows users to **immediately locate, open, and analyze files** without disrupting their workflow

2.22 Customizing Display Options

	**Edit GUI Preferences**	Used to change color bar, powder exaggeration, or powder cloud fade time

The **RAMMS::Extended** GUI utilizes a rainbow color bar (Color Table No. 34) by default. However, many users prefer **alternative color schemes** or **customized display settings** to better suit their analysis and visualization needs.

To modify these settings, display options—including color bars and other visual preferences—are defined in the "**GUI Preferences**" file. This file allows users to fine-tune how the GUI presents simulation results.

Accessing and Editing GUI Preferences

To customize display settings, users should:

1. **Click the "Edit GUI Preferences" icon**, located on the left-hand vertical toolbar. Or navigate to **Help** → **Advanced...** → **Edit GUI Preferences**.

2. Upon activation, the **preferences file will open**, allowing users to modify various settings (see Figure 2.57).

Key Preferences and Their Functions

The preferences file contains several important parameters (by keywords) that control the appearance and behavior of the **RAMMS::Extended** interface. The keywords are used to customize program settings. If a keyword is not present in the preferences file, the user may simply add it manually (before the END mark) together with the desired values. After saving the file, the new setting becomes active immediately. Below are some of the most adjusted options:

- **COLORTABLENR** – Determines the type of color bar used in the GUI.
    - The available Color Table Numbers (COLORTABLENR) are listed at the end of the preferences file.
    - Users can change this value to select a different color scheme that better suits their visualization needs.
    - Check out available color schemes by navigating to **Help** → **Advanced...** → **View Available Colortables.**

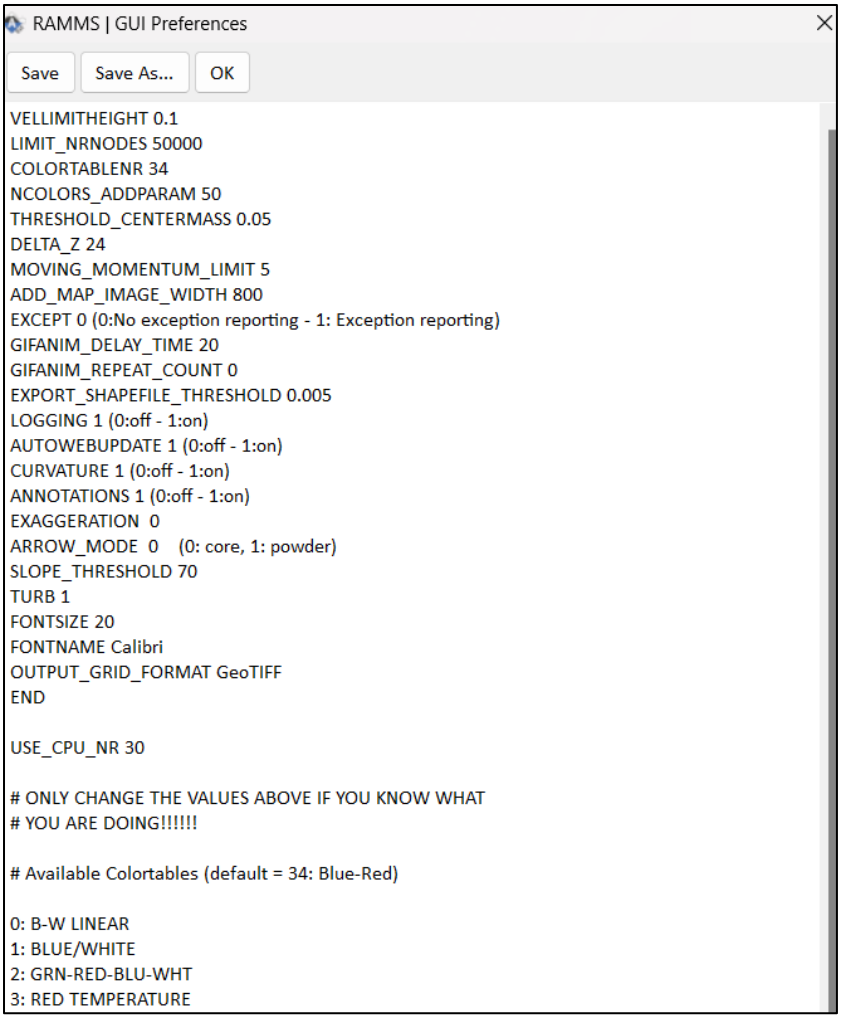


Figure 2.57: The RAMMS::Extended Preferences File. Colorbars are listed at the end.

- **EXAGGERATION** – Defines the scaling factor for displaying core and powder cloud heights.

  Syntax:

  `EXAGGERATION <value>`

  - **Default value: 0** Displays true height, matching the Digital Elevation Model (DEM) scale.

  - **Values > 0** Scales up heights, which can enhance animation realism but may obscure shapefiles and contour lines.

- **POWDER_FADE_TIME** – Controls how the powder cloud fades at the end of a simulation.

  Syntax:

  ```
  POWDER_FADE_TIME <value>
  ```

  - Default value: 20. The GUI begins f**ading the powder cloud 20 seconds before the simulation ends**.

- **TURB** – Enables automatic powder cloud turbulence settings.

  Syntax:

  ```
  TURB <value>
  ```

  - Setting **TURB = 1** activates this option by default.

- **SIMULATION CONTOUR LEVELS Keywords**
  These keywords allow the user to define the **contour levels for the simulation output results**. All contour level keywords share the same behavior:

  Syntax:

  ```
  <KEYWORD> <level_1> <level_2> <level_3> ...
  ```

  - Any number of levels may be provided.
  - Ascending order is important.
  - Units follow the plotted variable (e.g., pressure in kPa, velocity in m/s)..

  Precedence & scope

  - The specific keywords apply to their respective result parameters.
  - The general keyword CONTOUR_LEVELS applies to all other result parameters that do not have a dedicated contour keyword.
  - If both a specific keyword and the general keyword could apply, the specific keyword takes precedence for that parameter.

  Examples:

  ```
  POWDER_P_CONTOUR_LEVELS 1.0 3.0 5.0 10.0
  V_CONTOUR_LEVELS        2.5 5.0 10.0 20.0 40.0
  CONTOUR_LEVELS          0.5 1.0 2.0 4.0 8.0
  ```

  Default Values by Keyword:
  If you do not specify your own levels for a parameter, the program uses the

defaults below (and uses the general keyword CONTOUR_LEVELS for “other” parameters). V_CONTOUR_LEVELS define contour levels for all velocity results.

```
CONTOUR_LEVELS          1.0 3.0 5.0 10.0
POWDER_H_CONTOUR_LEVELS 5.0 10.0 20.0 30.0 50.0 80.0
POWDER_P_CONTOUR_LEVELS 1.0 3.0 5.0 10.0
POWDER_D_CONTOUR_LEVELS 1.5 3.0 5.0 7.0 10.0
CORE_H_CONTOUR_LEVELS   1.0 2.0 3.0 5.0 10.0
CORE_P_CONTOUR_LEVELS   3.0 10.0 30.0 100.0
CORE_D_CONTOUR_LEVELS   50.0 100.0 200.0 300.0 400.0
V_CONTOUR_LEVELS        1.0 5.0 10.0 20.0 30.0 40.0 50.0 60.0
```

- **ELEVATION CONTOUR LEVEL Keywords**
  These keywords allow the user to define the behavior of the **elevation contour levels**. All contour level keywords share the same behavior:

  Syntax:

  ```
  <KEYWORD> <value>
  ```

  - C_THICK: The line thickness is used to draw each contour level. Default value is 2.
  - C_CHARSIZE: The size of the characters used to annotate contour labels. Default value is 1.
  - C_CHARTHICK: The thickness of the characters used to annotate contour labels. Default value is 2.
  - C_LINE_INT: Contour line interval in meters. Default value is 20.
  - C_LABEL_INT: Contour label interval in meters. Default value is 100.

- **ARROW** Options – The preferences file also contains various ARROW settings, which allow users to customize vector displays and flow direction indicators within the simulation.

  Syntax:

  ```
  <KEYWORD> <value>
  ```

  - ARROW_MODE: This keyword controls whether **velocity arrows** in the output plots represent the **core** or the **powder** flow field. Core velocity arrows are shown by default. Values:

- 0 → show velocity arrows for the **core**
- 1 → show velocity arrows for the **powder**

  - ARROW_COLOR: This keyword defines the **color of the velocity arrows** in the output plots. The value must be given as a **string** (text), e.g. red, black, blue, etc. Most common color names are supported. Default is red.

  - ARROW_FACTOR: This keyword defines the **density of velocity arrows** in the output plots. It specifies how frequently arrows are drawn relative to the computational grid. Default value is 1 (every grid cell).
    Example: 2 → show one velocity arrow for **every second grid cell**

  - ARROW_THICK: Thickness of velocity arrows. Default value is 2.

  - ARROW_LENGTHSCALE: Length of velocity arrows. Default value is 1.

  - ARROW_HEADSIZE: Size of velocity arrow heads. Default value is 0.5.

- **MAXHPOW1KPA**
  This keyword acts as a **filtering option** for the visualization of the **maximum powder height**.

  Syntax:

  ```
  MAXHPOW1KPA <value>
  ```

  Behavior:
  - By default, all the powder height values are displayed.
  - If this keyword is enabled (1), the visualization is restricted: only those regions where the **powder pressure exceeds 1 kPa** are shown.
  - This affects **visualization only**; the underlying data remain unchanged.

- **FONTNAME**
  Specifies the **font family** used in the graphical user interface GUI.

  Syntax:

  ```
  FONTNAME <font_name>
  ```

  Default **font name** is Calibri. You can choose any font name available on your system.

- **FONTSIZE**
  Specifies the **font size** (in points) used in the graphical user interface GUI.

  Syntax:

  ```
  FONTSIZE <size>
  ```

  Default **font size** is 20.

- **OUTPUT_GRID_FORMAT**
  This keyword defines the **file format** used when exporting simulation results as raster grids. Default export file format is **GeoTIFF**.

  Syntax:

  ```
  OUTPUT_GRID_FORMAT <format>
  ```

  Format-values:
  - ASCII export results in **ESRI ASCII Grid** format.
  - GeoTIFF export results in **GeoTIFF** format

- **SLOPE_THRESHOLD**
  This parameter defines the **maximum slope angle** (in degrees) at which snow cover is still available. Slopes steeper than this threshold are assumed to be bare, with snow cover depth set to 0.0. Default value is 70° (degrees).

  Syntax:

  ```
  SLOPE_THRESHOLD <angle>
  ```

By modifying these preferences, users can tailor the **RAMMS::Extended** interface to better fit their specific visualization, analysis, and computational requirements.

2.22.1 Documenting and Controlling Workflow

RAMMS::Extended provides several effective options for documenting and managing simulation workflows. A foundational step is to use clear and descriptive file names that reflect the key characteristics of each simulation—such as varying return periods, different release zones, or specific entrainment conditions. This simple yet essential practice enhances traceability and reproducibility.

Beyond naming conventions, **RAMMS::Extended** offers the ability to inspect and save key simulation files, each of which contributes to workflow transparency:

1. **Input File (Extras → View Input File)**
   The input file is the core configuration file parsed directly by the **RAMMS::Extended** C++ code. It precisely captures all parameters defined through the graphical user interface (GUI), including their numerical precision. **This file serves as the definitive reference for simulation inputs.** For user convenience, a more readable version of this information is echoed in the **Standard Output File**, enabling quick validation of GUI-defined values.
2. **Standard Output File (Project → Output Log File)**
   This file provides a comprehensive summary of the simulation setup and results. It includes critical metadata such as (see Figure 2.58):
   - Termination condition (e.g., end time or low mass flux)
   - Real and virtual calculation times
   - Grid resolution details
   - Minimum and maximum values for height, velocity, and pressure of both the core and the cloud
   - Minimum core density (saltation front) and maximum cloud density
   - Release conditions and material parameters (e.g., friction, erosion, forest interaction)
   - Name of the DEM file used
   - Number of nodes and elements involved in the simulation
     This overview helps ensure that the intended physical conditions and parameters were properly implemented.
3. **Simulation Log File (Extras → View Simulation Standard Output Log)**
   This log file captures the **full execution output** generated by **RAMMS::Extended**, written in real time during the simulation run (Figure 2.59). It provides detailed, time-resolved data for each output dump step specified by the user. Analysts frequently use this log to **pinpoint the onset of specific physical processes**, such as the initiation of forest detrainment or the moment the avalanche reaches melting temperature. This makes the log file an invaluable resource for diagnostic and interpretive tasks.

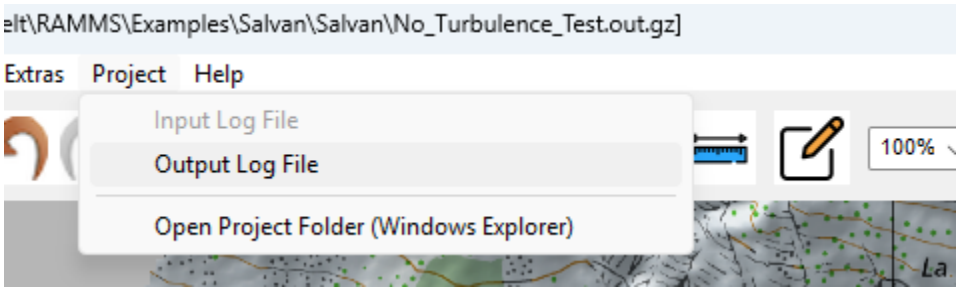


Figure 2.58: Input/output data can be visualized using *Extras → View Simulation Standard Output Log*.

Together, these tools offer a robust framework for documenting, verifying, and understanding the simulation process in **RAMMS::Extended.**

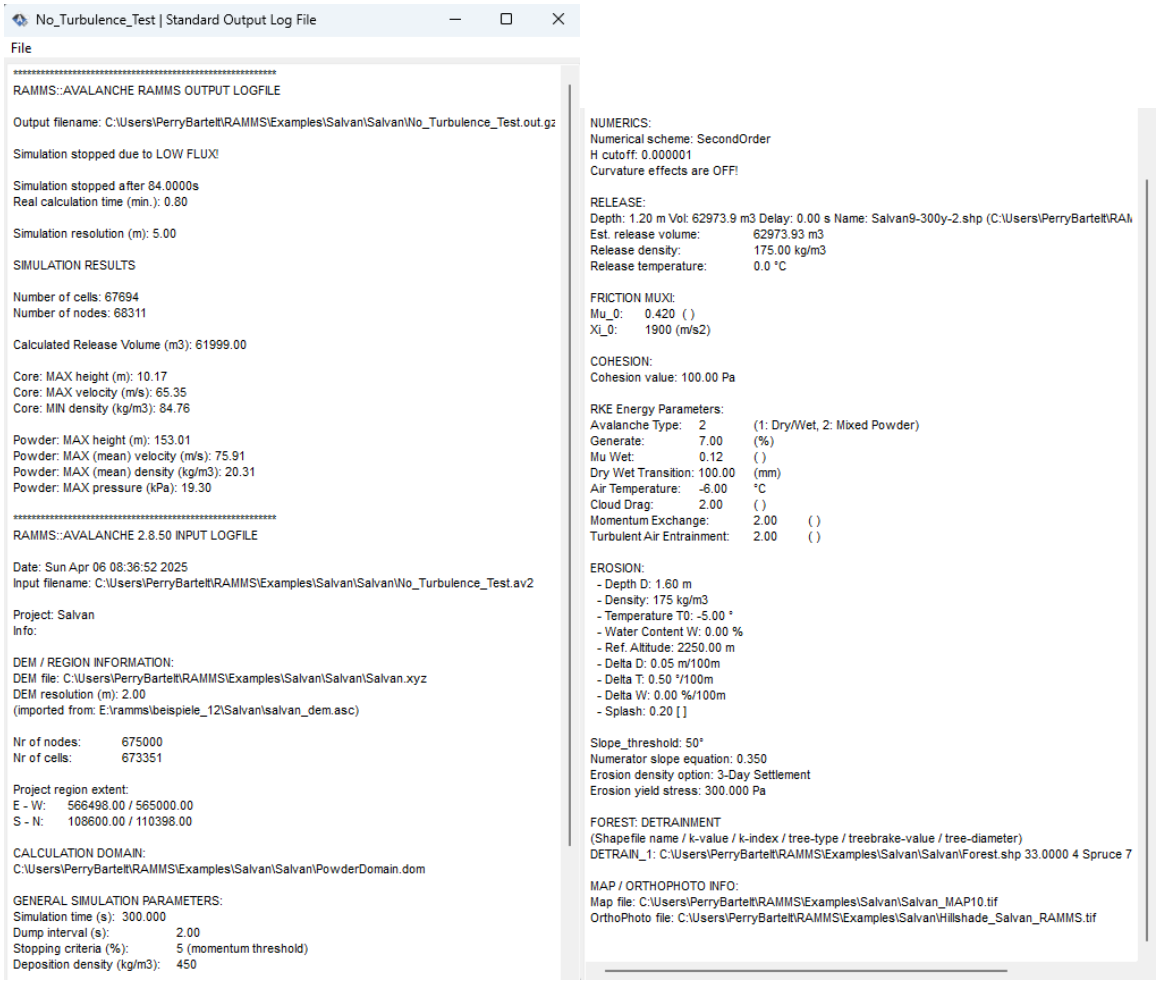


Figure 2.59: The RAMMS::Extended Standard Output File contains an overview of both input and output data of the simulation.

2.23 Closing an Output Session

	**Switch to Input**	Returns directly to Input mode
	**Close Input/Output**	Ends current problem. GUI awaits new project, input or output file read

As an output session nears completion, the user has **three distinct options** for proceeding:

1. **Switch to the Input File for a New Simulation**
   If the user wishes to immediately set up a new simulation with different boundary conditions and parameters, they can transition back to the input file. This is accomplished by selecting the **"Switch to Input File"** icon, conveniently located on the **vertical toolbar** on the left-hand side of the GUI. This option enables seamless continuity, allowing for quick modifications and reconfiguration without exiting the application.
2. **Close the Output Session and Return to Startup Mode**
   The second option allows the user to close the current output session while keeping the GUI active. This is done by clicking the **"Close"** icon, found on the **upper toolbar**. Represented by a simple **“X”**, this icon does **not** terminate the application; rather, it resets the GUI to its **startup mode**, where the user can either load an existing input/output file or create a new project. This option is ideal for users who wish to start fresh without completely closing the program.
3. **Exit the GUI Completely**
   The final option is to **fully close** the session and exit the application. This is done by selecting the **window’s close button (X)** in the upper-right corner of the interface (or by navigating to **Track** → **Exit**). Before closing, the system will prompt the user with a **confirmation dialog**, ensuring that the exit action is intentional. If confirmed, the GUI will shut down, ending the session entirely.

By understanding these options, users can efficiently manage their workflow within the GUI, whether they wish to continue working, restart, or exit the program altogether.

Chapter 3 Friction in the Avalanche Core

3.1 Introduction

The **RAMMS::Extended** module takes a different approach to modeling avalanche friction than the **RAMMS::Avalanche** module. This approach is grounded **in grain flow theory and informed by a combination of experimental data, field observations, and long-standing practical experience.** Decades of work with the **Voellmy model** have contributed to refining and extending its application for more complex flow behavior. Central to this formulation is the core friction term, $S_\Phi(R_\Phi, T_\Phi, m_\Phi)$, which appears on the right-hand side of the momentum equation. Here, R_Φ represents the degree of fluidization within the avalanche core, T_Φ denotes the mean core temperature, and m_Φ describes the water content—each playing a critical role in capturing the evolving rheology of snow under dynamic flow conditions.

$$\left(\hat{h}_\Phi\right)_t + div\left(\hat{h}_\Phi \vec{u}_\Phi\right) = \frac{\rho_\Sigma}{\hat{\rho}_\Phi}\dot{M}_{\Sigma\to\Phi} - \dot{M}_{\Phi\to\Psi} - \frac{\hat{\rho}_\Pi}{\hat{\rho}_\Phi}\dot{M}_{\Phi\to\Pi}$$

$$\left(\hat{h}_\Phi \vec{u}_\Phi\right)_t + div\left(\hat{h}_\Phi \vec{u}_\Phi \otimes \vec{u}_\Phi + p_\Phi I\right) = \vec{G}\hat{h}_\Phi - \frac{\vec{u}_\Phi}{\|\vec{u}_\Phi\|}S_\Phi(R_\Phi, T_\Phi, m_\Phi)$$

$$- \left[(1 + r_\Gamma)\frac{\rho_\Sigma}{\hat{\rho}_\Phi}\dot{M}_{\Sigma\to\Gamma} + \dot{M}_{\Phi\to\Psi} + \frac{\hat{\rho}_\Pi}{\hat{\rho}_\Phi}\dot{M}_{\Phi\to\Pi}\right]\vec{u}_\Phi$$

$$\begin{aligned}\left(\hat{h}_\Phi R_\Phi\right)_t + div\left(\hat{h}_\Phi R_\Phi \vec{u}_\Phi\right) &= [1 - \zeta_\Phi]\left[\alpha_\Phi \dot{W}_\Phi + \rho_\Sigma \varepsilon_\Phi \dot{L}_{\Sigma\to\Phi} - \beta_\Phi \hat{h}_\Phi R_\Phi\right] - \dot{M}_{\Phi\to\Pi}R_\Phi \\ &= [1 - \zeta_\Phi]\dot{\mathcal{P}}_\Phi - \dot{M}_{\Phi\to\Pi}R_\Phi\end{aligned}$$

$$\begin{aligned}\left(\hat{h}_\Phi E_\Phi\right)_t + div\left(\hat{h}_\Phi E_\Phi \vec{u}_\Phi\right) &= [1 - \alpha_\Phi]\dot{W}_\Phi + \rho_\Sigma[1 - \varepsilon_\Phi]\dot{L}_{\Sigma\to\Phi} + \beta_\Phi \hat{h}_\Phi R_\Phi + \varrho_\Sigma c_\Sigma T_\Sigma \dot{M}_{\Sigma\to\Phi} \\ &\quad - \dot{M}_{\Phi\to\Pi}E_\Phi\end{aligned}$$

Historically, Switzerland has relied on the **Voellmy Model** (1955) for avalanche dynamics calculations. The classical Voellmy model simplifies avalanche friction into a two-parameter system, using:

$$S_\Phi = \mu\, N_\Phi + \frac{\varrho g V_\Phi^2}{\xi}$$

where N_Φ is the normal force (stress) and V_Φ the avalanche velocity. The two-parameter **Voellmy model** has proven highly practical, as its parameters distinctly influence different aspects of avalanche behavior: Coulomb friction μ governs the **runout distance**, while ξ controls the **terminal velocity**. This clear separation of roles has made the model particularly valuable, with users—especially in Switzerland—developing extensive expertise in its application over the years.

In the **RAMMS::Avalanche** module, friction (μ, ξ) is assigned fixed values before the simulation begins and remains unchanged throughout the avalanche's evolution. In contrast, **RAMMS::Extended** models flow friction (S_Φ) as a **dynamic process** that adapts to the avalanche's changing state. The frictional resistance in **RAMMS::Extended** is described mathematically as:

$$S_\Phi = \mu(R_\Phi, T_\Phi, m_\Phi) N_\Phi + \frac{\varrho g V_\Phi^2}{\xi(R_\Phi, T_\Phi, m_\Phi)}$$

Here, R_Φ, T_Φ, and m_Φ represent the **fluctuation energy**, **thermal temperature (internal energy** E_Φ), and **wetness** of the avalanche, respectively. Thus, **RAMMS::Extended** builds upon the foundational **Voellmy framework** while expanding it to encompass the complexity of different avalanche flow regimes. Voellmy's assertion that "mixed forms of snow avalanches and powder avalanches are common, and their effects are assessed by studying the borderline cases" is directly addressed in the extended model. By allowing friction parameters to evolve dynamically during the simulation, the extended model eliminates the need to define fixed values for specific avalanche types. Instead, it captures the continuous transitions between **mixed avalanche forms**, providing a more realistic representation of avalanche behavior.

Voellmy noted that "when the air temperature is near freezing, the heat of friction and compression can melt ice, each of which will occur at the surface of the snow crystals." This observation on the role of temperature and friction is directly incorporated into **RAMMS::Extended**, which accounts for temperature-driven processes such as lubrication from melting snow and their influence on avalanche behavior. By integrating these effects, **RAMMS::Extended** builds upon established avalanche dynamics while addressing the **variability and uncertainty** found in real-world scenarios. This post-modern approach acknowledges the complexity of avalanche flow, leveraging the long-standing experience of the **RAMMS::Avalanche** module while allowing friction to evolve and adapt rather than being constrained by fixed parameter classifications.

It is, in this sense, a **natural choice to employ a Voellmy-type rheological model** within **RAMMS::Extended**. Voellmy himself recognized that dry-flowing avalanches and wet snow avalanches are not distinct phenomena but rather limiting cases along a continuum

governed by internal processes. At the heart of his insight was the competition between random kinetic energy—reflected in the degree of fluidization R_Φ and internal energy, represented here by the temperature T_Φ— a proxy for the system's thermodynamic state. In **RAMMS::Extended**, this conceptual framework is carried forward with mathematical precision: both R_Φ and T_Φ are understood to emerge from the same source—**shear deformation**—the essential generator of disorder and energy transformation within the flowing snow mass.

In the late 1990s, there was considerable interest in modeling **active-passive hydrostatic stress transitions** to capture the granular, solid-like behavior of snow (Bartelt et al., 1999). However, after extensive testing and application to real-world avalanches, this concept was eventually abandoned. The approach introduced **numerical instabilities**, particularly in **active flow regions** with **positive longitudinal velocity gradients**. Additionally, the lack of detailed experimental data on cohesive snow behavior made validation impossible. As a result, **active-passive pressure models are not used** in **RAMMS::Extended**.

Another popular avalanche model is the so-called $\boldsymbol{\mu(I_\Phi)}$ **rheology**. The $\mu(I_\Phi)$ rheology offers a theoretically elegant framework grounded in small-scale granular experiments; it is not directly implemented in **RAMMS::Extended** due to key limitations that impair its predictive capability for real geophysical mass flows. Specifically, the $\mu(I_\Phi)$ model assumes a **constant Coulomb friction coefficient** and a steady-state balance between fluctuation energy production and decay, which leads to unrealistic flow behavior during critical acceleration and deposition phases. This simplification results in notable mismatches with observed avalanche dynamics, including **overestimated runout distances** and **misrepresented deposit structures** (Zhuang et al., 2025). In contrast, the $S_\Phi(R_\Phi, T_\Phi, m_\Phi)$ rheology, which explicitly accounts for the non-steady evolution of granular temperature (fluctuation energy R_Φ), provides a more physically realistic and flexible description of flow resistance, yielding better agreement with measured velocities and deposition patterns. Nonetheless, it is possible to apply the $\mu(I_\Phi)$ rheology in **RAMMS::Extended** with the help of **special settings**. Interested users should contact the RAMMS development team.

3.2 Basic Idea: Fluidization and Lubrication

The **RAMMS::Avalanche** module applies depth-averaged equations commonly used in free surface hydraulic flow models, treating avalanches as a continuum fluid. In contrast, **RAMMS::Extended** models the avalanche as a granular ensemble of flowing clumps and clods of snow (Figure 2.1), incorporating the principles of Grain Flow Theory, first introduced by Haff (1983). While both approaches use continuum equations rather than discrete particle models, **RAMMS::Extended** expands hydraulic theory by integrating a grain flow framework.

This extension takes the form of an additional differential equation that tracks the kinetic energy associated with granular fluctuations R_Φ, a quantity sometimes referred to as **granular temperature** or **random kinetic energy**. By including this energy, the model accounts for the shearing-induced dispersion of snow granules, which increases particle spacing and reduces friction—a process known as **fluidization**. This concept aligns with **Voellmy's (1955)** ideas on avalanche flow and traces even further back to **Reynolds (1885)** and his work on granular dilatancy.

In **RAMMS::Extended**, a volume of dry snow particles is defined, with the densest possible packing of these particles referred to as the **co-volume** (Figure 3.1). The friction in this densely packed configuration is governed by the Voellmy parameters (μ_0, ξ_0). However, these values decrease both due to lubrication and fluidization effects.

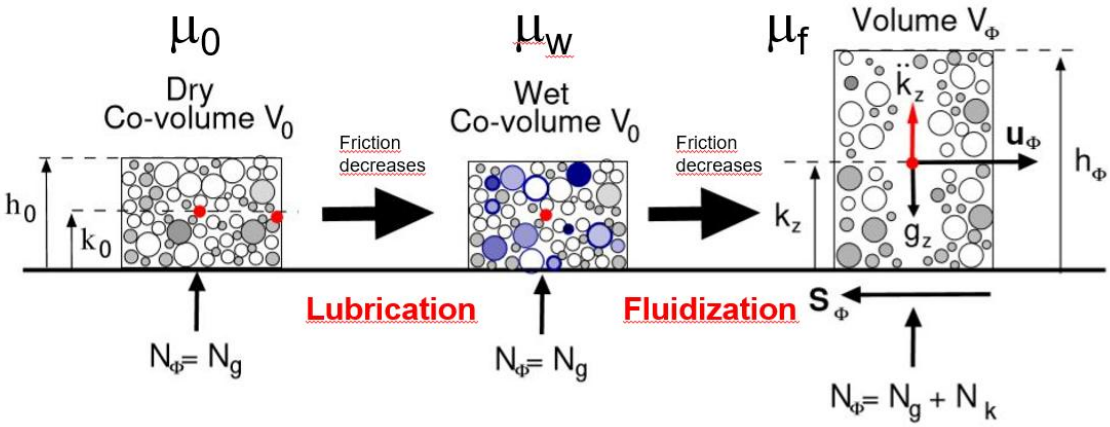


Figure 3.1: In RAMMS::Extended, the avalanche core is represented as a granular ensemble consisting of volumes of snow granules. These particle volumes can expand and contract, altering the flow density of the avalanche. The densest possible packing of granules is referred to as the co-volume. The height of the co-volume is h_0, the center of mass is located at k_0. Frictional processes within the granular ensemble are influenced by two key mechanisms: lubrication, where the wetting of granule surfaces reduces friction, and fluidization, where dispersion increases particle spacing and therefore raises the center-of-mass of the avalanche to k_z. The top surface of the granular ensemble remains dynamic, adjusting its height in response to changes in density and flow conditions.

Following Voellmy, **lubrication** occurs when water accumulates on the surface of snow particles, facilitating sliding with a reduction m-friction. For moderate water levels the surface water increases the cohesive bonding between particles causing them to lock and form particle agglomerates with the net effect of increasing the friction in the x-direction. Moreover, in moist flows, Coulomb friction is reduced and the velocity squared friction is increased, producing flows with long runout, but low travel velocities. The dry parameters of the densest packing are changed from $(\mu_0, \xi_0) \rightarrow (\mu_f, \xi_f)$.

Fluidization occurs when shearing causes the particles to disperse, increasing their spacing and reducing friction (Figure 3.1). In the densest packing state, both Coulomb and turbulent friction are at their highest, but as dispersion increases, friction decreases. Although **fluidization** and **lubrication** can theoretically occur together, wet flows develop strong cohesive forces that prevent dispersion, making the two mechanisms **largely mutually exclusive**. This aligns with Voellmy's approach, which considers avalanches as evolving between borderline cases—either highly fluidized powder avalanches or dense, lubricated wet snow avalanches. By incorporating this principle, **RAMMS::Extended** builds upon and extends the methodology of **RAMMS::Avalanche**, dynamically bridging these flow regimes rather than relying on predefined friction values. Within **RAMMS::Extended**, both cases emerge from the same mathematical model, with the temperature of the snow in the avalanche core often determining which regime develops. This dependence on temperature further complicates defining accurate boundary conditions, such as snowcover temperature, which plays a crucial role in avalanche behavior.

3.3 Snow Rheology

The **RAMMS::Extended** rheology introduces a structured **three-step calculation procedure** designed to capture the dual softening mechanisms that dominate wet snow avalanche dynamics: **basal lubrication** from meltwater and **internal agitation** due to particle-scale fluctuations. This framework extends the traditional Voellmy model by adapting its parameters dynamically based on evolving flow conditions.

3.3.1 Lubrication (Wet snow)

In the **first step**, **RAMMS::Extended** modifies the baseline Coulomb and velocity-dependent friction coefficients to account for the presence of liquid water in the avalanche. Specifically, the Voellmy friction coefficients (μ_0, ξ_0) are both **reduced** as a function of the total meltwater content within the flow according to the exponential relationships:

$$\mu' = \mu_w + (\mu_0 - \mu_w)e^{\left(-\frac{m_\Phi}{m_t}\right)}$$

$$\xi' = \xi_w + (\xi_0 - \xi_w)e^{\left(-\frac{m_\Phi}{m_t}\right)}$$

where:

- (μ_0, ξ_0) are the **dry (baseline) Voellmy friction coefficients**,
- (μ_w, ξ_w) are the **limit values** for wet snow. We take $\xi_w = 800\text{m/s}^2$
- (μ', ξ') represent the **residual friction values** in wet conditions,

- m_t is a **transition parameter** that controls how rapidly friction changes with increasing water content.

A key implication of the **RAMMS::Extended** wet snow rheology model is that **both the Coulomb friction coefficient** (μ_0) and the **turbulent friction coefficient** (ξ_0) are dynamically reduced as meltwater accumulates. This reduction signals a dual transition in the flow regime: **a softening of basal friction due to lubrication**, and **a simultaneous increase in velocity-dependent friction**, which reflects a shift toward more viscous, plug-like flow behavior.

When no meltwater is present $(\mu', \xi') = (\mu_0, \xi_0)$.

Physically, the transition from dry to wet snow avalanche behavior is governed by the **accumulation of meltwater on the surfaces of individual snow granules**. As a thin water film develops—typically due to frictional shear heating—basal friction is **reduced through lubrication**, weakening the Coulomb resistance (μ'). Simultaneously, the **lubricated grains begin to coalesce into larger aggregates or clumps through sintering processes**. These agglomerates form a semi-cohesive structure within the flow, inhibiting particle rearrangement and increasing internal resistance. This leads to a rise in effective viscosity and the emergence of plug-like deformation regimes. In **RAMMS::Extended**, this behavior is captured by reducing both μ_0 and ξ_0 dynamically, where the decrease in ξ_0 corresponds to an **increase** in the velocity-dependent friction. This **dual modification** reflects the **complex rheology of partially melted or wet snow**, where **lubrication softens basal sliding while aggregation stiffens internal flow**. To model this effectively, RAMMS::Extended adopts the Swiss standard for velocity-dependent resistance in wet snow conditions, using a recommended value of: $\xi_w = 800\text{m/s}^2$.

It is important to note that this approach is designed specifically for **modeling dry (cold) snow avalanches that transition into wet, lubricated regimes—not for flows that are fully saturated from the outset.**

For modeling such transitional flow behavior, we recommend the following parameter ranges:

- μ_w between **0.10 and 0.12**,
- m_t between **1 mm and 10 mm**

The **small values of** m_t reflect the high sensitivity of snow to initial wetting: even a thin meltwater **film can initiate sintering and aggregation**, dramatically increasing resistance. This recommendation comes directly from investigations of granules in wet snow flows (Bartelt and McArdell, 2009).

Figure 3.2: The granules of a wet snow avalanche. Particle agglomerates are formed that increase internal shear resistance, but basal sliding friction is reduced.

In contrast, when modeling flows that are **fully saturated from the onset**—such as flows where **significant water is already present in the pore space**—the mechanics are fundamentally different. In these cases, snow grains are unable to form agglomerates, and cohesive bonding is largely absent. The flow is dominated by water-saturated, free-shearing granular motion. For such regimes:

- We recommend setting μ_w **=0.08 – 0.15**
- The transition parameter m_t should be **increased to 100mm to 1m**, reflecting the diminished ability of the snow to generate cohesive structures.

Under these saturated conditions, the material shears easily, friction remains low, and the transition from dry to viscous flow behavior is either minimal or absent. Adjusting m_t accordingly helps prevent artificial stiffening in flows that are already fully saturated with little internal shear resistance.

3.3.2 Fluidization

In the fluidized section of **RAMMS::Extended**, we first consider the physical process of **lubrication**, where fine particles and interstitial fluids reduce friction between grains. Building on this, we then address **fluidization**—the phase in which particles begin to disperse and lose sustained contact with one another. A key factor in this process is **cohesion**, the inter-particle bonding that enables and controls the extent of dispersion. An important outcome of fluidization is the generation of **fluctuation energy** R_Φ, also known as **granular temperature** or **random kinetic energy**, which quantifies the intensity of particle motion and the degree of energy dispersion within the avalanche.

Increased fluctuation energy is associated with reduced flow density and a corresponding decrease in the fluidized friction coefficient (μ_f, ξ_f). The frictional parameters decrease from their wet (μ', ξ') or baseline values (μ_0, ξ_0) as the granular fluctuation energy R_Φ increases. This decrease in friction reflects the opening of spaces between granules due to shearing, which reduces particle collisions and interactions, thereby lowering friction.

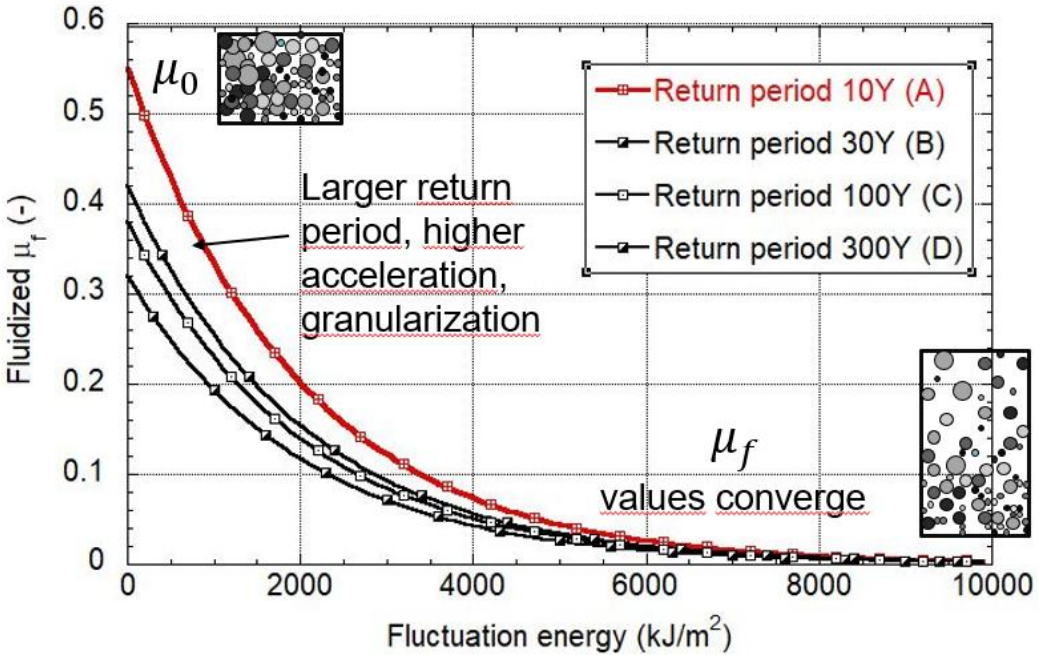


Figure 3.3: The greater the fluctuation energy, the lower the flow density and the smaller the fluidized friction coefficient, μ_f. At high fluctuation energy levels, (R_Φ) approaches a common value, $\mu_f \rightarrow 0$. The curves differ at low fluctuation energies. When a slab releases it has zero fluctuation energy. During the break-up process fluctuation energy is created. When a slab releases with a significant fracture height, the increased overburden stress enhances fragmentation and granularization of the slab. Deeper slabs —and by extension, higher return periods—are associated with faster break-up and smaller friction. To reflect this behavior, lower μ_0 values are assigned for longer return periods.

To quantify this behavior, we employ exponential relationships (Figure 3.2):

$$\mu_f = \mu' e^{\left(-\frac{R_\Phi}{R_0}\right)} \text{and} \qquad \xi_f = \xi' e^{\left(\frac{R_\Phi}{R_0}\right)}$$

where:

- (μ', ξ'): Initial friction coefficient, could be wet or dry, corresponding to the co-volume.
- R_Φ: Fluctuation energy or granular temperature (J/m³).
- R_0: Activation energy (J/m³), a constant valid across all snow types (2kJ/m³).

This **exponential relationship** is based on the idea that friction decreases and approaches zero as fluctuation energy increases. The rate at which friction **diminishes depends on the value of R_Φ,** capturing the shift from a dense, friction-dominated regime to a fluidized state with minimal friction. By accounting for these dynamics, the model effectively simulates the transition from dense, inertial flows to fluidized, turbulent regimes, providing insights into powder cloud formation.

This formulation is grounded in both **historical precedent** and experimental findings, each contributing critical insights into the evolution of avalanche friction modeling. Historically, the standard Voellmy law with a constant Coulomb friction coefficient μ has been widely applied, despite the lack of a clear physical explanation for the low value of μ =0.155 used in practice for 300-year extreme avalanches. Sliding friction measurements in snow typically yield significantly higher values, suggesting that this low coefficient stems not from direct measurement but from back-calculations of real avalanche events. **RAMMS::Extended** builds on this foundation by initially applying higher μ_0 values, which, under appropriate conditions—such as cold temperatures and significant entrainment—can evolve dynamically to approach μ_f=0.155 or even lower in specific cases. **Friction is not a fixed value, but a dynamic process controlled by the avalanche boundary conditions**.

The primary experimental evidence supporting this approach arises from snow chute experiments, which revealed the presence of hysteresis in the relationship between shear and normal stresses. These experiments demonstrated that for the same normal stress (or flow height), two distinct shear stresses could exist—an observation that cannot be captured using the standard Voellmy relationship with constant parameters. In avalanche flows, fluctuation energy is highest at the leading edge of the avalanche, where flow velocities and shearing gradients peak. By making Coulomb friction a function of fluctuation energy, the **RAMMS::Extended** formulation addresses longstanding inconsistencies in modeling while providing a mechanism to incorporate observed complexities in snow behavior more systematically.

As the fluctuation energy R_Φ increases, the resistance to shearing diminishes. This self-limiting mechanism inherently moderates the growth of fluctuations, as depicted in Figure

3.4. Should the shearing resistance approach zero, the system ceases to generate further fluctuation energy, and R_Φ naturally begins to decay. This interplay between the forces driving and resisting fluctuation growth creates a dynamic feedback loop: higher shearing accelerates the production of fluctuations, while the resulting fluctuations, in turn, reduce the shearing.

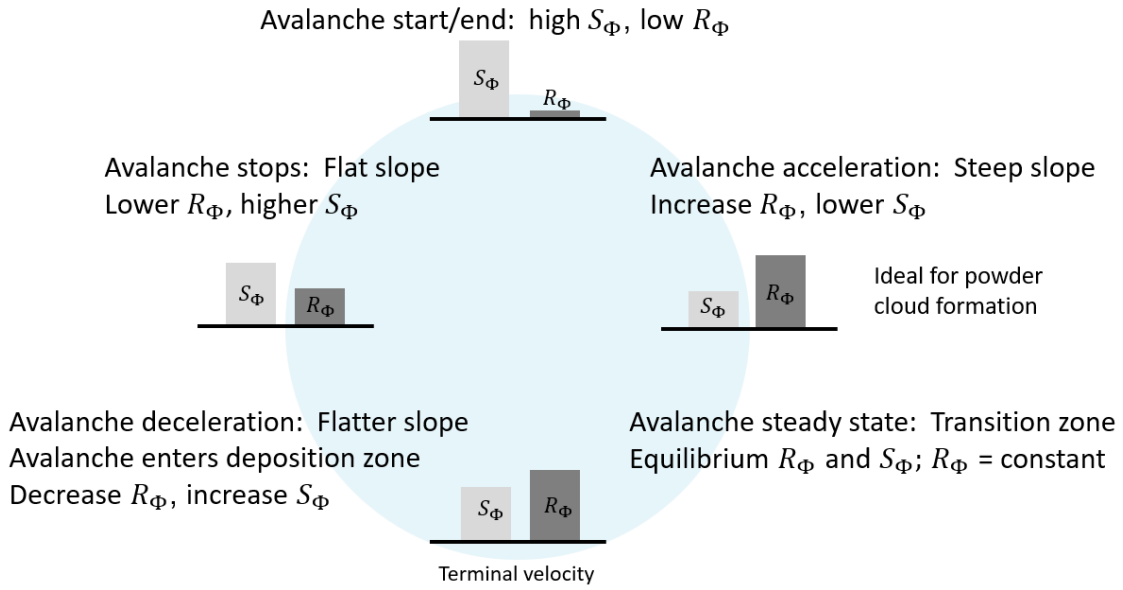


Figure 3.4: The relationship between shearing and fluctuations in the life cycle of an avalanche, starting from initiation, acceleration, attainment of steady state and runout. With this system, overshoots in fluctuations are always damped, leading to inherently stable systems.

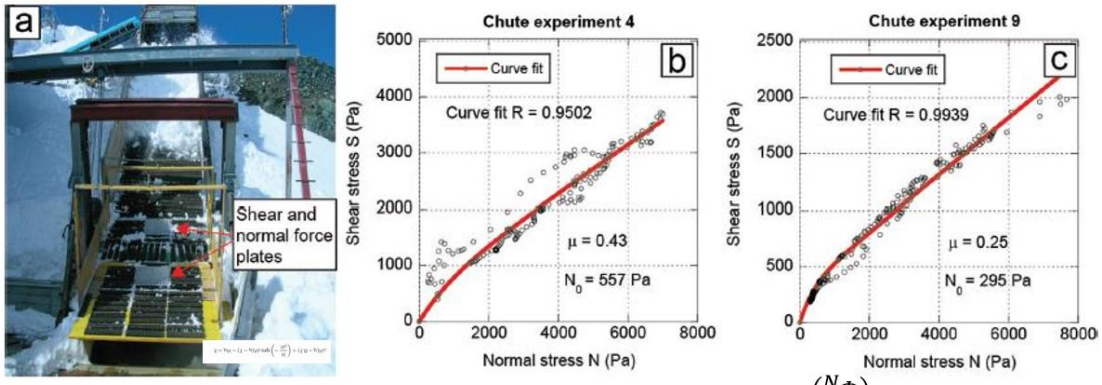
Crucially, this process is deeply influenced by the slope angle. On steep slopes, where gravitational forces dominate, the avalanche enters a regime of sustained, yet limited, fluctuation growth. The continuous downward motion of the avalanche sustains the production of fluctuations, while the limiting effect of the frictional model ensures stability. Overshoots in the production of R_Φ will necessarily decay, indicating **an intrinsic stability in the model**. Conversely, on flatter slopes, where gravitational driving forces are weaker, fluctuations are forced to decay rapidly, and the avalanche eventually comes to a halt. The delicate balance between the production and decay of fluctuations, regulated by the extended Voellmy shearing relationship (Figure 3.3), thus governs the avalanche's behavior, and this dynamic "race" plays a pivotal role in determining the extreme runout distances characteristic of large avalanches.

3.3.3 Cohesion

Experimental measurements reveal that shearing in snow avalanches can be effectively described by a Coulomb-type friction law. However, at low overburden stresses, snowflows display cohesive behavior, deviating from pure Coulomb mechanics (Platzer et

al., 2007; Bartelt et al., 2015). This cohesion introduces additional complexity to modeling shear stress in snow avalanches, particularly under varying stress conditions.

The Coulomb friction parameter μ is determined as the **secant modulus** in the space of shear stress (S) versus normal stress (N). Unlike the tangent modulus, which measures the instantaneous slope, the **secant modulus** provides a measure of the average friction across the stress range. Experimental data also highlight a distinct change in curvature of the S vs. N relationship at a specific value, denoted as N_0, where the flow transitions between cohesive and frictional regimes (Figure 3.5).



$$S_\Phi = \mu_f N_\Phi - (1-\mu_f) N_0 e^{\left(\frac{N_\Phi}{N_0}\right)} + (1-\mu_f)\, N_0$$

Figure 3.5: **In the early 2000s the SLF conducted snow chute experiments to investigate the behavior of flowing snow (Platzer et al., 2007). Researchers embedded shear and normal force plates into an experimental chute to measure the relationship between shear stress and normal stress (S/N). Their findings revealed two key results: (1) The S vs N curves were not perfectly linear, as predicted by a Coulomb stress law. (2) Many experiments exhibited hysteresis from the front to the tail of the avalanche, meaning that for a given normal stress, two different shear stress values could exist.**

At N_0, the curvature of the shear stress curve changes markedly, defining the transition between cohesive and friction-dominated behavior. Notably:

- At $N_\Phi = 0$**:** The Coulomb friction coefficient reaches a maximum value, with μ=1, as cohesive forces dominate.
- At $N_\Phi \to \infty$**:** The friction coefficient asymptotes to its fluidized value, $\mu = \mu_f$, representing fully frictional interactions at high normal stresses.

This transition is empirically modeled using the following relationship (red lines in Figure 3.5):

$$S_\Phi = \mu_f N_\Phi - (1-\mu_f) N_0 e^{\left(\frac{N_\Phi}{N_0}\right)} + (1-\mu_f)\, N_0$$

Here, μ_f represents the fluidized friction coefficient, while N_0 signifies the characteristic normal stress at which cohesion begins to diminish. This equation captures the nonlinear interplay between shear and normal stress, offering a framework for simulating realistic avalanche behavior under a wide range of stress conditions.

The N_0 value, a critical parameter in defining the shear-normal stress relationship of avalanches, is adjustable in the Friction tab (see below). This parameter helps determine the transition between dry and wet avalanche behaviors and influences whether basal slip occurs.

Dry Avalanches:

- **Range:** 0.0 < N_0 < 200 Pa.
- Represents minimal slip resistance, characteristic of granular, non-cohesive snow (mixed flowing, or powder snow avalanches).

Wet Avalanches:

- **Block-Flow Scenario, No Internal Shearing:**
    - **Value:** $N_0 = 0$ Pa
    - Indicates no resistance at the base, leading to a completely lubricated flow (block flow regime).
- **No-Slip Scenario with Internal Shearing:**
    - **Value:** $N_0 =$500 Pa
    - Represents significant basal resistance, limiting fluidized flow and emphasizing cohesive, dense wet avalanche characteristics.

The cohesion parameter N_0 effectively increases the shear stress across all normal stress levels, particularly under conditions of low overburden pressure. Its implementation in **RAMMS::Extended** ensures that shear stress becomes negligible when normal stress approaches zero, avoiding the unrealistic scenario of non-zero shear stress at zero normal stress. Additionally, because N_0 primarily affects low normal heights, it significantly hinders the movement of flows with small flow heights. This often prevents the formation of secondary flow arms, contributing to a more realistic simulation of avalanche behavior.

3.4 The Friction Input Tab

The **Friction Tab** (Figure 3.6) provides an interface for configuring all friction parameters required for a **RAMMS::Extended** simulation. Like **RAMMS::Avalanche**, **RAMMS::Extended** employs return period and volume to define three key friction parameters: μ_0, ξ_0, and N_0.

The Friction Tab is divided into two distinct sections:

1. **Upper Region**: This section allows users to set the general avalanche parameters, including return period, size, and the global values for μ_0, ξ_0, and N_0. These parameters are foundational for capturing the overall frictional behavior of the avalanche as it evolves across the terrain.

2. **Lower Region**: This section is dedicated to defining localized friction zones, which can either increase or decrease friction in specific areas. Such adjustments are crucial for simulating complex avalanche scenarios involving variations in surface properties or terrain features.

Upper section: Return period, size and μ_0, ξ_0 and N_0

Lower section: Local friction areas

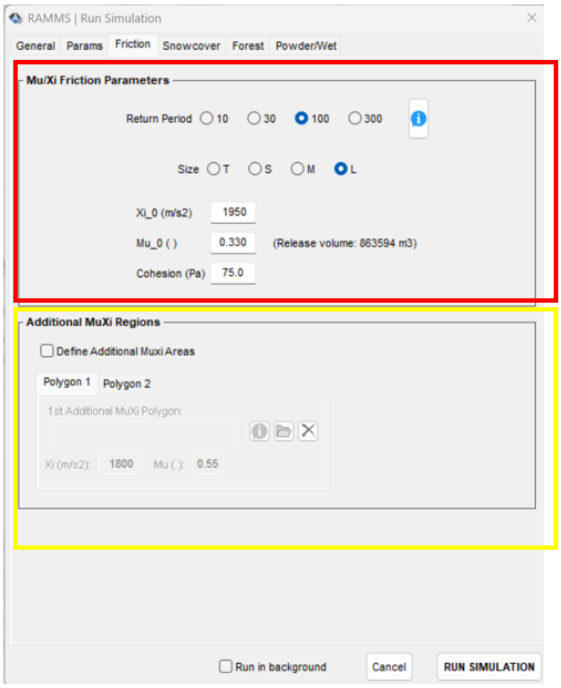


Figure 3.6: The RAMMS::Extended friction tab. The tab is divided into sections. The upper section is used to select avalanche return period, size and base friction values. The lower section is used to define up to two local friction polygons.

3.4.1 Return Period and Volume

RAMMS::Extended is specifically designed to simulate **avalanche scenarios**, with a primary focus on applications in the Swiss Alps. These scenarios are structured around two fundamental parameters: **return period** and **three-day new snow accumulation**, ensuring a data-driven approach to hazard assessment.

To determine **release heights** and **entrainment depths**, extreme value statistics are employed to estimate new snow accumulations for return periods of **10**, **30**, **100**, and **300** years. This method provides a statistically robust and consistent framework for defining realistic avalanche scenarios based on historical snowfall data and climatological trends. The resulting snow accumulation values are specified in the **Snowcover Tab**, where users input relevant snowcover parameters to tailor the simulation to specific conditions. A detailed explanation of this process is provided in the following chapter.

The **Friction Tab** is used to define the **avalanche return period** for a given scenario. Friction parameters are then assigned based on both **avalanche size** and **return period**, ensuring that the simulation reflects realistic flow conditions.

3.4.1.1 Volume Categories in RAMMS::Extended

Similar to **RAMMS::Avalanche**, **RAMMS::Extended** classifies avalanches into four **volume categories**:

- **Tiny (T):** <5000m^3
- **Small (S):** 5’000-25’000m^3
- **Medium (M):** 25’000-60’000 m^3
- **Large (L):** >60’000 m^3

Before selecting the **Friction Tab**, it is essential to first define both the **release zone** and **fracture depth**. **RAMMS::Extended automatically calculates the avalanche release volume** and assigns it to one of the predefined categories. However, in certain cases, the user should **manually override** the automatically assigned volume category to ensure a more accurate representation of avalanche dynamics.

3.4.1.2 When to Manually Override the Volume Category

There are three key situations where the **automatically assigned volume category** may not accurately reflect real-world conditions, requiring **manual adjustment**:

1. **Multiple Release Zones**
    - **RAMMS::Extended** assigns a volume category based on the **total** release volume from **all** release zones combined.

- If multiple release zones are used in the simulation, the total volume often results in classification as a **Large (L) avalanche**.

- Each individual avalanche may be smaller.

- **Solution:** The user should select the **appropriate volume category (T, S, M, L) based on the size of a single release zone, rather than the total of all release zones.**

2. **Avalanches Spanning Multiple Torrents**

    - A single release zone may span several **torrents**, with the avalanche mass naturally **dividing into multiple smaller flows** after release.

    - Although the total release volume may classify the avalanche as **Large (L)**, the actual **individual channelized avalanches** are smaller.

    - **Solution:** The user should manually select the **volume category based on the expected avalanche size within each torrent or drainage channel.**

3. **Avalanches Near Category Boundaries**
    - Often, an avalanche's **release volume** falls near the boundary between two volume categories. For example, a release zone with a volume of **25,500 m³** is close to both the **Small (S)** and **Medium (M)** classifications. In such cases, relying solely on automatic assignments may not fully capture the variability in avalanche behavior.
    - **Solution:** The user should **test both scenarios** by manually overriding the automatic classification and running simulations in both **adjacent categories**. This approach allows for a better understanding of how the avalanche behaves under different friction conditions, improving the robustness of the analysis.

By carefully assigning **return period and volume classifications**, users ensure that simulations capture the true behavior of avalanches, preventing misclassification and improving the accuracy of hazard assessments.

The **friction parameters** (μ_0, ξ_0, N_0) for each **volume category** and **return period** are detailed in Table 3.1. These parameters define the resistance characteristics of avalanche flow under different conditions, ensuring consistency with observed avalanche behavior.

Release Volume (m^3)	Category	10Y	30Y	100Y	300Y
< 5000	Tiny (T)	A	A	B	C
5’000-25’000	Small (S)	B	B	C	D
25’000-60’000	Medium (M)	C	C	D	E
>60’000	Large (L)	D	D	E	F

Table 3.1: Friction categories for return periods 10Y, 30Y, 100Y and 300Y and release volumes. Friction categories are defined in Table 3.2.

RAMMS::Extended categorizes avalanches into **six distinct friction classes**: **A, B, C, D, E, and F**. Each class is associated with a unique set of **friction parameters**, which are listed in Table 3.2. These classifications provide a structured framework for modeling avalanches with varying levels of mobility, energy dissipation, and flow characteristics.

Parameter	A	B	C	D	E	F
μ_0	0.55	0.48	0.42	0.37	0.33	0.30
ξ_0 (m/s^2)	1800	1850	1900	1900	1950	2000
N_0 (Pa)	200	150	100	100	75	50

Table 3.2: RAMMS::Extended friction categories. The largest change is in the Coulomb friction which starts at $\boldsymbol{\mu_0 = 0.55}$ **(A)** and ends at $\boldsymbol{\mu_0 = 0.30}$ **(F).**

An analysis of the **six friction categories** reveals that the most significant variation occurs in the μ_0 **parameter**, which plays a dominant role in defining avalanche flow resistance. In contrast, the ξ_0 **parameter** remains relatively stable, fluctuating only slightly between **1800 m/s² and 2000 m/s²**.

Notably**,** this range is consistent with the **extreme values**—corresponding to **100-year to 300-year return periods**—used in **RAMMS::Avalanche**, reinforcing the model’s continuity with established hazard assessment practices.

Similarly, the N_0 **values** associated with **cohesion effects** exhibit minimal variation across categories, indicating that while cohesion influences avalanche behavior, it does not fluctuate significantly across different return periods or volume classifications.

The information regarding friction parameters, avalanche volume and return period can be accessed via the **blue info button** in the **Friction Tab.**

3.4.1.3 Variation of μ_0 with Return Period

The differences in μ_0 **values** for each **return period** are illustrated in **Figure 3.3**. It is important to emphasize that μ_0 represents the **initial friction** at $R_\Phi = 0$ (when no fluctuation energy is present), rather than the friction experienced during avalanche motion. As the avalanche evolves, **friction decreases** according to an **exponential flow law**, meaning all return periods tend to the **same asymptotic value** at high **fluctuation energy** levels.

As fluctuation energy increases:

- **Flow density decreases**, leading to a more fluidized state.
- **The fluidized friction coefficient** μ_f **drops**, eventually converging toward $\mu_f \to 0$.
- **Primary differences in** μ_0 **occur at low fluctuation energy**, meaning that the distinction between return periods is most evident at **avalanche initiation** rather than in fully developed flow.

Influence of Release Slab Break-Up on μ_0 Values

The key takeaway from these variations is that **the primary difference in** μ_0 **values is linked to the process of slab break-up and granularization.** When an avalanche is released, the slab initially has **zero fluctuation energy**. Fluctuation energy is generated during the break-up phase, as the overburden stress facilitates **fragmentation** and **granularization** of the snowpack.

- **Deeper fracture heights (longer return periods) result in faster slab break-up.**
- **Higher overburden stress accelerates granularization, reducing friction more rapidly.**
- **This effect is particularly pronounced in extreme return periods, corresponding to friction categories D, E, and F.**

Implications for RAMMS::Extended Users

An important consequence of this analysis is that **fracture depth and return period are closely linked**. Users should be aware that **larger fracture depths**—associated with extreme return periods—naturally lead to **lower initial friction values** (μ_0). This relationship is critical for selecting appropriate friction parameters and understanding how avalanche behavior changes across different hazard scenarios.

3.4.1.4 Consideration for Artificially Released Avalanches

An important side note is that **RAMMS::Extended is frequently used to simulate artificially triggered avalanches**, such as those initiated by **explosive charges** placed in the snowpack for controlled release. In these cases, the **slab break-up process is significantly accelerated**, as the detonation disrupts the snow structure and promotes rapid fragmentation.

Because of this **enhanced granularization**, artificially triggered avalanches often behave similarly to **natural avalanches with longer return periods**, where **lower μ_0 values** are observed due to faster break-up. **When back-calculating artificially released avalanches, it is often necessary to manually reduce the μ_0 coefficient** to account for the effect of the explosive charge and achieve a more accurate simulation.

3.4.2 Additional Mu/Xi Regions

The lower region of the **Friction Input Tab** allows users to define additional (μ_0, ξ_0) regions, enabling localized friction adjustments. This feature is particularly useful for refining simulations in areas where friction differs from the general avalanche parameters. **Typical applications include:**

- **Zones with reduced friction:**
    - **Starting Zones with Weak Layers**: Avalanches originating in starting zones with weak layers often experience a significant reduction in friction. The collapse of weak layers creates smooth, low-resistance surfaces beneath the release slab, facilitating faster initial movement.
    - **Iced Surfaces**: Simulating avalanche flow over frozen surfaces, such as iced-over lakes or glaciers, requires defining regions with minimal friction to accurately represent the nearly frictionless conditions.
- **Increased Friction Zones:**
    - **Dams and Energy Dissipation**: In scenarios involving avalanche interaction with dams, localized regions of increased friction are often necessary. Dams typically dissipate more energy upon impact than would be expected from standard surface roughness changes captured by the DEM. This

adjustment allows for more realistic modeling of energy loss and runout distances after dam impact.

To enable this option, first select **"Define Additional Mu/Xi Areas"**, then upload a **shapefile** into the **"Additional Mu/Xi Polygon"** input field. Within a single simulation, users can specify up to two polygon regions with customized μ_0 and ξ_0 values, allowing for localized friction adjustments. The N_0 cohesion parameter cannot be modified locally, as it is primarily influenced by snow properties rather than surface characteristics.

The **"Define Additional Mu/Xi Areas"** option **should not** be used to represent forested regions. Instead, users should configure forest effects through the **Forest Tab**, which is specifically designed to account for the impact of trees on avalanche dynamics.

Model Parameter	Component	Symbol	Parameter Range Constant Scenario
Voellmy Friction Parameters	Φ	$S_\Phi(\mu_0, \xi_0, N_0)$	Scenario-based return period recommendations
Production of RKE by Shearing	Φ	α_Φ	Production of fluctuation by shearing $\alpha_\Phi = 7\%$ (Dry, powder) $\alpha_\Phi = 5\%$ (Wet snow) $\alpha_\Phi = 0\%$ (Voellmy) $0 \leq \alpha_\Phi \leq 8\%$ (Range)
Dispersion	Φ	ζ_Φ	Fraction of random energy creating dispersion in avalanche core $\zeta_\Phi = 0.20$
Surface Melting Temperature	Φ	T_m	Melting temperature $T_m = -1℃$.
Activation Energy	Φ	$S_\Phi(R_0)$	Change of friction with increasing fluctuation energy $R_0 = 2$ kJ/m³
Splashing	Γ	γ_Γ	Determines how much erodible mass enters the splashing front $\gamma_\Gamma = 0.2$ (Dry, powder) $\gamma_\Gamma = 0.0$ (Wet snow) $0 \leq \gamma_\Gamma \leq 0.5$ (Range)
Particle restitution coefficient	Γ	r_Γ	Particles reach double the speed of the avalanche core, r_Γ =1

Table 3.3: User-defined parameters in RAMMS::Extended. Recommended values and parameter ranges.

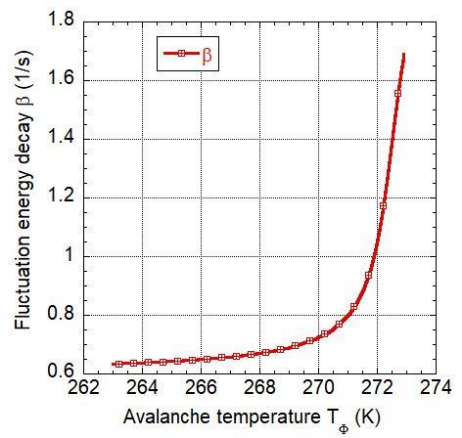
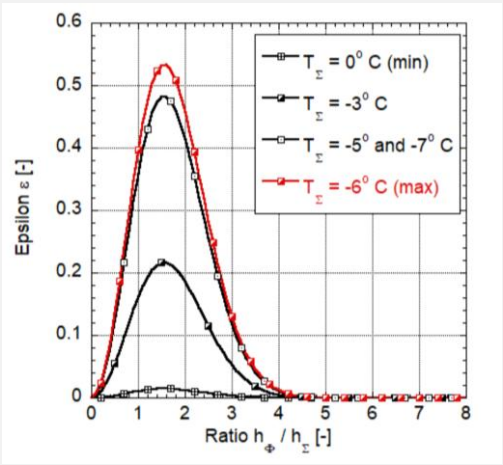
Parameter	Definition/GUI	Range
β_Φ	**Decay** $\beta_\Phi = 1.40 + \frac{1.7}{\pi} 1.7\, atan\left[1.6(T_\Phi - 272.5)\right]$ 	β_Φ = 1/s (Dry, powder) β_Φ = 2/s (Wet snow) $0.5 \leq \beta_\Phi \leq 2.0$ (Range)
ε_Φ	**Partitioning Entrainment** $\varepsilon_\Phi = \boldsymbol{f(T_\Sigma)g(r)} = \left[e^{\frac{-(T_\Sigma - 267)^2}{10}}\right]\left[\frac{r^2 e^{\left(\frac{-r^2}{2a^2}\right)}}{a^3}\right]$ $r = h_\Phi / h_\Sigma$ $a = 1.1$ 	$0 \leq \varepsilon_\Phi \leq 0.5$ (Range) See Chapter 4

Table 3.4: Temperature-dependent parameters in RAMMS::Extended. The RAMMS::Extended model contains four constants that cannot be changed by the user: (1) the splashing restitution coefficient $r_\Gamma = 1$ leading to particles in the pre-front travelling with twice the velocity of the leading edge, (2) the melting temperature $T_m = -1$℃ indicating that melting occurs before the mean temperature T_Φ of the avalanche reaches $T_\Phi = 0$℃ and (3) the activation energy R_0 =2kJ/m³ controlling the change in the baseline friction values (μ_0, ξ_0) with changes in the fluctuation energy R_Φ and finally, (4) the fraction of random kinetic energy that drives the dispersion equation, $\zeta_\Phi = 0.2$.

Chapter 4 Snowcover Entrainment

4.1 Introduction

The **RAMMS::Extended** model incorporates detailed input specifications for the **snowcover** to enable accurate simulation of entrainment processes in **dense and powder snow avalanches**.

This chapter outlines the **key parameters**, **input possibilities**, and procedures for defining the **snow cover properties** for a **RAMMS::Extended** calculation. These inputs influence the behavior of the avalanche core, powder cloud, and their interactions with the terrain. More importantly they allow the consideration of different climatic regions, or the investigation of changing climate scenarios.

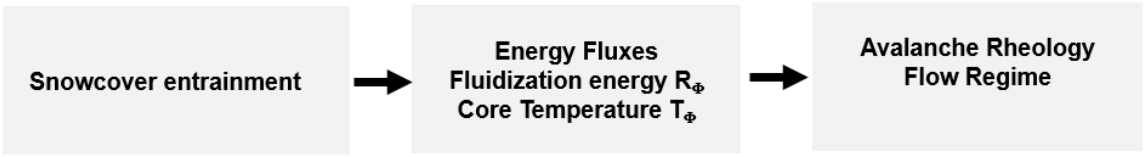


Figure 4.1: Snow entrainment plays a key role in shaping the dynamic processes within the avalanche core, influencing internal energy fluxes and mass exchange between the core and the cloud. By affecting the balance between random kinetic energy and internal thermal energy, entrainment leads to noticeable changes in the flow's rheological properties. This shift in energy pathways impacts frictional behavior and can even trigger transitions between different flow regimes, altering the structure of the avalanche.

Entrainment not only alters **the mass balance** of the avalanche but also impacts **two internal energy fluxes** (see Figure 4.1):

1. **Production of Random Kinetic Energy** (R_Φ): Entrainment contributes to the generation of random kinetic energy, which governs the dispersive motion of snow granules within the avalanche core.

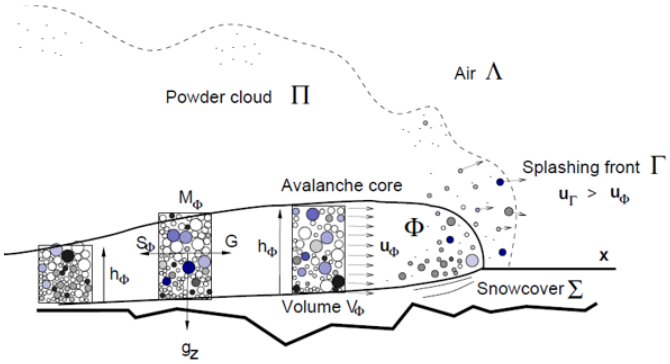
2. **Internal Energy** (T_Φ and E_Φ): Entrainment influences the overall internal energy (E_Φ) of the avalanche core, directly affecting its temperature (T_Φ)and thermal state of the flowing snow.

These relationships—between **entrainment and the production of random energy**, and between **entrainment and avalanche temperature**—are pivotal in shaping the behavior of the avalanche. By dynamically modifying energy fluxes, entrainment can drive transitions in flow regimes, such as shifts from dry to wet avalanches or from flowing to powder avalanches.

4.2 Entrainment Model

In the **RAMMS::Extended** model, the **interaction of the avalanche** with the **snowcover** is modeled as an **elastic/plastic collision** (Figure 4.2).

The **snowcover** is represented by the **symbol Σ**, which not only denotes the snowcover itself (the letter S) but also emphasizes the concept of snow mass being **"added"** to the avalanche (the sigma summation symbol).



Three snowcover variables

h_Σ : **Snowcover height (m)**

ϱ_Σ : **Snowcover density (kg/m³)**

T_Σ : **Snowcover temperature (C)**

At the moment we define only one snowcover layer. But it is possible to define a complicated multilayer snowcover in RAMMS.

Figure 4.2: The avalanche core interacts with the snowcover through a combination of elastic and plastic collision dynamics. The snowcover is represented by the symbol Σ, signifying the addition of snow mass to the avalanche core.

The three fundamental properties of the snowcover are its **height** (h_Σ), **density** (ϱ_Σ), and **temperature** (T_Σ). The **elastic** part of the avalanche/snowcover interactio is manifested as the **splashing front**, where snow particles are ejected upon impact. The **plastic** component involves the **entrainment process**, whereby the static, pre-existing snowcover is accelerated and incorporated into the avalanche, reaching the velocity of the avalanche core.

The **interaction** between the **avalanche** and the **snowcover** occurs **exclusively** between the avalanche core (Φ) and the snowcover (Σ). The powder cloud (presently) does not entrain snow in the **RAMMS::Extended** model.

4.1.1 Erodibility

The concept of **erodibility** defines the mass (M_Σ) involved in the collisional interaction between the avalanche core and the snowcover (Figure 4.4). This mass represents the portion of the snowcover that undergoes a series of dynamic processes, including displacement, granularization, acceleration, mixing, splashing, and eventual entrainment into the avalanche core. Moreover, **the mass M_Σ is the sum** of the **splashed mass** (which

does not belong to the avalanche) and the **entrained mass** (which becomes part of the avalanche). This distinction is necessary to define the thermodynamic boundaries of the avalanche.

The **maximum mass** that can be eroded is constrained by the available snowcover depth (h_Σ) and density (ϱ_Σ). Specifically, $M_\Sigma \leq \varrho_\Sigma \; h_\Sigma$. This equation ensures that the eroded mass cannot exceed the physical limitations of the snowcover at any given point and time.

The rate at which **snowcover mass is eroded** by the avalanche core, is expressed as $\dot{M}_\Sigma$.

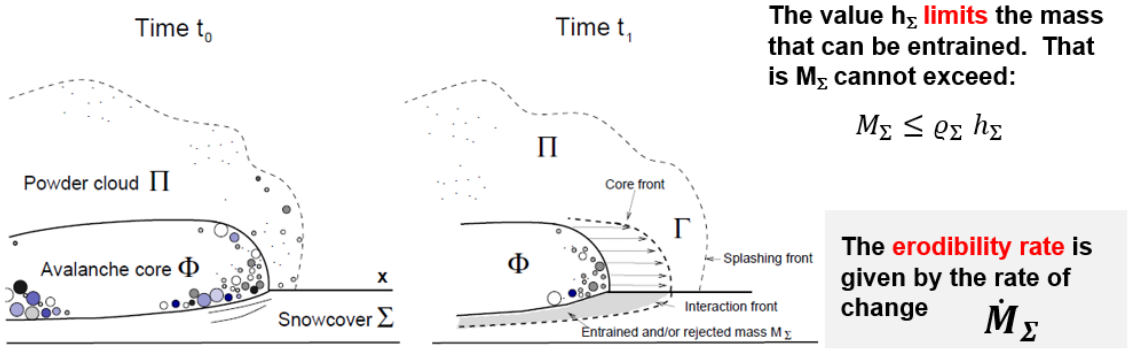


Figure 4.3: The avalanche core moves from location 0 to location 1 from time t_0 to t_1. During this time interval the avalanche erodes snow at the rate $\dot{M}_\Sigma$.

The **rate of snow erosion**, whether rapid or gradual, plays a pivotal role in avalanche dynamics. Erosion can occur through different mechanisms, ranging from **fast frontal erosion** to **slow basal abrasion**, each influenced by various environmental and physical factors.

Key Influencing Factors

1. **Avalanche Velocity:** Higher core velocities lead to greater erosion rates, as the distance travelled by the avalanche is directly related to the core velocity V_Φ.

2. **Snow Temperature (T_Σ):** Colder temperatures result in higher erosion rates due to the reduced bonding strength within the snowpack.

3. **Slope Angle:** Steeper slopes facilitate increased erosion, as gravitational forces amplify the interaction between the avalanche core and the snowcover.

4. **Bonding Effects:** Variations in snowpack cohesion and stability modify the efficiency of the erosion process.

Empirical Relations

The **rate of erosion** has been derived from back-calculations of real avalanche events, **resulting in a general formula that integrates the critical factors** (Figure 4.4):

$$\dot{M}_{\Sigma} = k_0 k_T k_{\psi} \|\vec{u}_{\Phi}\|$$

Where:

- $\dot{M}_{\Sigma}$: Rate of eroded snow mass.
- k_0: Empirical scaling coefficient, $k_0 = 0.005$.
- k_T: Factor accounting for snow temperature.
- k_{ψ}: Factor accounting for slope angle.
- $V_{\Phi} = \|\vec{u}_{\Phi}\|$: Magnitude of the avalanche core velocity.

This formula provides a framework for estimating erosion rates under varying conditions, enabling precise simulations of snowcover interaction.

4.1.2 Entrainment and Splashing

Erodibility mathematically defines the **total mass** involved avalanche-snowcover interaction.

Entrainment refers to the fraction of snow mass that is **incorporated into the avalanche core,** where it becomes part of the flowing system **(plastic collision).**

Splashing describes the mass fraction that is ejected **forward** with velocities exceeding that of the avalanche core, creating a **high-speed, dispersive motion** at the leading edge of the avalanche (elastic collision, creation of splashing front Γ, see Figure 4.5). Splashed mass is mathematically not part of the avalanche core, but **energy is extracted from the avalanche core to create the splashing front**.

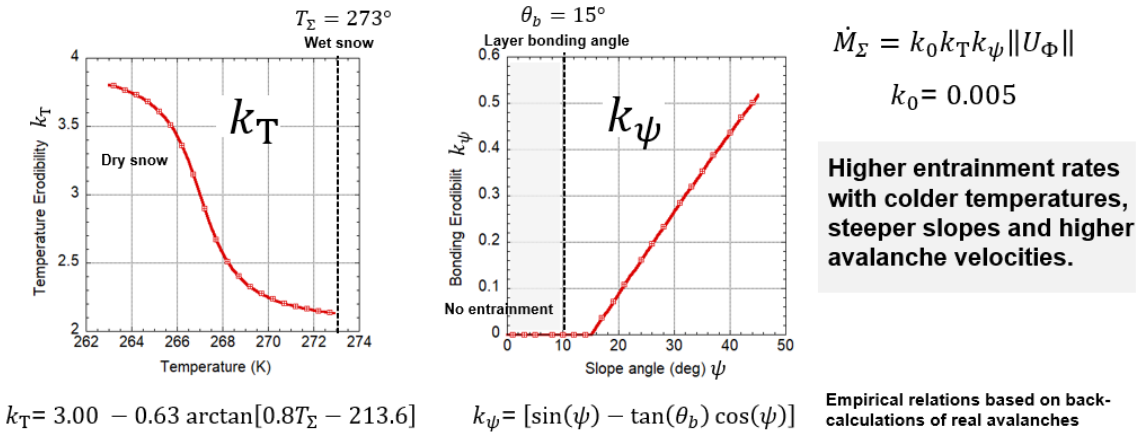


Figure 4.4: Empirical erosion coefficients for three-day new snow.

The **proportion of snow mass allocated to entrainment versus splashing** is governed by the **splashing parameter** $\gamma \leq 1$

$$\dot{M}_\Sigma = \dot{M}_{\Sigma\to\Phi} + \dot{M}_{\Sigma\to\Gamma} = [1-\gamma]\dot{M}_\Sigma + \gamma\dot{M}_\Sigma$$

A higher value of γ would favor splashing, resulting in more forward-projected mass, while a lower γ would emphasize entrainment, incorporating more of the snowcover into the avalanche flow. This balance is crucial for accurately modeling avalanche behavior, as it affects both the mass balance and the energy dynamics of the system, influencing flow **regime transitions** and overall **avalanche mobility**.

In the literature, different mechanisms of snow entrainment during avalanches have been identified. These mechanisms describe how snowcover interacts with the avalanche, influencing the mass balance and flow behavior. The primary mechanisms include **basal erosion**, **frontal entrainment** and **ploughing**. Each mechanism exhibits distinct physical characteristics and is applicable under specific conditions.

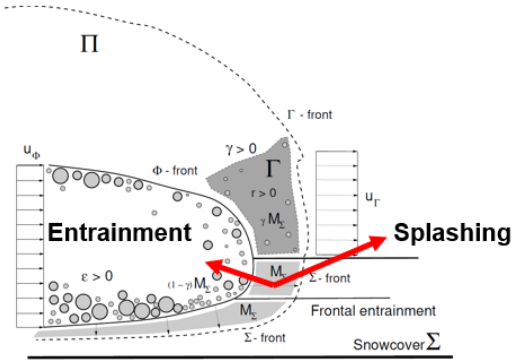
Basal Erosion ($k_0 = 0.001$, $\gamma = 0.0$)

Basal erosion occurs at the basal interface between the avalanche core and the snowcover. This mechanism is characterized by:

- **Scraping and Shearing**: The basal running layer scrapes and shears the snowcover beneath the avalanche.
- **Low Entrainment Rates**: Minimal amounts of snow mass are incorporated into the avalanche.

- **No Splashing**: The snowcover is not ejected forward, but rather smoothly incorporated into the flow.

This mechanism represents steady, low-impact erosion of the snowcover and is ideal for scenarios where basal interaction dominates the avalanche flow



Snow entrainment requires energy!

$$\dot{M}_\Sigma = \dot{M}_{\Sigma\to\Phi} + \dot{M}_{\Sigma\to\Gamma} \quad \textbf{Interaction}$$

$$\dot{M}_{\Sigma\to\Gamma} = \gamma \dot{M}_\Sigma \quad \textbf{Splashing}$$

$$\dot{M}_{\Sigma\to\Phi} = [1-\gamma]\dot{M}_\Sigma \quad \textbf{Entrainment}$$

Energy Loss Avalanche Core

$$\dot{L}_\Sigma = \frac{1}{2}[(1-\gamma) + \gamma(1-r^2)]\,\dot{M}_\Sigma U_\Phi^2$$

r **Elastic part of collision „restitution coefficient“**

Figure 4.5: The mass that is displaced by the avalanche is either entrained or splashed forward. The entrained mass becomes part of the avalanche, the splashed mass becomes part of the splashing front or fluidized pre-front. We calculate the energy lost in both cases.

Ploughing ($k_0 = 0.0, \gamma = 0.0, \ r = 0$)

Ploughing occurs when the avalanche core interacts with the snowcover but does not incorporate snow into its flow. Instead, the snowcover is displaced or pushed aside. Key characteristics include:

- **No Entrainment**: Snow is not added to the avalanche core.
- **Mass Displacement**: The snowcover is pushed aside rather than incorporated.
- No Splashing: Forward motion of snow occurs, but it does not attain a velocity higher than the avalanche.

Ploughing is typically observed in avalanches with dense cores that interact with settled or compacted snowpacks.

Frontal Entrainment ($k_0 = .005, \gamma = 0.2, r = 1$)

Frontal entrainment is a high-energy mechanism where the leading edge of the avalanche core interacts with the snowcover. This is the default mechanism used in hazard mapping calculations and is characterized by:

- **High Entrainment Rate**s: Large quantities of snow are incorporated into the avalanche.
- **Splashing**: Some snow is ejected forward with velocities exceeding that of the avalanche.
- **Dynamic Interaction**: This mechanism dominates in steep release zones or areas with loose, erodible snowcover.

Frontal entrainment is critical for modeling the formation of powder avalanches with splashing/saltation fronts. Presently in the **RAMMS::Extended** model **we allow only for frontal entrainment**. It is not possible to specify lower k_0 values for basal erosion or ploughing. The intended use of RAMMS extended is to simulate hazard scenarios involving critical, new snow conditions. **RAMMS::Extended always assume frontal entrainment because hazard scenarios are based on three-day new snow accumulations**.

Interested users should contact the RAMMS development team if alternative entrainment mechanisms are of interest.

4.1.3 Entrainment and Powder Snow Avalanches

Entrainment is considered a **fully plastic interaction**. However, energy is not entirely dissipated to heat. Instead, a portion of this energy generates **macroscopic granular fluctuations**, which we refer to as random kinetic energy R_Φ.

At the avalanche front, **fluidization energy** is generated through the entrainment process. This energy leads to:

- **Particle dispersion**
- **Particle saltation**
- Facilitates the formation of the **powder cloud**

Just as **shearing in the avalanche core**, produces granular fluctuations, the **plastic interaction with the snowcover** also produces fluctuations. The **entrainment model** in **RAMMS::Extended** is constructed to be consistent with grain flow theory.

This model approach Implies that the entrained particles do not uniformly move at the speed of the avalanche core (V_Φ); rather, **velocity fluctuations** emerge during the entrainment process.

These fluctuations are quantified by the **parameter ε** which defines the amount of **fluidization energy** produced during the avalanche interaction with the snowcover.

Once the erodibility rate is defined, the associated energy dissipation rate is also known,

$$\dot{L}_{\Sigma\to\Phi} = \dot{Q}_{\Sigma\to\Phi} + \dot{P}_{\Sigma\to\Phi} = \frac{1}{2}[1-\gamma]\dot{M}_\Sigma V_\Phi^2$$

The dissipated energy $\dot{L}_{\Sigma\to\Phi}$ is partitioned into heat $\dot{Q}_{\Sigma\to\Phi}$ and fluctuation energy $\dot{P}_{\Sigma\to\Phi}$ by the parameter ε:

$$\dot{Q}_{\Sigma\to\Phi} = [1-\varepsilon]\dot{L}_{\Sigma\to\Phi}$$

$$\dot{P}_{\Sigma\to\Phi} = \varepsilon\dot{L}_{\Sigma\to\Phi}$$

These two terms appear on the right-hand side of the internal energy equations. The parameter ε is thus like the parameter α as it partitions dissipated energy into heat and random kinetic energy. In fact, the structure can be divided into three zones (see Figure 4.6):

The parameter ε defines the amount of fluidization energy created during the entrainment process

$$\dot{L}_{\Sigma\to\Phi} = \tfrac{1}{2}[(1-\gamma)]\ \dot{M}_\Sigma U_\Phi^2 = \dot{Q}_{\Sigma\to\Phi} + \dot{P}_{\Sigma\to\Phi}$$

$\dot{Q}_{\Sigma\to\Phi} = [1-\varepsilon]\dot{L}_{\Sigma\to\Phi}$ **Heat**

$\dot{P}_{\Sigma\to\Phi} = \varepsilon\dot{L}_{\Sigma\to\Phi}$ **Fluctuation energy**

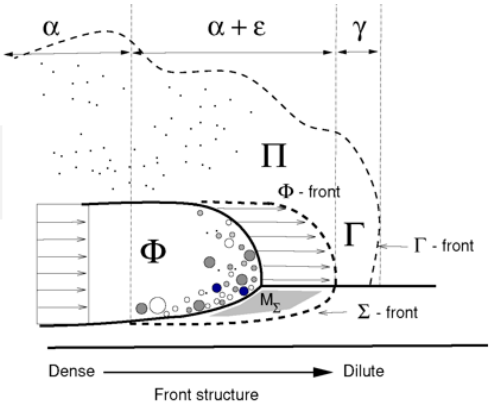


Figure 4.6: Entrainment primarily occurs at the avalanche front. Fluidization energy and heat are generated by the interaction. The avalanche front is divided into the pre-front or splasing front (Γ), the collision/entrainment zone (α + ε) and finally the dense granular shear zone (α).

1. **The Splashing Front** (γ): The leading edge where snow is displaced forward. The particles in the splashing front are not part of the avalanche core, however, energy is extracted from the core to accelerate the particles forward.

2. **Collisional/Entrainment Zone** ($\alpha + \varepsilon$): The region where interaction with the snowcover and granular energy transfer occur. This region is the actual front of the avalanche core.

3. **Dense Granular Shear Zone** (α)**:** The zone following the front, outside of the entrainment zone. In this region, no additional mass is being entrained.

Parameter ε and Its Dependencies

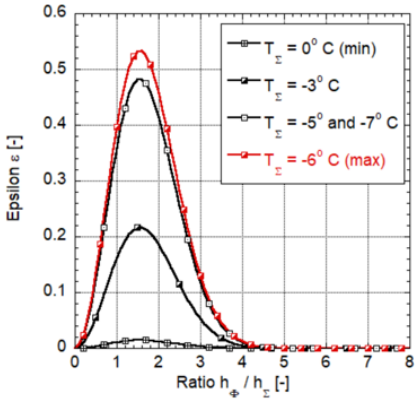
The parameter ε, which **controls the production of fluidization energy**, depends on several geometric and environmental factors:

- **Core Height vs. Snowcover Height** (h_Φ / h_Σ):
    - When $h_\Phi / h_\Sigma > 4$, the **overburden pressure is too high**, suppressing fluidization, and $\varepsilon = 0$. The snowcover depth is too small to contribute to the fluidization of the avalanche core.
    - When $h_\Phi / h_\Sigma < 1$, the avalanche "**drowns**" in the snowcover, and $\varepsilon = 0$. The snowcover is so deep that particles get stuck and disappear in the snowcover.
- **Snowcover Temperature** (T_Σ)**:**
    - Warmer snow temperatures ($T_\Sigma > -3℃$) reduce ε, as warm snow produces less fluidization energy. This leads to the formation of **plug flows** rather than highly fluidized regimes.

We have introduced these functional dependencies using a **Maxwell-Boltzmann type distribution** (Figure 4.7). The Maxwell-Boltzmann distribution covers the two extreme cases, both the effects of too little snow and, alternatively, too much snow.

In **RAMMS::Extended** the height of the avalanche with respect to the height of the snowcover influences the amount of particle dispersion produced and the structure of the avalanche front. This dispersion is **maximized at a snowcover temperature of $T_\Sigma = -6℃$.** Colder temperatures are associated with longer random kinetic energy decay times, thus once the random energy is produced, it remains longer, again facilitating the formation of the powder avalanches at colder temperatures. **Colder temperatures therefore facilitate the development of powder avalanches** by both increasing **the fluidization/dispersion energy** (ε) as well as increasing its **longevity** (β_Φ).

In **channelized terrain** where the avalanche is forced to follow gullies and encounters channel restrictions, **the height of the avalanche flow increases** causing the ratio h_Φ/h_Σ increases with the effect that the value of ε falls, resulting in slower, denser avalanche that restricts the formation of the cloud. **Open slope avalanches running on steep slopes with loose snow** have the tendency to develop powerful air blasts in **RAMMS::Extended**.



1 **When the ratio h_Φ / h_Σ > 4, the overburden pressure is too high, ε = 0.**

2 **When the ratio h_Φ / h_Σ < 1, the avalanche will „drown" in the snowcover, ε = 0.**

3 **When the temperature increases T_Σ > -3C, then ε will decrease. Warm granules produce less fluidization energy, leading to plug flows.**

Figure 4.7: The parameter ε characterizes the production of random kinetic energy during the entrainment process, embodying the chaotic motion and energy fluctuations within the avalanche flow. This parameter adheres to a Maxwell-Boltzmann type distribution function. Such a distribution emphasizes how geometric and dynamic conditions during entrainment determine the dispersion of granular particles, contributing significantly to the development of complex flow regimes and the generation of turbulent powder clouds.

4.2 Setting the Entrainment Parameters

The **snowcover input settings** can be accessed by selecting the **Snowcover Tab** within the **Run Simulation** panel (Figure 4.8). This tab is organized into four distinct sections:

1. **Scenario Reference Altitude**
2. **Erosion Parameters**
3. **Release/Erosion/Deposition Densities**
4. **Release Information**

These sections serve the following purposes:

- **Scenario Reference Altitude:** Specifies the reference altitude for the simulation, which is used to define snow height and temperature gradients.
- **Erosion Parameters:** Provides an input interface for specifying erosion depth, snow steepness, splash behavior, and yield stress.
- **Release/Erosion/Deposition Densities:** Displays release, erosion and deposition densities. Allows the user to select the spatial distribution of density in the model domain.

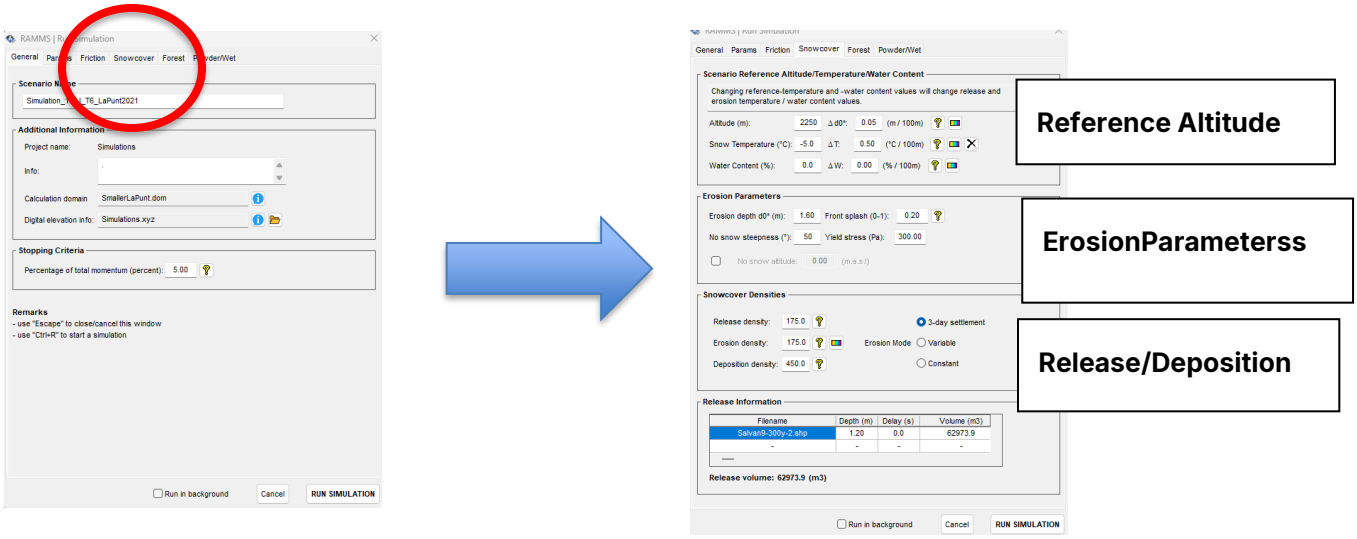


Figure 4.8: The snowcover input settings can be accessed by selecting the **Snowcover** tab within the **Run Simulation** panel**.**

To set the entrainment parameters are its **height** (h_Σ), **density** (ϱ_Σ), and **temperature** (T_Σ) involves a six-step process shown in Figure 4.9.

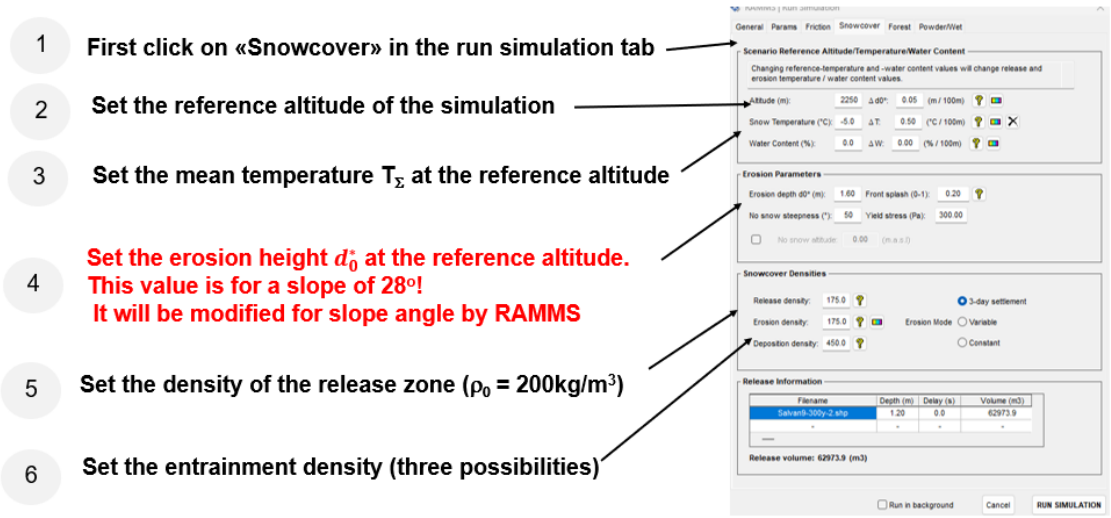


Figure 4.9: Setting the parameters in the snowcover tab. Six steps are required to set the spatial distribution of the snowcover in the solution domain and the entrainment parameters.

4.2.1 Snowcover Reference Altitude

The **Reference Altitude** section contains six input fields (Figure 4.10). The input fields on the right column are used to specify **height**, **temperature** and **water content gradients**. It is possible to immediately visualize the input specifications by clicking on the three colorbars located to the right of the gradient fields (Figure 4.11).

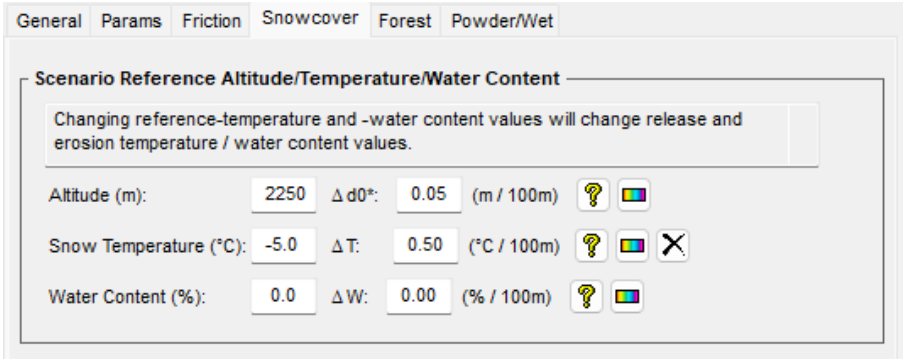


Figure 4.10: This section of the snowcover tab is used to specify the reference altitude and spatial gradients of snowcover height, temperature and water content.

The Six Input Fields

1. **Altitude Z_0 :**
   This value defines the **reference altitude Z_0 for the simulation**. It is typically the altitude of the release zone but can also represent the elevation of a measurement station or the deposition zone. All gradient calculations are based on this reference altitude. For example, if the reference altitude is $Z_0 = 2600$ m, the snow temperature specified in the input box below (e.g., -6.0°C) applies to this elevation.

2. **Snow Temperature T_Σ:**
   Specified in degrees Celsius, this value represents **the mean snow temperature of the snow cover**. Importantly, it is not the surface or air temperature. If the reference altitude corresponds to the release elevation, the snow temperature reflects the average temperature of the snow within the release zone.

3. **Water Content m_Σ:**
   This input defines the percentage of **water content in the snow** at the reference altitude.

4. **Snow Height Gradient Δh_Σ:**
   The term h_Σ represents the height of the erosion layer, and Δh_Σ represents the gradient in snowcover height. Both are defined in the slope-perpendicular direction. This gradient indicates the rate of **decrease** in snow cover height per 100 m of vertical elevation loss. For instance, a value of 0.03 corresponds to a 3 cm reduction in snow cover height for every 100 m drop in elevation.

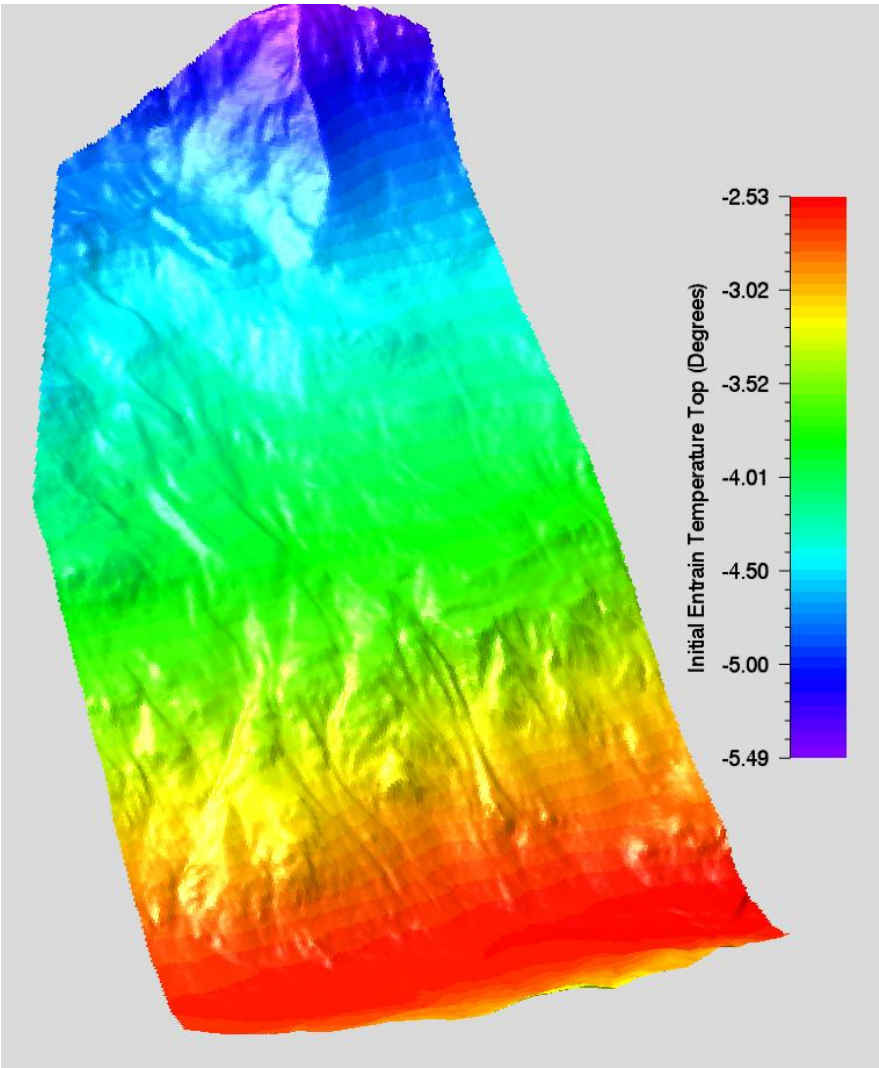


Figure 4.11: Example of the visualization of the specified temperature gradient. Here we specified a reference temperature at 2600m to be $T_\Sigma = -6.00$C and a gradient of $\Delta T_\Sigma = 0.3$C/100m. The visualization is created by hitting the colorbar tab next to the gradient input in the Reference Altitude section of the Snowcover menu.

5. **Temperature Gradient ΔT_Σ:**
   Similar to the snow height gradient, ΔT_Σ represents the **increase** in temperature (in Celsius) per 100 m of vertical elevation loss. For example, a value of 0.30 indicates a temperature rise of 0.3°C for every 100 m descent in elevation. See Figure 4.11.

6. **Water Content Gradient Δm_Σ:**
   This parameter defines the change in snow cover water content per 100 m of vertical elevation loss.

4.2.2 Erosion Parameters

This section of the **Snowcover Tab** (Figure 4.12) focuses on defining the height of the erodible layer and the erosion parameters. This section includes four key input fields:

Erosion Depth, **No Snow Steepness**, **Front Splash**, and **Yield Stress**. Among these, the first two fields, **Erosion Depth** and **No Snow Steepness**—play a crucial role in determining the spatial distribution of snow across the terrain.

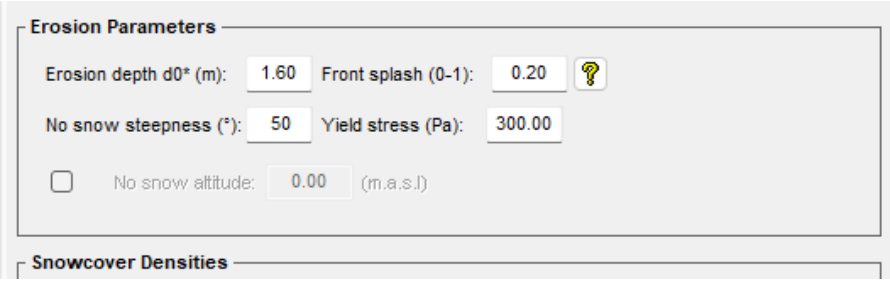


Figure 4.12: Input field for the erosion parameters in the Snowcover tab

4.2.2.1 Erosion Depth

The **Erosion Depth** h_Σ is the snow depth on a **28° slope at the reference altitude** Z_0. The reference altitude is defined in the top section of the Snowcover input tab. In Switzerland the snow depth is given by the three-day new snow accumulation,

$$h_\Sigma = d_0^*$$

To determine the actual snow depth (h_Σ), this value is adjusted using two factors:

1. **Snow Height Gradient (Δh_Σ):** Accounts for variations in snow depth with changes in elevation. In **RAMMS::Extended** we take $\Delta h_\Sigma = \Delta d_0^*$

2. **Slope Steepness Correction $f(\psi)$:** A slope-dependent adjustment based on the formula also applied in the **Swiss guidelines** for avalanche calculations (Figure 4.13). The plot of this formula is shown in Figure 4.14. In many cases the guideline formula can generate too large snow heights, especially if the slopes are windblown. We therefore modified the guideline formula to remove snow on steep slopes (usually located at higher elevations). Both formulas are depicted in Figure 4.13 and plotted in Figure 4.14.

$$h_\Sigma = d_0^* * f(\psi) \qquad f(\psi) = \frac{0.291}{\sin\psi - 0.202\cos\psi}$$

This procedure is in accordance with the Swiss guidelines on avalanche calculations

$$h_\Sigma = d_0^* * f(\psi) \qquad f(\psi) = \frac{0.350}{\tan\psi - 0.202\cos\psi}$$

This formula removes more snow on steep slopes to mimic windblown slopes

Figure 4.13: Slope steepness correction factor $f(\psi)$ (angle ψ). The value of 0.291 is a normalized cohesion value, whereas the value 0.202 is a typical Coulomb friction. The upper formula is the Swiss guideline formula; the lower formula removes more snow from steep slopes.

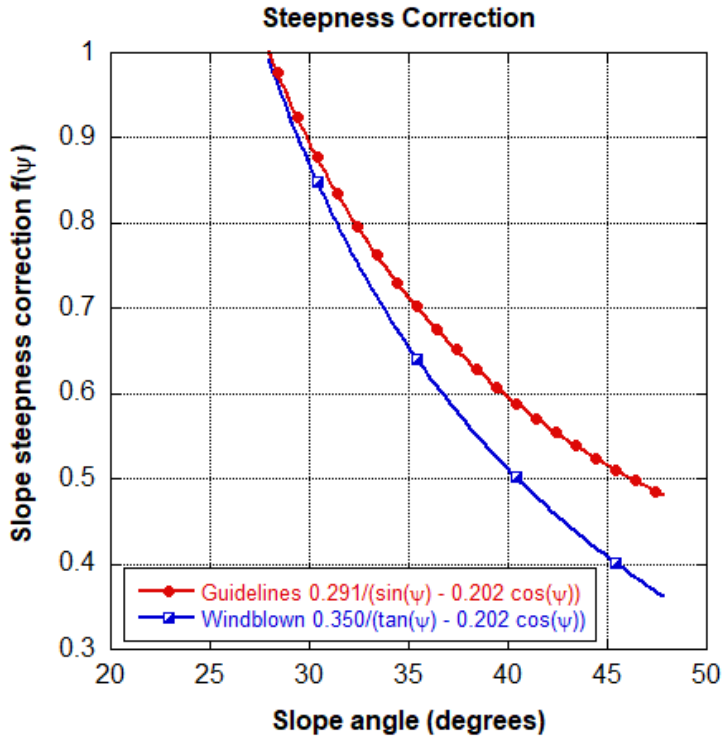


Figure 4.14: Two formulas for the slope steepness factor are implemented in RAMMS::Extended. The first (red line) follows the Swiss guidelines. The second option (windblown) reduces the amount of snow on steep slopes.

The guideline formula is the default value. To use the windblown slope steepness formula, it is necessary to change the GUI preferences. By setting the preference SLOPE_NUM to 0.350, the user selects the windblown formula (Figure 4.15).

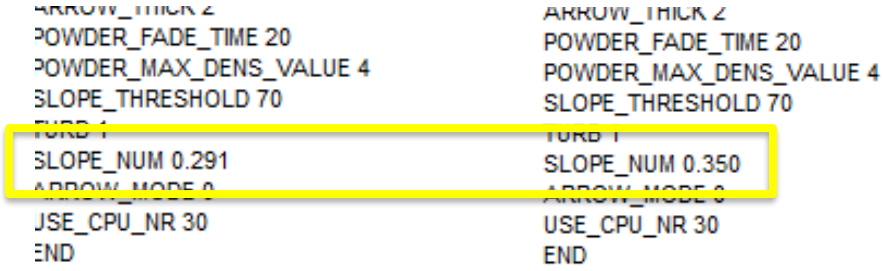


Figure 4.15: In the preferences file it is possible to define which slope steepness correction formula should be applied, either by selecting SLOPE_NUM 0.291 or SLOPE_NUM 0.350.

By combining the elevation gradient and slope steepness corrections, the model generates a complex yet realistic spatial distribution of snowcover, as illustrated in Figure 4.16.

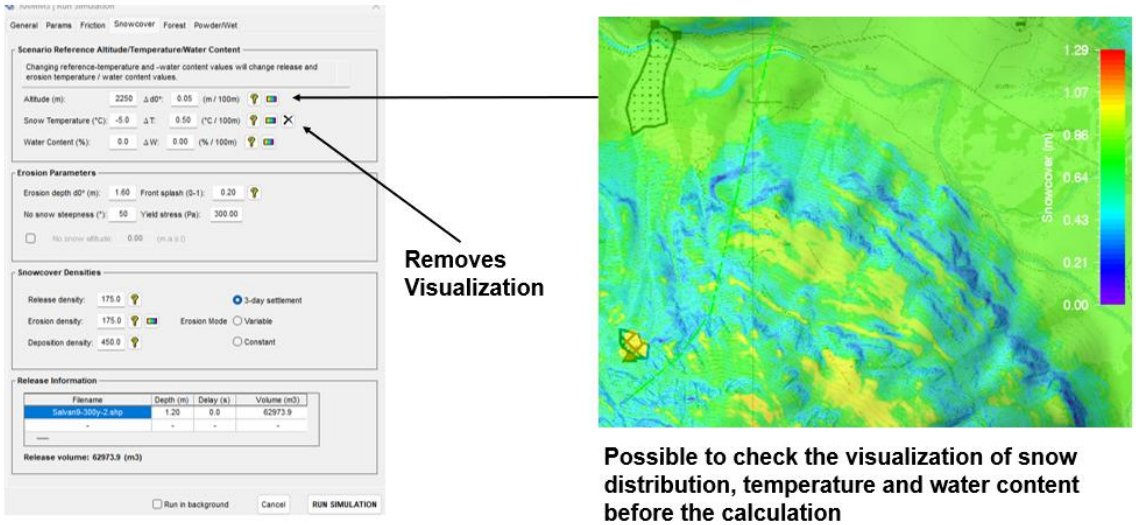


Figure 4.16: Visualizing the entrainment parameters T_Σ, h_Σ and ϱ_Σ. Once the entrainment height ($h_\Sigma = d_0^*$) and elevation gradient ($\Delta h_\Sigma = \Delta d_0^*$) are specified, the spatial distribution of snow can be visualized by selecting the color bars in the upper field. In the example shown, the deep blue areas represent steep, cliff-like regions within the calculation domain where snow does not accumulate.

4.2.2.2 No Snow Steepness

The input field **No Snow Steepness** specifies the critical slope angle beyond which snow cannot accumulate on the surface. This parameter reflects the maximum angle of stability for snowpack retention, with typical values ranging between **50° and 70°**, depending on snow type and surface conditions.

4.2.2.3 Splash

The parameter γ_Γ is inputed in the input field **Splash**. A typical value is $\gamma_\Gamma = \mathbf{0.2\ (20\%).}$ We always assume that the splashed mass is moving at twice the speed of the avalanche $r_\Gamma = 1$.

4.2.2.4 Yield Stress

The yield stress parameter helps control the behavior of snow entrainment and flow dynamics in **RAMMS::Extended** simulations, see Figure 4.17. By adjusting this input, users can fine-tune avalanche interactions with the snowcover, particularly in limiting **secondary flow arms** and reducing. **lateral entrainment** at the avalanche sides.

- The **yield stress parameter sets the minimum stress** required for the avalanche to entrain snow from the snowcover. **If the snowcover's yield strength exceeds the applied stress, no entrainment will occur**.

- **Flow Control**: Yield stress is effective in limiting secondary flow arms from forming, ensuring that the avalanche flow remains concentrated along primary channels.

- **Lateral Entrainment Reduction**: It can also be used to reduce the amount of snow entrained along the lateral edges of the avalanche, which helps to refine the shape and mass of the avalanche flow.

Parameter Range

- Standard Value: The default yield stress used in simulations is **300 Pa**.

- Adjustable Range: Users can specify values between:

    - **0 Pa**: Represents no resistance to entrainment, allowing maximum interaction with the snowcover.

    - **1000 Pa**: Represents high resistance, effectively preventing most entrainment.

Guidelines for Use

1. **Default Settings:** Start with the standard value of 300 Pa for typical hazard mapping applications.

2. **Reducing Entrainment:** Increase the yield stress value (e.g., 600–1000 Pa) if the goal is to limit lateral entrainment or simulate conditions with compacted, harder snowcovers, see Figure 4.17.

3. **Maximizing Entrainment:** Lower the yield stress value (e.g., 0–100 Pa) to represent conditions with weak or loose snowpacks, such as freshly fallen snow, see Figure 4.17.

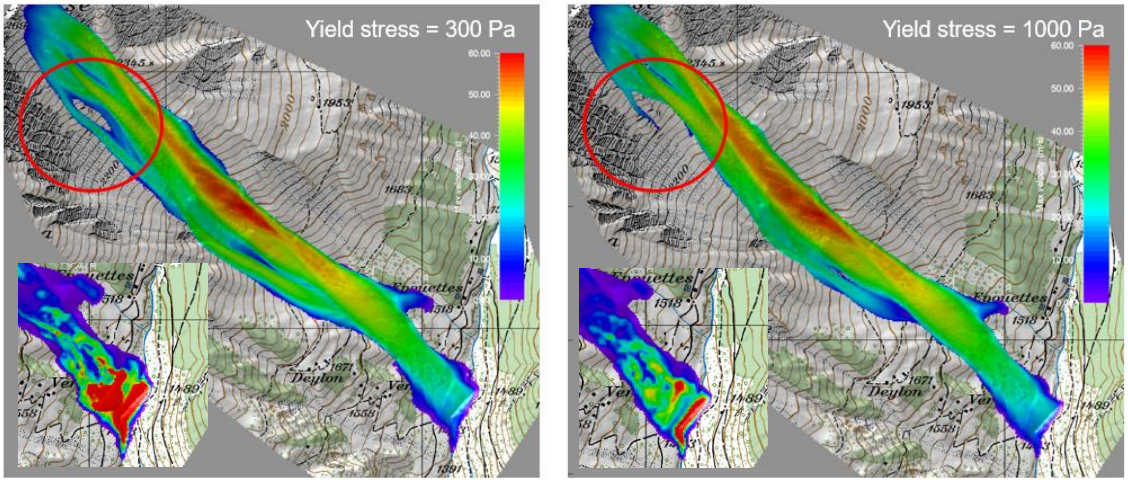


Figure 4.17: The influence on yield stress in a simulation of a Vallée de la Sionne avalanche (2016). Secondary flow arms develop with a low yield stress value (300Pa). Higher values (1000Pa) inhibit the formation of flow arms and reduce the lateral spread of the avalanche in the runout zone.

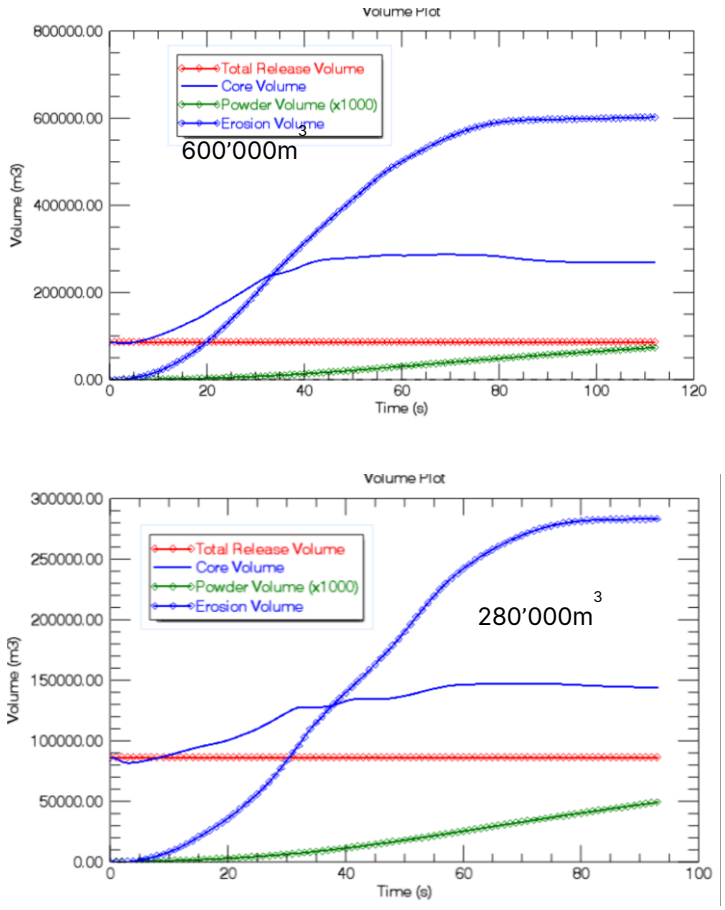


Figure 4.18: The value of yield stress can greatly affect the overall mass balance of the avalanche. With 300Pa the avalanche entrains 600'000m³ of snow; with 1000Pa only 280'000m³.

4.3 Release, Erosion and Deposition Densities

The **release**, **erosion** and **deposition** densities are defined in the third section of the **Snowcover Tab** (Figure 4.19):

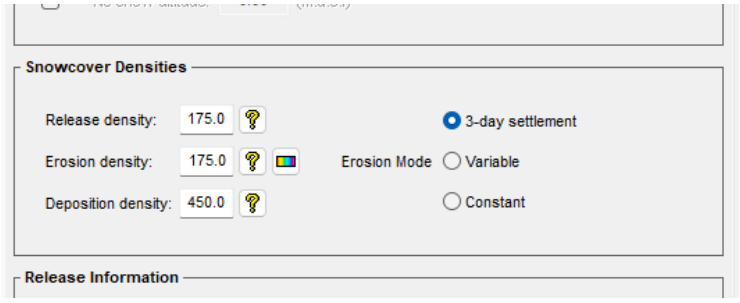


Figure 4.19: The release, erosion and deposition densities of the avalanche are defined in the third section of the snowcover tab.

4.3.1 Release Density

The **release density**, denoted as ϱ_0, represents the **average density of the snow** in the release zone. A commonly used **standard value for release density is $\boldsymbol{\varrho_0 = 200}$ kg/m³**, though variations depending on snow conditions are possible. For colder snow (with temperatures $T_\Sigma \leq$ -7°C), recommended **release densities range between $\boldsymbol{\varrho_0 =}$150 kg/m³ and $\boldsymbol{\varrho_0 =}$175 kg/m³,** reflecting the lighter, less compact nature of cold snow. In contrast, for wetter snow with higher moisture content, release densities can exceed $\boldsymbol{\varrho_0 \geq}$ **300kg/m³**, as the increased water content leads to heavier and more compact snow. Users should select a value that aligns with the specific snow conditions of their scenario to ensure realistic simulations.

4.3.2 Erosion Density

The **RAMMS::Extended** avalanche model provides three methods for specifying snow density, each designed to address different scenarios: **3-Day Snow Settlement**, **Variable Density**, and **Constant Density**. These methods provide flexibility and adaptability to a wide range of snow and terrain conditions.

4.3.2.1 Three-Day Snow Settlement Method

This method is based on experimental measurements and analytical models to determine snow density after a 3-day snowfall, considering the effects of precipitation, slope angle, and temperature. It is particularly suited for applications such as **hazard mapping** and **climate studies** (Figure 4.20).

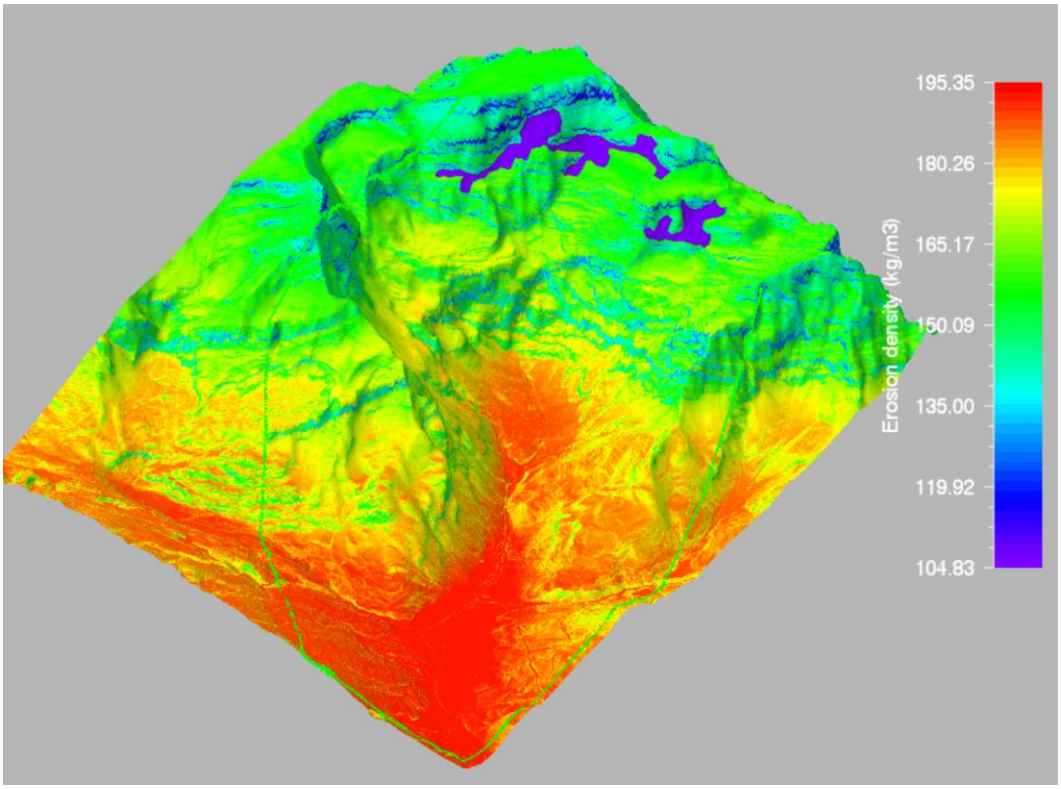


Figure 4.20: The Three-Day Snow Settlement Method provides atool to model hazard scenarios based on snow precipitation. This method enables the calculation of three-day snow heights and densities, tailored to specific climatic conditions and elevations. By incorporating precipitation amounts and settlement dynamics, this approach delivers realistic estimations of snowpack properties over time, making it ideal for hazard assessments and avalanche forecasting in varying environmental contexts.

Key Principles:

- The model utilizes a **settlement equation** derived from the work of **Kojima (1967, 1974) on new snow settlement**:

$$\frac{1}{\varrho_\Sigma(t)}\frac{d\varrho_\Sigma(t)}{dt} = \frac{\sigma}{\eta_c\left(\varrho_\Sigma, T_\Sigma\right)}$$

  Here:

  - $\varrho_\Sigma(t)$: **Snow density** over time.
  - σ: **Overburden stress** from the 3-day snowfall.
  - $\eta_c\left(\varrho_\Sigma, T_\Sigma\right)$: Density/temperature-dependent **compactive viscosity**.

- **Nonlinear Compactive Viscosity Relationship:**
  The **compactive viscosity**, $\eta_c\left(\varrho_\Sigma, T_\Sigma\right)$, is determined by:

$$\eta_c\left(\varrho_\Sigma, T_\Sigma\right) = \eta_{c1} e^{k\rho}$$

  where:

  - η_{c1}**:** A temperature-dependent coefficient:

$$\eta_{c1} = \eta_{c0}\left[10^{\left(\frac{1}{T_\Sigma}-\frac{1}{271}\right)*0.4550}\right]$$

  - k**:** A constant based on experimental results.

  - T_Σ**:** Snow temperature in Kelvin.

Application:

- The model simulates settlement processes to predict snow densities. For example, after a 3-day snowfall with 150–240 mm of snow water equivalent, typical densities range from $\boldsymbol{\varrho_\Sigma}$ **= 150 to 200 kg/m³**, depending on temperature and slope angle.

- For a release area at $\boldsymbol{T_\Sigma}$**= -6°C**, a slope angle of $\boldsymbol{\psi}$ **= 40°**, and a fracture depth (d_0) of 1.2 m, the calculated snow density is $\boldsymbol{\varrho_\Sigma} \approx$ **170 kg/m³**.

- Example calculations are reported in Table 4.1 relating precipitation (in mm water equivalent) to temperature, slope angle and density.

Input			Output	
Precipi-tation (mm)	Tempe-rature (C°)	Slope Angle (°)	Calculated 3-day d_0 (cm)	Calculated Density 3-days (kg/m³)
152	-7	30	98	155
152	-6	30	94	162
152	-5	30	90	170
152	-5	0	85	177
200	-7	30	119	169
200	-6	30	113	176
200	-6	40	118	170
200	-5	30	109	184
200	-5	40	113	177
240	-7	30	135	178
240	-6	30	129	186
240	-5	30	124	194
240	-5	20	121	199
240	-5	10	119	201
240	-5	0	119	202
240	-7	40	140	172
240	-6	40	134	179
240	-5	40	128	187

Table 4.1: Calculated 3-day snow density based on settlement equations (Kojima) for snow temperatures between -5° and -7°C, slope angle in the release areas of 30° and 40° and 85 cm to 140 cm as fracture depths d_0. The densities range between $\varrho_\Sigma = 150$ kg/m³ and $\varrho_\Sigma = 200$kg/m³.

4.3.2.2 Variable Density Method

The **Variable Density** approach offers a simpler, empirical method based primarily on slope angle (Figure 4.21). Unlike the 3-day settlement method, it does not account for temperature effects or elevation gradients, making it suitable for specific scenarios such as **entrainment of old snowcovers** and **wet snow avalanches**.

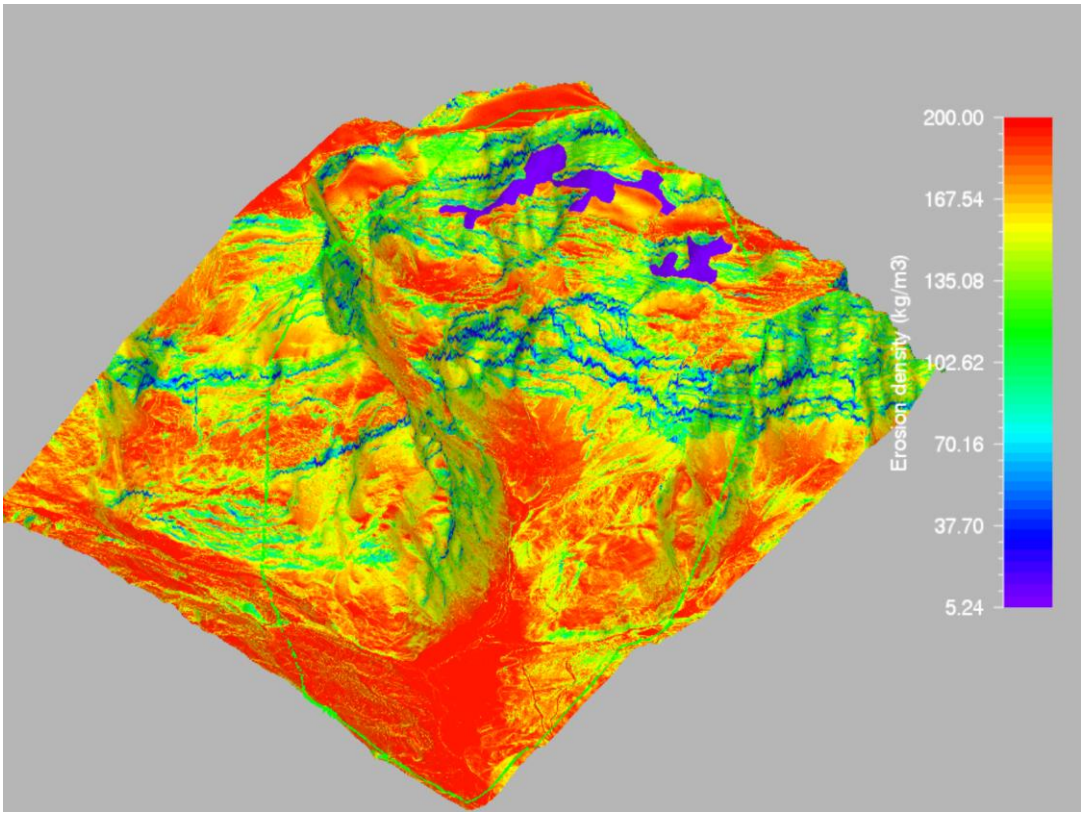


Figure 4.21: The variable densiy method. This method does not consider the effects of temperature or temperature elevation gradients, simplifying its application for wet snow or older, compacted snow layers.

Key Characteristics:

- Relies solely on **empirical relationships between snow density and slope angle**.
- Does **not consider** the effects of **temperature or temperature elevation gradients**, simplifying its application for wet snow or older, compacted snow layers.
- Ideal for **scenarios where real-time snow properties or field measurements are unavailable.**

Application:

- Particularly useful for modeling the entrainment of **old snowcovers** and **wet snow avalanches**.

4.3.2.3 Constant Density Method

The **Constant Density** method provides the simplest approach, where snow density is assumed to remain uniform across the domain (Figure 4.22). It does not account for temperature effects, slope angles, or elevation gradients, making it suitable for basic or specific use cases.

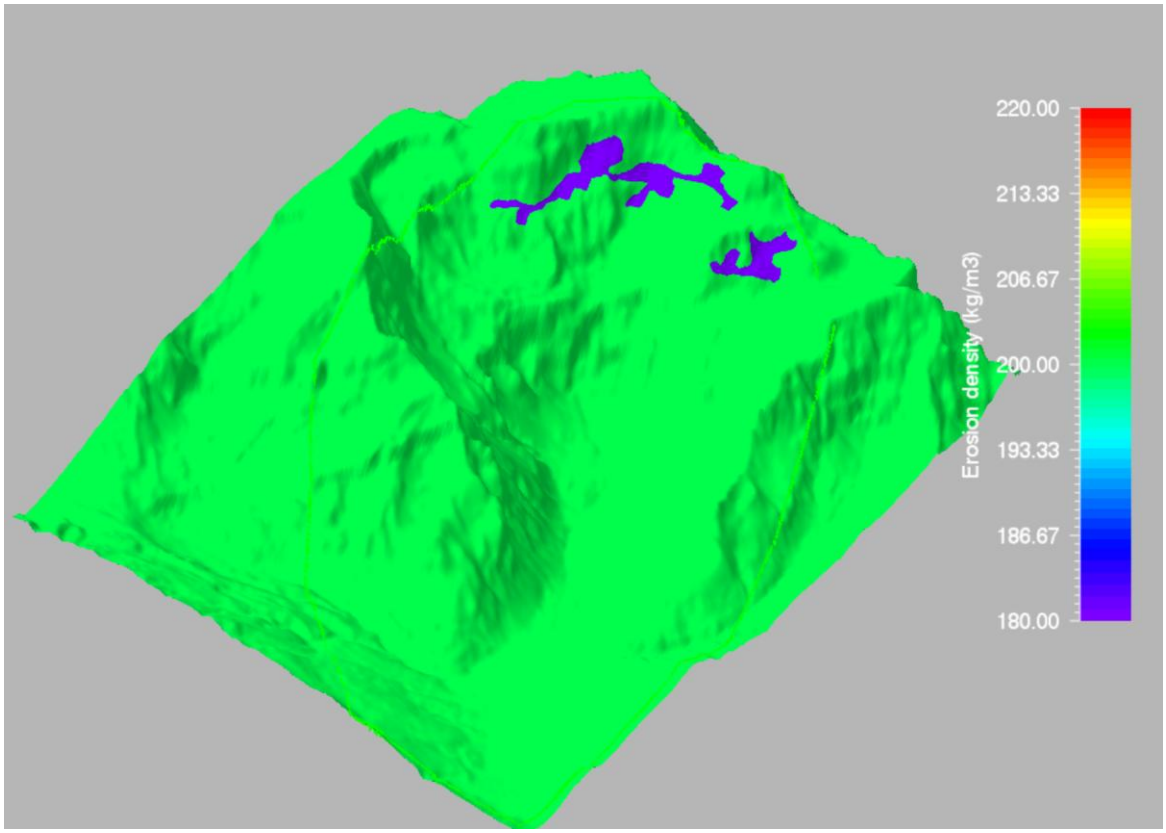


Figure 4.22: The constant densiy method. Green represents a density of $\varrho_\Sigma = 200\text{kg/m}^3$. The non-green colors are avalanche release zones.

Key Characteristics:

- Assumes a constant snow density throughout the simulation.
- Does not include temperature effects or elevation gradients.

Application:

This method is particularly useful for:

- **Old snowcovers**: Ideal for scenarios with well-settled, homogeneous snow layers.

- **Gully avalanches**: Suitable for confined, channelized flow paths.
- **Warm regions** and **lower elevations**: When temperature and elevation gradients are negligible.
- **Debugging**: Provides a simplified snow density framework for testing or troubleshooting simulations.

Summary

Each method offers distinct advantages depending on the application:

- The **3-Day Snow Settlement** method is detailed and incorporates physical processes, making it ideal for forecasting and hazard mapping.
- The **Variable Density** method is a more generalized approach, suitable for entrainment modeling and wet snow avalanches.
- The **Constant Density** method provides the simplest configuration, tailored for scenarios with uniform snow conditions or debugging.

These methodologies ensure that the **RAMMS::Extended model remains versatile and effective across diverse snow and terrain conditions**.

4.3.3 Deposition Density (= Co-volume Density $\hat{\rho}_{\Phi}$)

The deposition density, denoted as $\hat{\rho}_{\Phi}$, is the density **associated with the densest packing of snow granules** after the avalanche comes to rest. A standard value for **deposition density** is $\hat{\boldsymbol{\rho}}_{\boldsymbol{\Phi}} = 450\ \mathbf{kg/m^3}$**.** This parameter serves a dual purpose: it represents the core co-volume density (refer to the theoretical framework) and is used to calculate the compacted state of the avalanche core.

In the simulation's output mode, the deposition density also influences how snow height metrics are interpreted:

- **Core Height:** Refers to the height of the dispersed avalanche core, reflecting its natural, less compact state during flow.
- **Flow Height:** Represents the height of the avalanche core when compacted to the co-volume density, offering a more realistic depiction of the snow’s final, densely packed state upon deposition.

4.4 Visualization of Volume and Mass Balance

Entrainment plays a critical role in shaping the volume and mass balance of an avalanche. While the initial and final volumes of an avalanche can often be approximately measured using tools like drones, determining the mass is more challenging due to the often-unknown snow density. Density measurements are usually limited to a few specific locations, if they are taken at all, and accurate data on release densities is extremely rare. **Volume and mass balances are essential for constraining the return period and play a key role in selecting appropriate friction parameters.**

RAMMS::Extended offers advanced visualization tools to capture and depict the temporal and spatial evolution of avalanche volume and mass. These tools provide critical insights into the dynamic processes shaping the avalanche from its release to final deposition.

Information about the volume and mass balance can be readily accessed via the **Volume** and **Mass** tabs on the display console, conveniently located on the right-hand side of the GUI (see Figure 4.23)

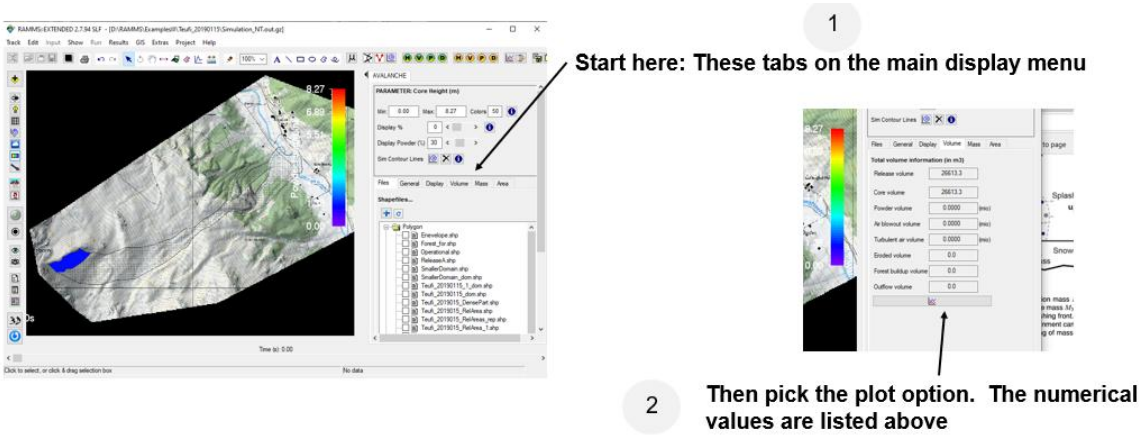


Figure 4.23: On the right-hand side of the main GUI window, two tabs are labeled Volume and Mass. When these tabs are selected the numerial values for the time step (set by the time slider) are given. By picking the plot option, the total volume and mass evolution of the avalanche can be depicted.

Numerical values for the entire avalanche are displayed at the time specified by the time/step slider located below the main window. These values automatically update as the slider is adjusted. (Recall the number of steps is defined by "Dump Step(s)" in the Parameter input tab. At the bottom of each tab, a **plot button** enables users to **visualize the evolution of these parameters over time**. Typical plots for both volume (Figure 4.24) and mass (Figure 4.25) are provided.

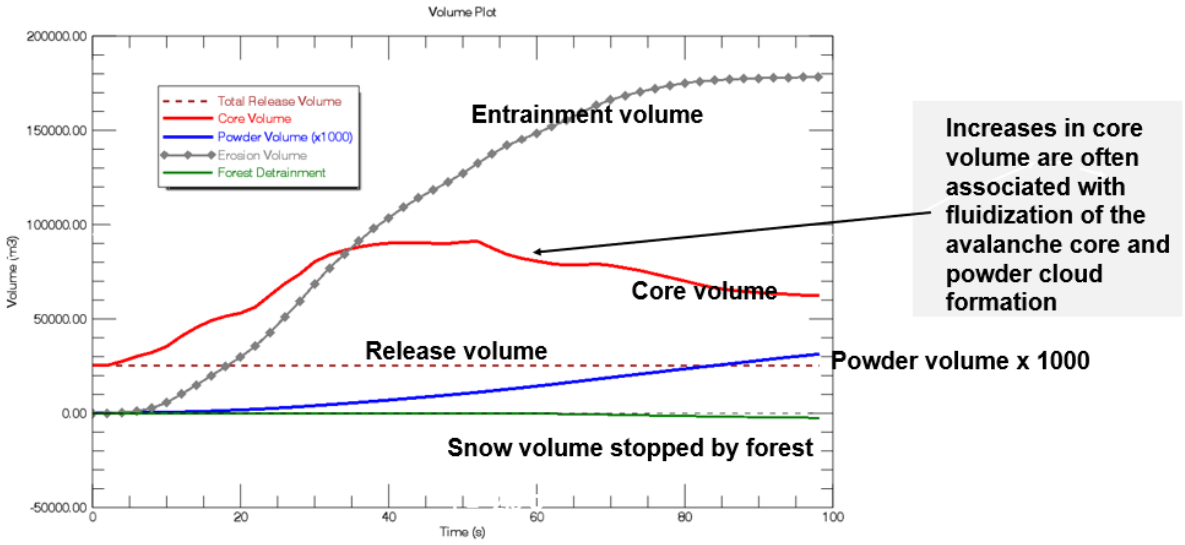


Figure 4.24: RAMMS::Extended volume plot.

The volume plot provides a detailed depiction of the time evolution of various components of the avalanche, including:

1. **Core Volume:** This represents the moving avalanche core as it evolves.

2. **Erosion Volume:** The snow volume entrained from the snowcover during the avalanche's progression.

3. **Forest Stop Volume:** Snow intercepted and halted by forested areas.

4. **Powder Cloud Volume:** The vast volume of the powder cloud, which is significantly larger than the core volume. Note that the powder cloud volume displayed must be multiplied by 1000 for accuracy.

This plot offers a dynamic representation of how the avalanche interacts with its environment over time, providing a clear breakdown of volume changes in different processes.

The mass plot is divided into two key visualization windows:

1. **Upper Window – Moving Mass:** This illustrates the mass of the avalanche core, powder, and entrained snow as they evolve over time. The stop criterion for the avalanche's calculation, as defined in the "Params" tab, directly affects these dynamics.

2. **Lower Window – Evolution of avalanche mass components.** The red line with diamonds provides the total mass of the avalanche. If the avalanche leaves the calculation domain, the total mass can decrease.

The plot includes the following elements:

- **Core Mass (solid blue line):** Represents the mass of the dense avalanche core, which can decrease as mass is transferred to the powder cloud or stopped by obstacles such as forests.

- **Powder Mass (solid green line):** Depicts the mass of the powder cloud, which grows as mass is ejected from the core.

- **MassToPowder (green diamonds):** Tracks the mass transferred from the core to the powder cloud. This value is negative, indicating mass loss from the core and gain by the powder cloud.

- **Forest Mass (red line):** Represents the snow mass stopped by the forest. This information can be used to assess the effectiveness of forests in mitigating avalanche impacts.

- **Water Mass (green diamonds):** Illustrates the amount of water mass within the avalanche, allowing users to evaluate the transition of the avalanche from a dry, dense flow to a wet snow avalanche as water content increases.

Key Observations

- **Initial Release Metrics**: Both the volume and mass plots clearly display the initial release volume and mass, offering a reference point for all subsequent calculations.

- **Core vs. Entrained Volume and Mass**: While the entrained volume may surpass the core volume due to the expansive erosion of snowcover, the same is not true for mass. The disparity between snowcover density and core density is the primary reason for this difference.

By visualizing these plots, users can gain a holistic understanding of the avalanche's behavior, including its interaction with the terrain and snowcover. These insights are crucial for hazard assessment and mitigation planning, enabling informed decision-making based on detailed volume and mass dynamics.

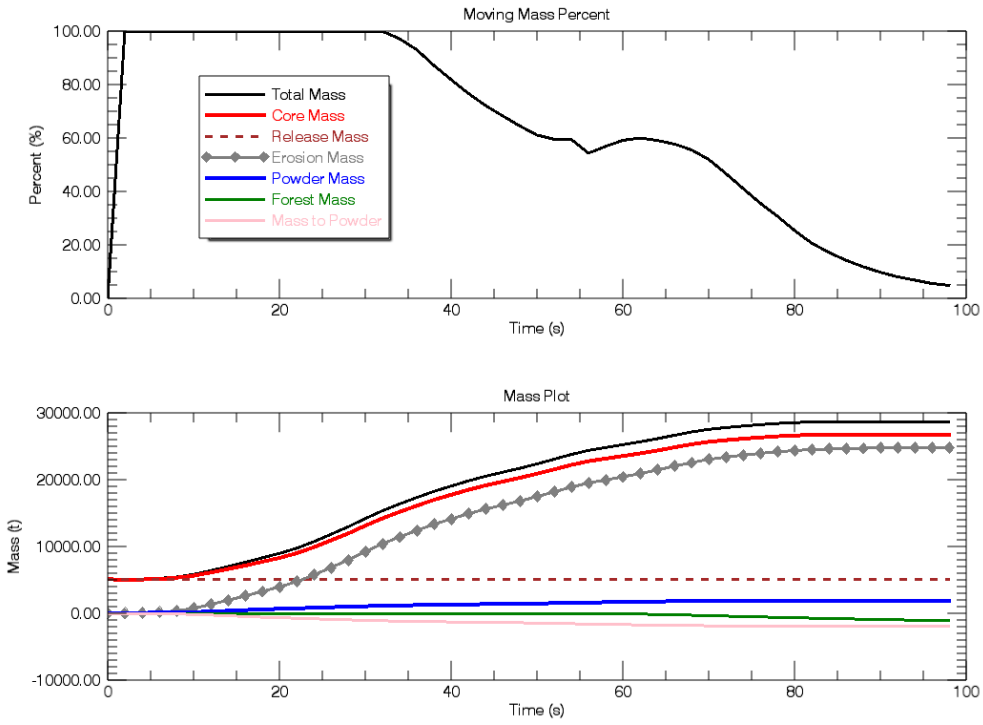


Figure 4.25: RAMMS::Extended mass plot. The upper plot depicts the moving mass. The lower plot provides (in tons) the mass evolution of the avalanche. The difference between the total mass (red line with diamonds) and the core mass (solid blue line), is the mass in the powder cloud.

Chapter 5 The Powder Cloud

5.1 Introduction

The powder cloud in an avalanche behaves as a turbulent, inertial flow of fine ice particles suspended in air, forming what is known as a **suspension current**. This airborne cloud originates from the initial momentum imparted by the avalanche core (Φ), which consists of a dense, granular shear flow of snow clods. **RAMMS::Extended** is designed to predict the full spatial evolution of a powder cloud, providing critical insights into its **extent, propagation dynamics, and associated air-blast pressure effects** (Figure 5.1).

The **exchange of mass and momentum between the core and the suspension cloud is governed by the transfer term** $\dot{M}_{\Phi\to\Pi}$, with the most significant interaction occurring at the leading edge of the avalanche. The formation and evolution of the cloud are intrinsically linked to the fluidized state of the core, where snow entrainment plays a critical role in feeding and sustaining the suspension.

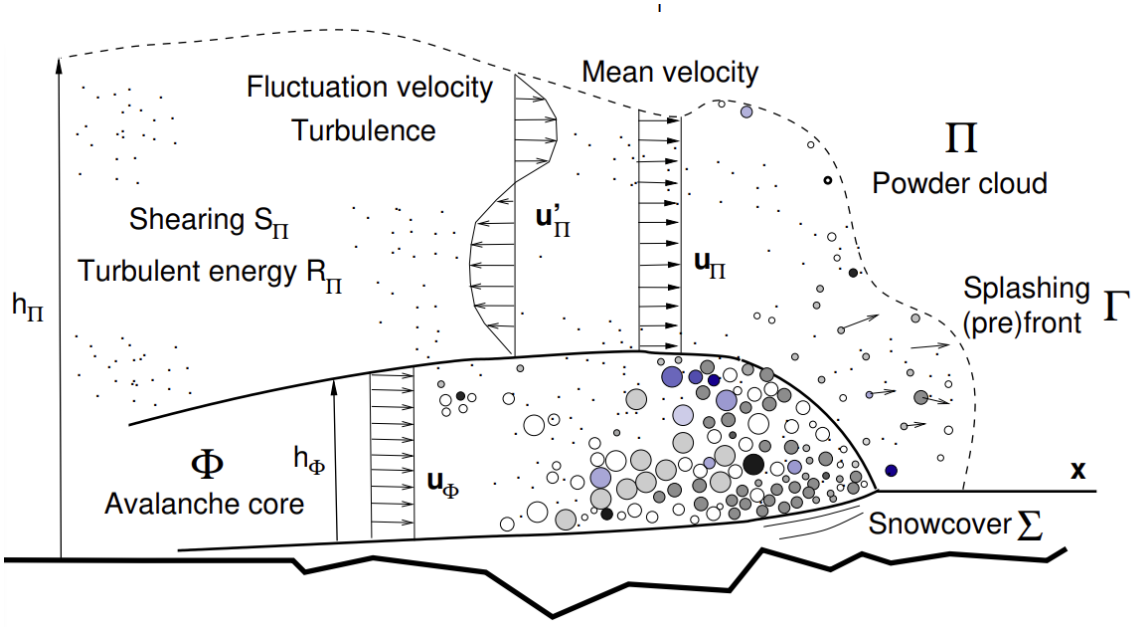


Figure 5.1: Powder cloud dynamics. RAMMS::Extended calculates the mean velocity of the cloud $\vec{u}_\Pi$, coupled with velocity fluctuations $\vec{u}'_\Pi$. The kinetic energy associated with the fluctuations is R_Π.

Recall from Chapter 1 that the powder cloud motion is described by four partial differential equations describing the **mass**, **volume**, **momentum** and **turbulent energy**. The four primary unknowns in this formulation are the **co-volume height** $\hat{h}_{\Pi}$, (mass = $\hat{\rho}_{\Pi}\hat{h}_{\Pi}$), **the cloud height** h_{Π}**, velocity** $\vec{u}_{\Pi}$ and the **kinetic energy associated with turbulent fluctuations** R_{Π}, which collectively characterize the evolving behavior of the flow:

$$\left(\hat{h}_{\Pi}\right)_t + div\left(\hat{h}_{\Pi}\vec{u}_{\Phi}\right) = \dot{M}_{\Phi\to\Pi}$$

$$(h_{\Pi})_t + div(h_{\Pi}\vec{u}_{\Pi}) = \frac{(\rho_i-\hat{\rho}_{\Pi})}{(\rho_i-\rho_{\Lambda})}\dot{M}_{\Phi\to\Pi} + \dot{M}_{\Lambda\to\Pi}$$

$$\left(\hat{h}_{\Pi}\vec{u}_{\Pi}\right)_t + div\left(\hat{h}_{\Pi}\vec{u}_{\Pi}\otimes\vec{u}_{\Pi} + p_{\Pi}I\right) = \dot{M}_{\Phi\to\Pi}\vec{u}_{\Phi} + \frac{(\hat{\rho}_{\Pi}-\rho_{\Lambda})}{\hat{\rho}_{\Pi}}\vec{G}\hat{h}_{\Pi} - \frac{\vec{u}_{\Pi}}{\|\vec{u}_{\Pi}\|}S_{\Pi}$$

$$\left(\hat{h}_{\Pi}R_{\Pi}\right)_t + div\left(R_{\Pi}\hat{h}_{\Pi}\vec{u}_{\Pi}\right) = \dot{M}_{\Phi\to\Pi}R_{\Phi} + [\hat{\rho}_{\Pi}S_{\Pi}]\|\vec{u}_{\Pi}\| + \frac{1}{2}\rho_{\Lambda}\dot{M}_{\Lambda\to\Pi}u_{\Pi}^2 - \beta_{\Pi}\hat{h}_{\Pi}R_{\Pi}$$

The dominant factor is the **momentum imparted by the core to the cloud** $\dot{M}_{\Phi\to\Pi}\vec{u}_{\Phi}$, which exceeds the acceleration due to gravity,

$$\dot{M}_{\Phi\to\Pi}\vec{u}_{\Phi} \gg \frac{(\hat{\rho}_{\Pi}-\rho_{\Lambda})}{\hat{\rho}_{\Pi}}\vec{G}\hat{h}_{\Pi}$$

This **imbalance shapes the longitudinal structure of the cloud**, producing distinct velocity gradients from the leading edge to the wake. As a result, the cloud maintains forward momentum with little lateral spreading.

Once the ice particles become airborne, however, the **cloud's behavior decouples from the core**, evolving independently to a flow containing turbulent structures. Unlike the dense core, which remains in contact with the ground and interacts with surface roughness, **the suspension cloud is largely unaffected by terrain variations**.

In the **RAMMS::Extended** module there are four parameters that can be inputted by the user to control the behavior of the powder cloud. These are:

1. Cloud drag S_{Π}
2. Turbulent decay β_{Π}
3. Cliff factor for $\dot{M}_{\Phi\to\Pi}$

4. Turbulent air entrainment $\dot{M}_{\Lambda\to\Pi}$

These values are all input in the **Powder/Wet** input tab. There are four primary output results that can be visualized over time and space: the powder cloud velocity $\vec{u}_{\Pi}$, height h_{Π}, density ρ_{Π} and impact pressure p_{Π}. These are described in the sections below.

5.2 Input: Cloud Drag

The input parameter **Cloud drag** controls the **flow resistance** experienced by the powder cloud. **Cloud drag** is set in the **Powder/Wet input tab** (Figure 5.2).

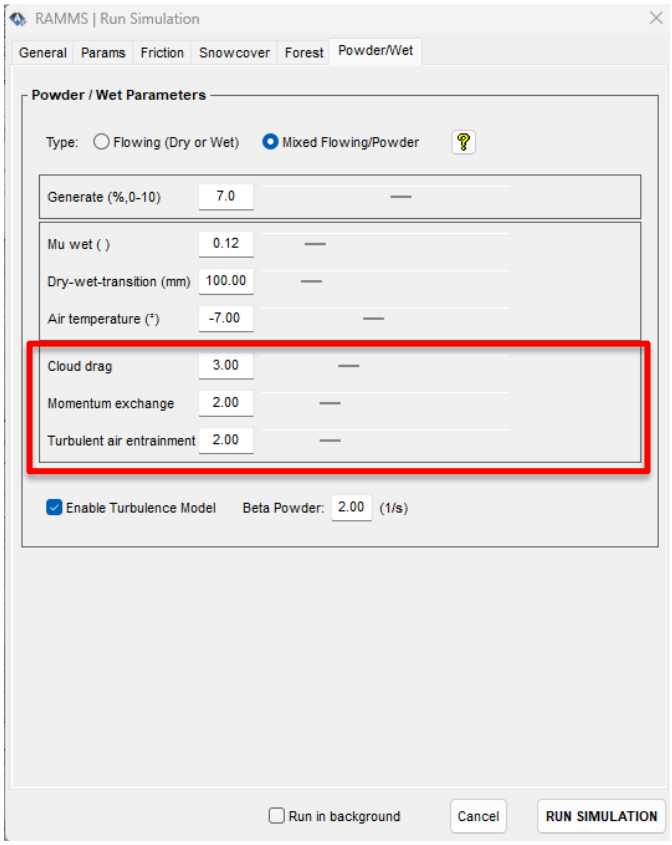


Figure 5.2: Cloud drag is set in the Powder/Wet tab.

In **RAMMS::Extended**, the **powder cloud drag** S_{Π} is modeled as a combination of **laminar and turbulent contributions**, reflecting the dual nature of resistance encountered by the suspension cloud:

$$S_{\Pi} = \mu_L \|\vec{u}_{\Pi}\| + \mu_T R_{\Pi} \hat{h}_{\Pi}$$

where μ_L represents the **laminar viscous drag**, and μ_T denotes the **turbulent viscosity**. The transition between these two regimes is governed by the state of turbulence within the cloud.

Crucially, both $\mu_L$and μ_T are determined by a single **dimensionless input quantity, "Cloud Drag,"** encapsulating the overall resistance experienced by the cloud (Figure 5.3). This unified parameter is itself a function of the cloud's **mean velocity and total turbulent energy**, meaning that a single value effectively governs the interplay between laminar and turbulent contributions.

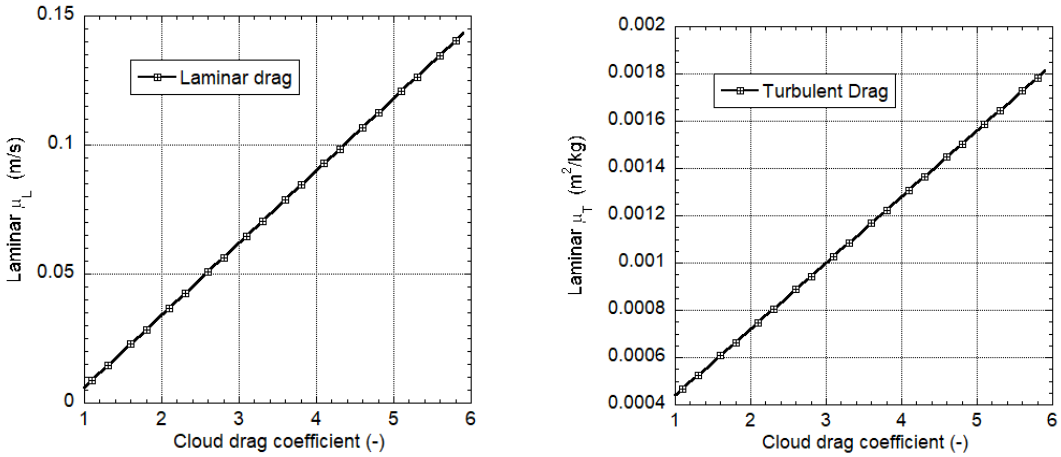


Figure 5.3: The values of $\mu_L$and μ_T as a function of the Cloud Drag input coefficient. Both values increase linearly with the cloud drag coefficient. The recommended value of cloud drag for medium to large powder avalances is CloudDrag = 3.0. The laminar and turbulent drag coefficients are given by $\boldsymbol{\mu_L = 0.28\delta - 0.22}$ and $\boldsymbol{\mu_T = 0.08[0.02 + 0.035\delta]}$ where $\boldsymbol{\delta}$ = **Cloud Drag**.

The linear relationships

$$\mu_L = 0.28\delta - 0.22$$

$$\mu_T = 0.08[0.02 + 0.035\delta]$$

of the laminar and turbulent viscosities and cloud drag δ were determined by examining tree destruction by powder avalanches (Zhuang et al., 2024).

5.3 Input: Turbulence (Decay β_Π)

The precise magnitude of cloud drag is further modulated by the **turbulence decay parameter,** β_Π, which controls the rate at which turbulent energy dissipates over time and space. This value is set in the **Powder/Wet input** tab (Figure 5.4).

Note that the turbulent model does not have to be used in **RAMMS::Extended**. If the **Enable Turbulence Model** is NOT ACTIVATED**, RAMMS::Extended** will apply an empirical drag model utilizing only the cloud velocity. When the **Enable Turbulence Model** is ACTIVATED it is necessary to input a β_Π value. The recommended values are provided in Table 5.1. The advantage of activating the turbulence model is that turbulent powder pressures are subsequently calculated and output in the powder pressure output.

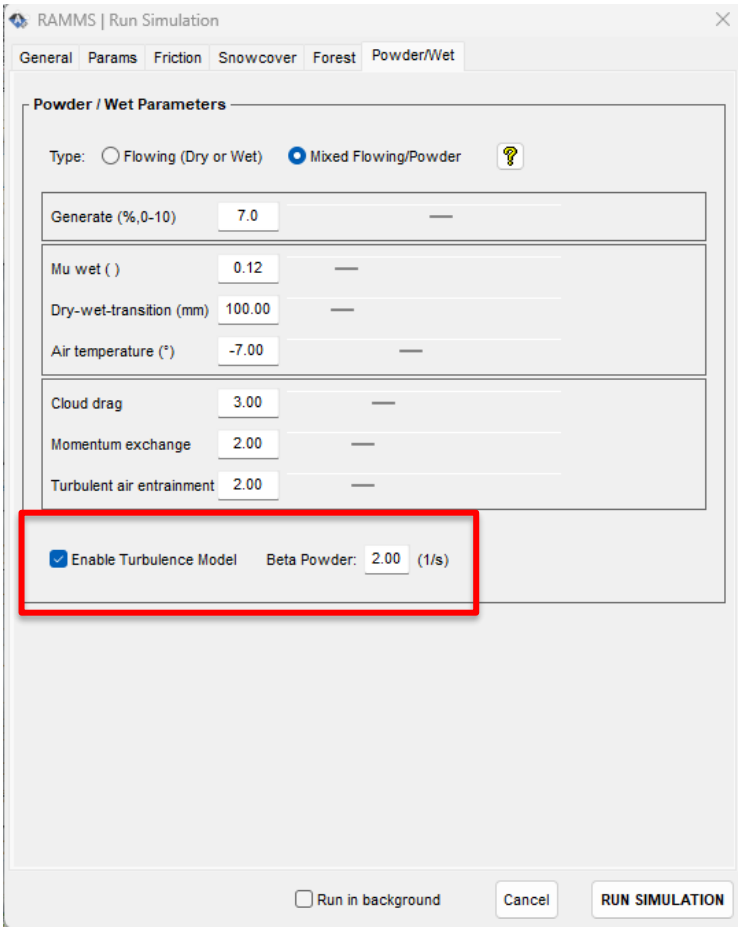


Figure 5.4: The **turbulent decay parameter** β_Π is set in the Powder/Wet tab. First the **Enable Turbulence Model must be turned ON**. Recommend values for β_Π are listed in Table 4.1.

In conditions where turbulence dominates, **drag is primarily influenced by turbulent kinetic energy** R_Π, which enhances resistance (recall that there are three sources of R_Π –

production by internal shearing, production by air entrainment and the production of turbulence during the transfer of mass between the core and cloud). Importantly, as turbulence gradually decays, the flow enters a **laminar regime**, characterized by a smoother, more uniform motion. In this phase, drag is significantly reduced and becomes a direct function of the **avalanche velocity** $\|\vec{u}_{\Pi}\|$, allowing the powder cloud to propagate with minimal resistance (Figure 5.5). This formulation captures how turbulence governs resistance at high speeds while enabling **long-range, low-drag transport** once turbulence has dissipated.

Variable	Definition	Recommended Range
$S_{\Pi}(\delta_{\Pi})$	Cloud Drag **Powder/Wet Tab**	δ_{Π} =3.0 (**S, M** and **L**) δ_{Π} =4.0 (**T**) $2.0 \leq \delta_{\Pi} \leq 5.0$ (Range)
β_{Π}	Turbulence decay cloud **Powder/Wet Tab**	β_{Π} =2.0/s (M and **L**) β_{Π} =1.0/s (**T** and **S**) $0.5 \leq \beta_{\Pi} \leq 3.0$

Table 5.1: Recommended values of CloudDrag $S_{\Pi}(\delta_{\Pi})$ and turbulent decay β_{Π}.

This shift in drag characteristics has profound implications for cloud propagation. In the laminar regime, with drag at a minimum, the powder cloud can persist over considerable distances, carried forward by its residual momentum. However, in the absence of significant turbulent structures, these **laminar-phase winds** possess little destructive capacity, spreading gently rather than impacting terrain with force. Thus, the evolution of cloud velocity and structure is intrinsically tied to the interplay of drag forces, with turbulence acting as both the primary driver of resistance at high speeds and the mechanism that, upon dissipation, allows for extended yet benign propagation at lower speeds.

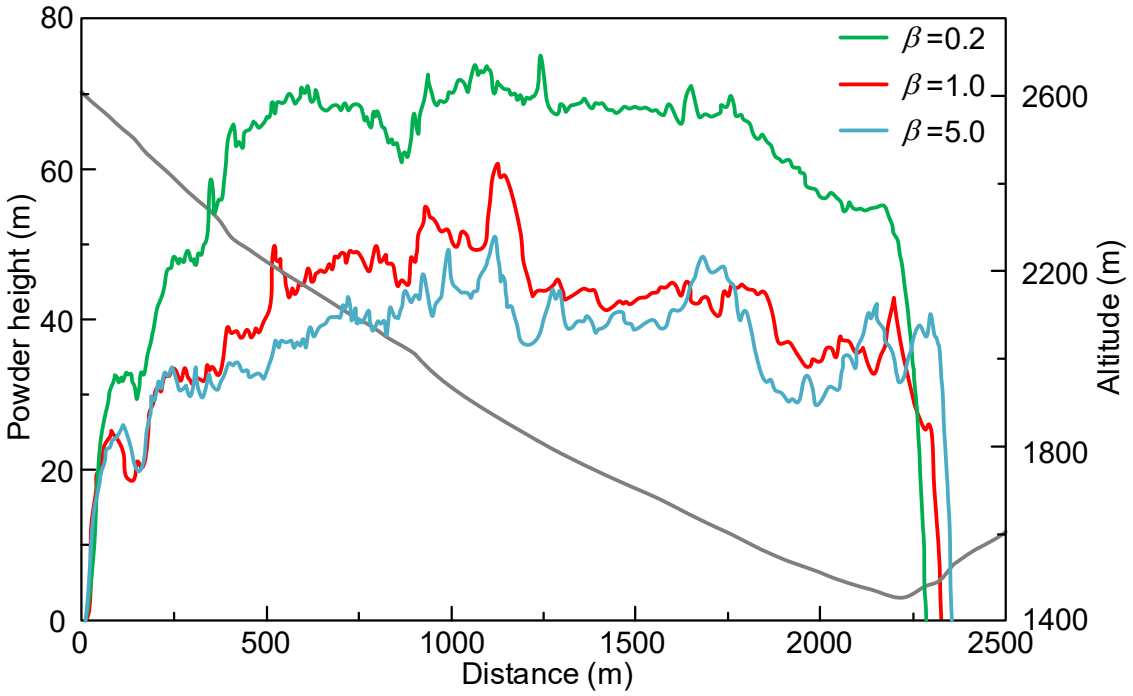


Figure 5.5: How the turbulent decay parameter β_{Π} influences the maximum calculated powder cloud height. Low rates of turbulent decay (β_{Π}=0.2) produce higher powder clouds due to higher air entrainment rates. Example: Vallèe de la Sionne avalanche no. 628.

5.4 Input: Plunging Avalanches and the Cliff Factor

When an avalanche descends a **steep cliff**, there is a **significant increase in the mass and momentum transfer from the avalanche core to the powder cloud**. As the core plunges downward, it undergoes two critical processes: dispersion as it falls over the cliff and compaction when the slope flattens. Both mechanisms contribute to an increased mass exchange rate, $\dot{M}_{\Phi\rightarrow\Pi}$, which plays a crucial role in shaping the dynamics of plunging powder-snow avalanches.

To account for the effects of steep drops, the extended module enhances the mass transfer from the core to the cloud, adjusting it to $\dot{M}'_{\Phi\rightarrow\Pi}$,

$$\dot{M}'_{\Phi\rightarrow\Pi} = \lambda(\theta) * \dot{M}_{\Phi\rightarrow\Pi}$$

This adjustment is particularly significant at steeper slope angles θ. For instance, when the slope reaches $\theta = \theta_0 = 35°$, the exchange of mass—and consequently, momentum—from the core to the cloud increases noticeably. At $\theta = \theta_{max} = 55°$, this transfer reaches its peak, determined by the input value λ_{Π}, a key parameter controlling the extent of mass exchange (Figure 5.6Figure 5.6).

$$\lambda(\theta) = \begin{cases} 1 & \theta \leq \theta_0 \\ \left(\frac{\theta - \theta_0}{\theta_{max} - \theta_0}\right)\theta & \theta_0 < \theta < \theta_{max} \\ \lambda_\Pi & \theta \geq \theta_{max} \end{cases}$$

The cliff factor λ_Π is specified in the "**Cliff factor**" input field (Figure 5.6), allowing users to calibrate the effect of steep terrain on avalanche behavior.

While this **intensified transfer strengthens the powder cloud**, it comes at the expense of the avalanche core, which loses mass and momentum as a result. Understanding and accounting for this trade-off is essential for accurately modeling the dynamics of plunging avalanches and their impact zones.

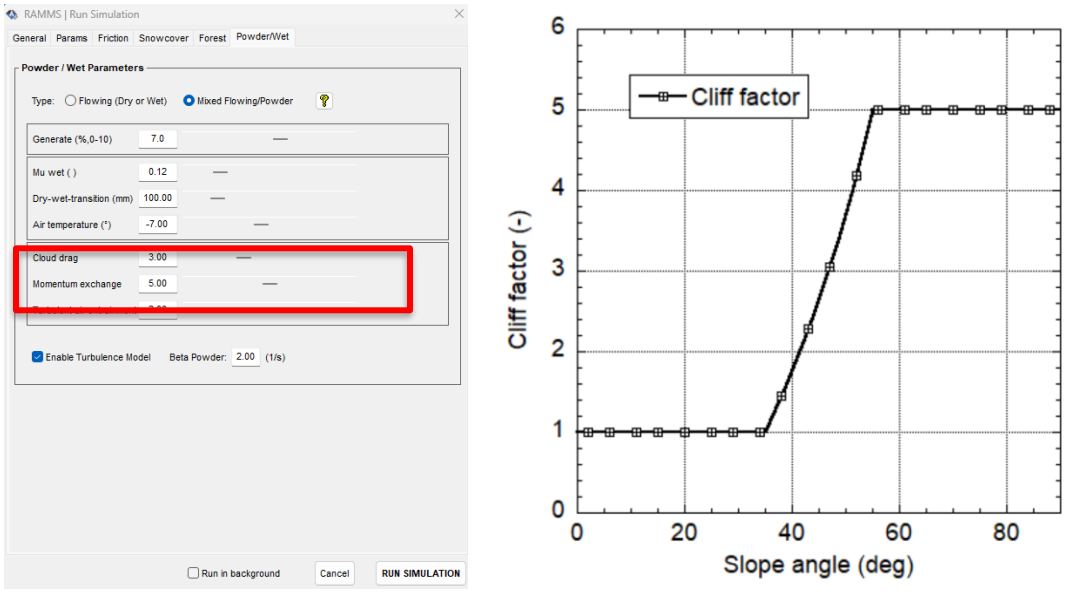


Figure 5.6: The Cliff Factor λ_Π is set in the Powder/Wet tab. The Cliff Factor magnifies the mass and momentum transfer from the core to the cloud when the avalanche core plunges over a steep slope. In this case the cliff factor magnifies the mass exchange by a factor 5 (see plot right).

5.5 Input: Turbulent Air Entrainment

Air entrainment plays a crucial role in governing volume, momentum, and energy balance within a powder avalanche. It directly influences the height of the powder cloud, moderates its velocity, and generates turbulence, making it a key parameter in avalanche modeling. The entrainment rate determines both the height (h_Π) and density (ρ_Π) of the powder cloud, allowing for direct comparison with photogrammetric and videogrammetric measurements.

Key Effects of Air Entrainment:

1. Regulates the **intensity of the avalanche air-blast**.

2. Acts as a **braking mechanism** for the powder cloud by accelerating ambient air into the flow.

3. Is inherently linked to **turbulence production.** Entrainment and turbulence generation are not independent processes; rather, each sustains the other through shear instabilities.

4. Is most pronounced at the **leading edge** of the avalanche, where the coupling with the avalanche core is strongest.

In **RAMMS::Extended** the air entrainment rate ($\dot{M}_{\Lambda\to\Pi}$, in m/s) depends on **turbulent energy** (R_Π) and the **density contrast** between the powder cloud and surrounding air $(\rho_\Pi - \rho_\Lambda)$. This relationship is expressed as:

$$\dot{M}_{\Lambda\to\Pi} = \left(1.16\,\psi + 0.013\sqrt{R_\Pi \hat{h}_\Pi}\right)(\rho_\Pi - \rho_\Lambda)$$

For practical applications, we recommend an air entrainment parameter of ψ = 2.00, which yields:

- $\dot{M}_{\Lambda\to\Pi}$ = 20m/s during the formation phase (high values of turbulent energy R_Π). This is the approximate speed of cloud growth at the leading edge of the avalanche.

- $\dot{M}_{\Lambda\to\Pi}$ = 0 to 4m/s in the wake of the powder cloud (low values of turbulent energy R_Π). In the wake of the cloud the density difference $(\rho_\Pi - \rho_\Lambda) \approx 0$, which sets $\dot{M}_{\Lambda\to\Pi}$ = 0.

These values align well with experimental measurements from Vallée de la Sionne (VdlS). Powder cloud settlement, indicating a removal of air from the cloud, is not considered in **RAMMS::Extended**, $\dot{M}_{\Lambda\to\Pi} \geq 0$, always.

The parameter ψ is inputted in the powder/wet tab, see Figure 5.7. The default value of $\psi = 2$. The plot in Figure 5.7 provides a range of possible entrainment rates (shaded region in m/s) for low and high turbulence $R_\Pi \hat{h}_\Pi$ =100J/m² and $R_\Pi \hat{h}_\Pi$ =5000J/m².

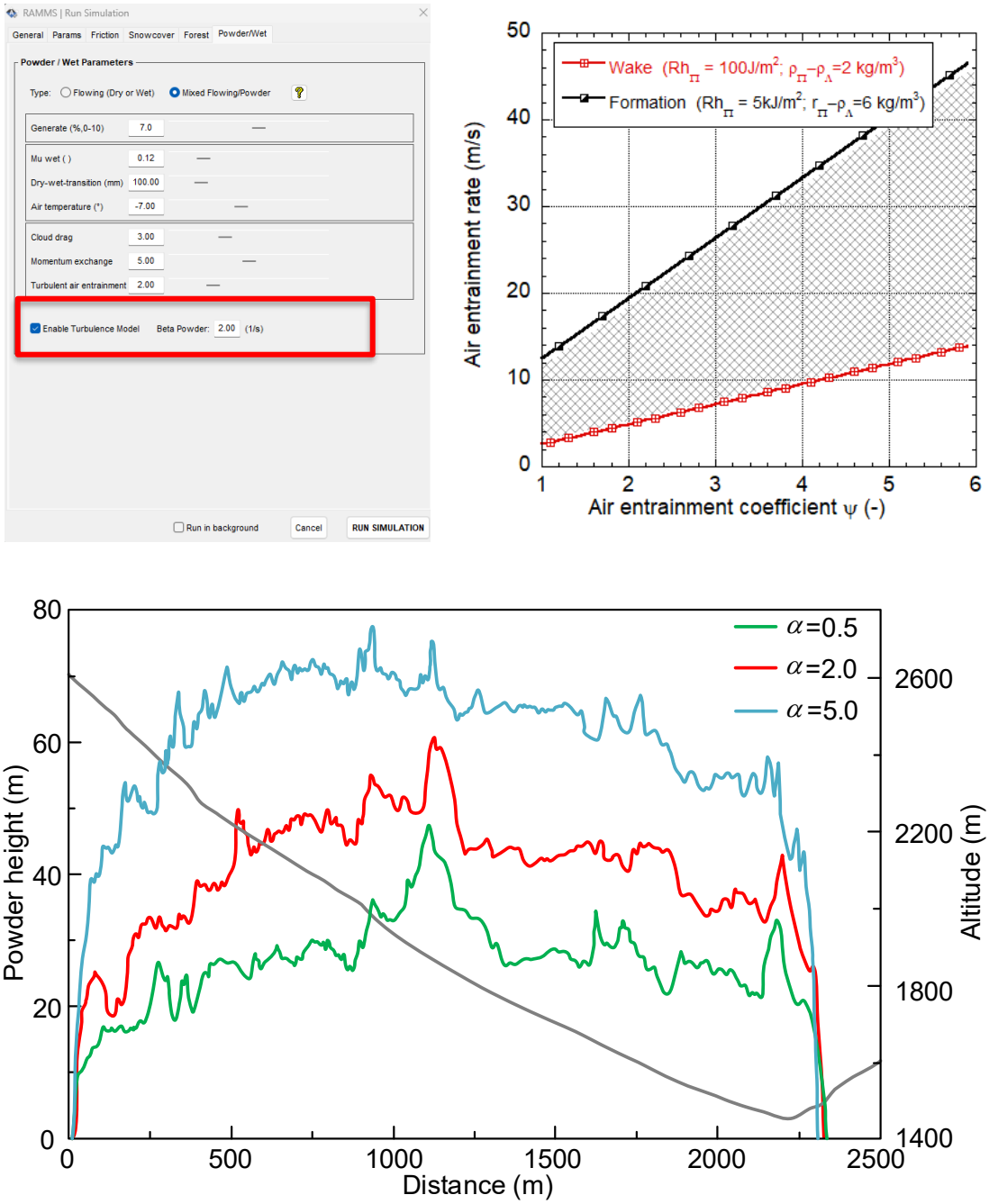


Figure 5.7: The Turbulent entrainment ψ is set in the Powder/Wet tab. Values between ψ = 1.0 and ψ = 6 should be selected. In the lower figure the calculated maximum powder cloud height of a Vallée de la Sionne avalanche 628 for ψ = 0.5, ψ = 2.0 and ψ = 5.0 are depicted.

5.6 Summary of Input Parameters

To assist readers in understanding and applying the cloud model, the following Table 5.2 summarizes the key **cloud-related parameters** along with their **recommended default values**. This overview is intended to support consistent setup and interpretation across simulations. Each parameter is briefly described to clarify its role within the model and to guide appropriate adjustments when needed. Users are encouraged to start with the suggested defaults and refine them based on specific site conditions or calibration data.

Model Parameter	Component	Symbol	Parameter Range Constant Scenario
Cloud Drag	$\prod$	$S_{\Pi}(\delta_{\Pi})$	Determines the force of the avalanche air-blast $2.0 \leq \boldsymbol{\delta} \leq 4.0$
Cliff factor	$\prod$	$\dot{M}_{\Phi \to \Pi}(\lambda_{\Pi})$	Determines momentum exchange of plunging powder avalanches $2.0 \leq \lambda_{\Pi} \leq 5.0$
Decay of Turbulent Energy	$\prod$	β_{Π}	Determines longevity of turbulence in powder cloud $0.5 \leq \beta_{\Pi} \leq 3.0$
Turbulent Air Entrainment	$\prod$	$\dot{M}_{\Lambda \to \Pi}(\psi_{\Pi})$	Determines air entrainment by turbulence in powder cloud $0.5 \leq \psi_{\Pi} \leq 4.0$

Table 5.2: Summary of RAMMS::Extended input parameters for the powder cloud.

5.7 Output: Air-Blast Pressure

One of the most critical output options for analyzing powder cloud dynamics is visualizing the pressure distribution in space. In the **RAMMS::Extended** GUI, this is achieved using the **yellow "P" (pressure) icon** , conveniently located in two places: the **upper menu bar** and the **Display** tab (Figure 5.8).

To simplify the notation, we often use $\|\vec{u}_{\Pi}\| = V_{\Pi}$ and $\|\vec{u}'_{\Pi}\| = V_{\Pi}^{T}$ which are the laminar and turbulent velocities.

Users can choose to represent pressure data in either **2D (map view)** for a comprehensive top-down perspective or in **3D mode** for a more immersive, spatially detailed visualization. This flexibility allows for an in-depth assessment of pressure variations within the powder cloud, enhancing both analysis and interpretation.

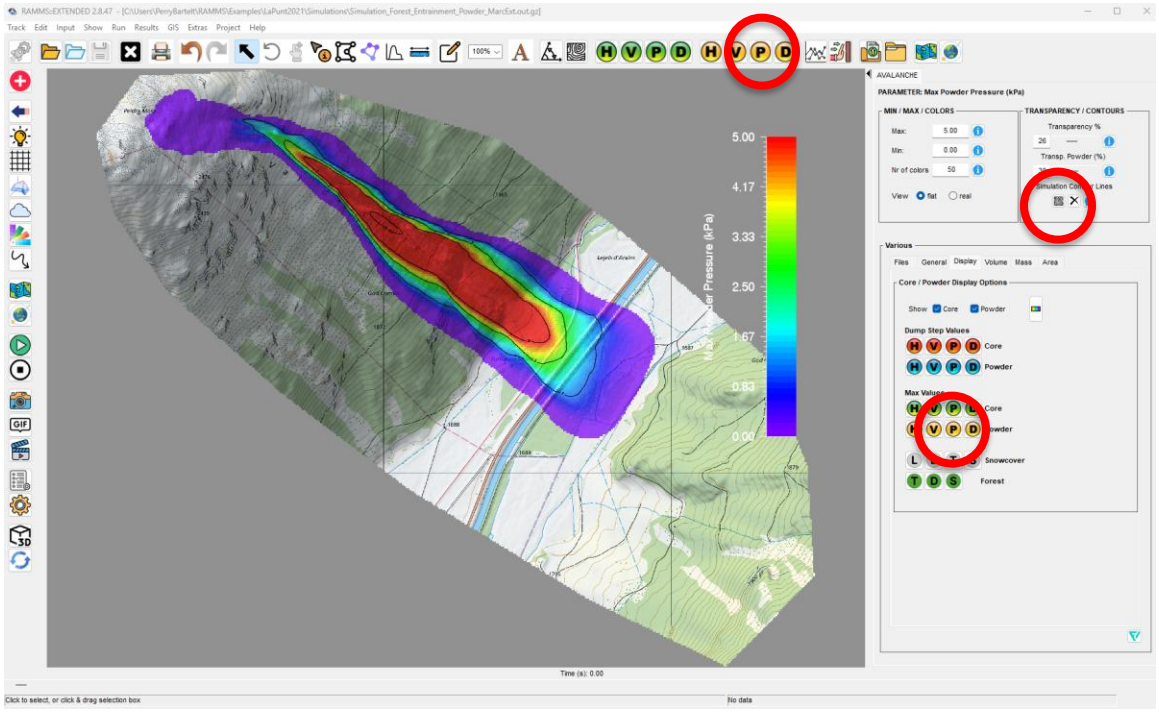


Figure 5.8: Maximum powder pressures can be visualized using the yellow P button. Contour lines can be drawn to determine ranges of pressure in conjunction with user defined min and max values. The maximum pressures are calculated on the ground.

Naturally, the **blue "P" button** allows users to visualize the **evolution of powder pressures along the track** at a given time step, offering insight into how pressure distributions develop and change over time.

When analyzing powder cloud dynamics, it is crucial to assess how velocity, density, and pressure vary with height. The plots presented in Figure 5.9 illustrate the **maximum value envelopes** for these parameters when the turbulence option is **disabled**, offering a comprehensive overview of the extreme conditions within the powder cloud.

The pressure is calculated from the cloud velocity and density as,

$$p_{\Pi} = \frac{1}{2}\rho_{\Pi}(V_{\Pi} + V_{\Pi}^{T})^2$$

with $V_{\Pi} = \|\vec{u}_{\Pi}\|$. This formula assumes that the moving cloud can flow around any object, large or small and is used to find pressures on house walls, building roofs or trees. As we

calculate velocity and density profiles in the z -direction, the pressure can also be ascertained at any height z in the powder cloud. Spatial depictions of the pressure represent the maximum pressures in the profile.

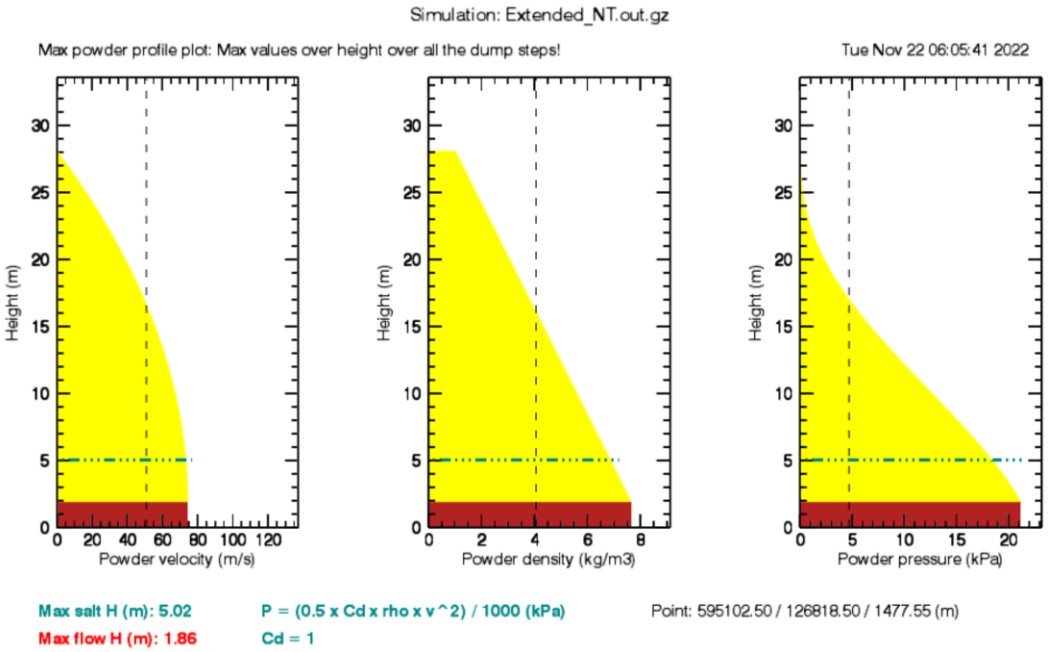


Figure 5.9: Powder Cloud Velocity, Density, and Pressure Envelopes Without Turbulence Enabled. These plots illustrate the maximum value envelopes for velocity, density, and pressure at various heights when the turbulence option is disabled. The yellow regions represent the envelope of maximum values across all profiles, determined independently for each height. Notably, these envelopes are not tied to a specific time step but rather encompass the highest recorded values across all times. Vertical dashed lines indicate the mean values of velocity, density, and pressure, while the red region highlights the core height. Additionally, the green horizontal line marks the height of the saltation layer, corresponding to the highest granule position.

The **yellow regions** in the profile plots denote the **envelopes of maximum values**, which are determined independently at each height by examining all computed profiles. Unlike time-specific outputs, these envelopes do **not correspond to a single moment** but rather represent the highest recorded values across **all time steps** throughout the simulation. This approach ensures that the results reflect the full range of possible extreme conditions at any given height.

To further aid interpretation, **vertical dashed lines** indicate the **mean values** of velocity, density, and pressure, providing a reference for how these parameters behave on average. The **red-shaded region** highlights the **core height**, which represents the central, most concentrated region of the powder cloud. Additionally, the **green horizontal line** marks the **saltation layer height**, corresponding to the highest point reached by individual granules. This distinction is particularly important for understanding particle motion and dispersion dynamics.

When the Enable Turblence Model is switched ON/Checked in the Powder/Wet input tab, **RAMMS::Extended** will overlay the turbulent velocities and turbulent pressures on the powder cloud height plots (Figure 5.10). In these plots, the gray regions correspond to **turbulent velocities** and and **turbulent pressures** (the density panal is unchanged).

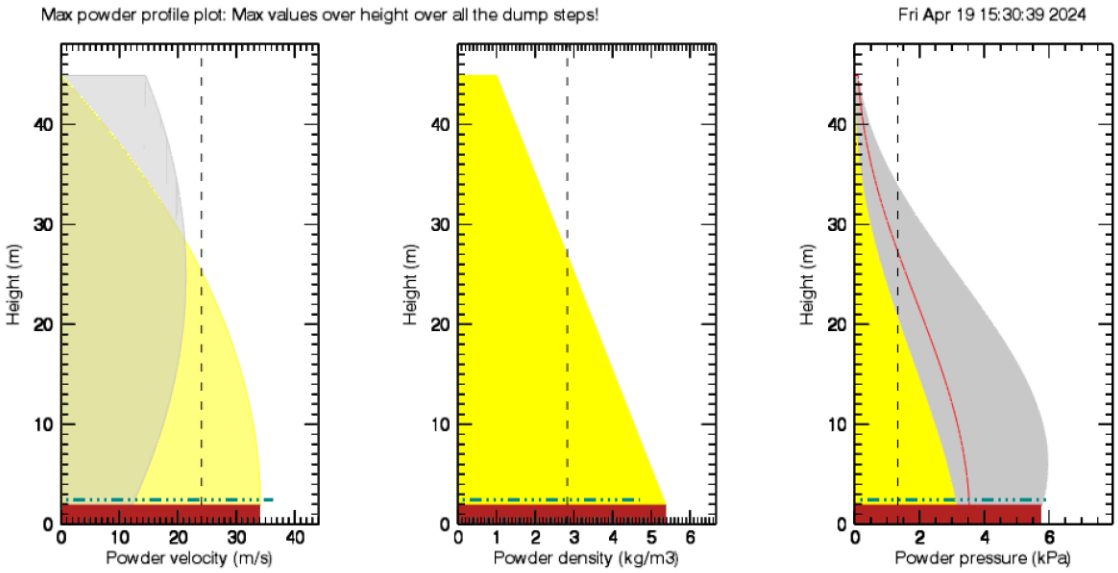


Figure 5.10: Powder Cloud Velocity, Density, and Pressure Envelopes withTurbulence Enabled. As before, the yellow regions represent the envelope of maximum values across all profiles, determined independently for each height. The gray regions represent the turbulent velocities and turbulent pressures. In the pressure panal, the gray region is found by adding the turbulent pressure to the laminar pressure, assuming the turbulent velocity is aligned in the direction of the flow. This is unlikely to occur. The gray region is found by assuming only one third of the velocity is aligned in the flow direction.

In the **velocity panel**, the distribution of **fluctuation velocities** is determined by first computing the mean fluctuation velocity, derived from the **calculated turbulent energy** R_{Π}. At each **dump step**, the total rate of **turbulence generation** $\dot{P}_{\Pi}$ is computed as:

$$\dot{P}_{\Pi} = \dot{M}_{\Phi\to\Pi} R_{\Phi} + \dot{W}_{\Pi} + \frac{1}{2}\rho_{\Lambda}\dot{M}_{\Lambda\to\Pi} u_{\Pi}^{2}$$

where each term represents distinct contributions to the turbulence:

- $\dot{M}_{\Phi\to\Pi} R_{\Phi}$ accounts for the random kinetic energy introduced from the avalanche core,
- $\dot{W}_{\Pi}$ represents the total **shear energy**,

- The final term, $\frac{1}{2}\rho_{\Lambda}\dot{M}_{\Lambda\to\Pi}u_{\Pi}^{2}$, quantifies the turbulence induced by **air entrainment** at the upper boundary of the powder cloud.

To define the relative contributions of these turbulence sources, we introduce the parameters:

$$\theta_0 = \left(\dot{M}_{\Phi\to\Pi}R_{\Phi} + \dot{W}_{\Pi}\right)/\dot{P}_{\Pi},\ \ \theta_h = \frac{1}{2}\dot{M}_{\Lambda\to\Pi}u_{\Pi}^{2}/\dot{P}_{\Pi}$$

which characterize the turbulence distribution at the **base** and **top** of the powder cloud, respectively.

The **parabolic distribution** of turbulent energy follows the property:

$$R_{\Pi}(\mathrm{z}=0) = \frac{\theta_0}{3}R_{\Pi}\ ,\ R_{\Pi}(z=h_{\Pi}) = \frac{\theta_h}{3}R_{\Pi}$$

This formulation highlights the distinct turbulence regimes within the powder cloud:

- At the **bottom**, turbulence is primarily dictated by the **total shear energy** $\dot{W}_{\Pi}$ and the **random kinetic energy** injected from the **avalanche core** $\dot{M}_{\Phi\to\Pi}R_{\Phi}$.

- At the **top**, turbulence is predominantly controlled by **air entrainment**, represented by $\frac{1}{2}\rho_{\Lambda}\dot{M}_{\Lambda\to\Pi}u_{\Pi}^{2}$, which influences the cloud's overall dissipation and expansion.

This formulation expresses how turbulence is distributed within the powder cloud, linking its variation to the underlying physical processes of shear, entrainment, and random kinetic energy input.

Owing to the parabolic distribution of turbulence and the nonzero end conditions, the **region of maximum turbulence is displaced from the surface**, shifting the pressure distribution upward. This, in turn, raises the **nose of the avalanche from the ground to an elevated position above the surface (**Figure 5.10**)**.

In the **pressure panel**, the turbulent pressure is computed by adding the **laminar (mean flow) pressure** to the contribution from the fluctuating **turbulent velocity**.

The total air-blast pressure in **RAMMS::Extended** is computed as the square of the sum of laminar (mean flow) and turbulent contributions,

$$p_{\Pi} = \frac{1}{2}\rho_{\Pi}[V_{\Pi} + V_{\Pi}^{T}]^{2}$$

The turbulent velocity field is given by

$$V_{\Pi}^{T}(z) = \sqrt{\frac{2R_{\Pi}(\mathrm{z})}{3\rho_{\Pi}(z)}}$$

This formulation assumes that only one-third of the turbulent kinetic energy is effectively aligned with the direction of the mean flow—a simplification that accounts for the multidirectional nature of turbulence within the powder cloud. Because turbulent fluctuations are not fully oriented with the flow, only a portion of their energy contributes directly to dynamic pressure. This adjusted contribution is visualized as the **grey region** in the pressure plots.

This adjustment better represents the expected contribution of turbulent fluctuations to the overall pressure field within the powder cloud, acknowledging the inherently chaotic and three-dimensional nature of turbulent flow.

In contrast, the **red line** in the pressure plot represents the sum of the squares of the velocities, expressed as

$$p_{\Pi} = \frac{1}{2}\rho_{\Pi}[(V_{\Pi}^{2}) + (V_{\Pi}^{T})^{2}]$$

This method is rooted in statistical and physical interpretations of kinetic energy or turbulent contributions, as it effectively captures the average of the squared velocities (similar to a variance in statistical terms). The sum of squares inherently accounts for the **individual contributions without regard to their relative directions**. We interpret this measure of pressure as representing the **minimum possible air-blast pressure** within the turbulent regime of the powder snow avalanche.

The **powder cloud pressure calculation** accounts solely for the **air-blast** component generated by the ice-dust suspension within the cloud. However, the powder cloud is also expected to contain a significant concentration of suspended particles. These particles are believed to exert **intense, localized impact forces** F_{π} on objects engulfed by the cloud. Such impact forces are highly localized (manifesting as point loads measured in kN) and exhibit a random, stochastic distribution. They are primarily associated with the dynamic processes at the **splashing or saltation front** of the avalanche. The calculation of these forces is expressed by the formula

$$F_{\pi} = \frac{4}{3}\rho_{\pi}{V_{\Pi}}^{2}A_{\pi}$$

where $A_{\pi} = \pi r_{\pi}^{2}$ is the cross-sectional area of the particle and $\rho_{\pi}$is the particle density. **Please** refer to **Section 6.15** for details, including the derivation of this formula.

5.8 Output: Cloud Velocity

The **maximum powder velocity** can be displayed using the **yellow V button** in both two-dimensional map view and full three-dimensional display modes (Figure 5.11). The **blue V button** depicts the powder cloud at a specific time step (Figure 5.10).

RAMMS::Extended computes the **mean, depth-averaged powder velocity**, $\|\vec{u}_\Pi\|$, under the assumption that the velocity at the upper surface of the cloud is zero, $\|\vec{u}_\Pi\| = 0,\ z = h_\Pi$. To capture the internal velocity distribution, a parabolic profile is adopted, ensuring that the mean velocity remains $\|\vec{u}_\Pi\|$. Consequently, the velocity reaches a maximum of 1.5 times $\|\vec{u}_\Pi\|$ at $z = 0$. This factor is incorporated into the visualization of the maximum values, reflecting the underlying structure of the velocity field.

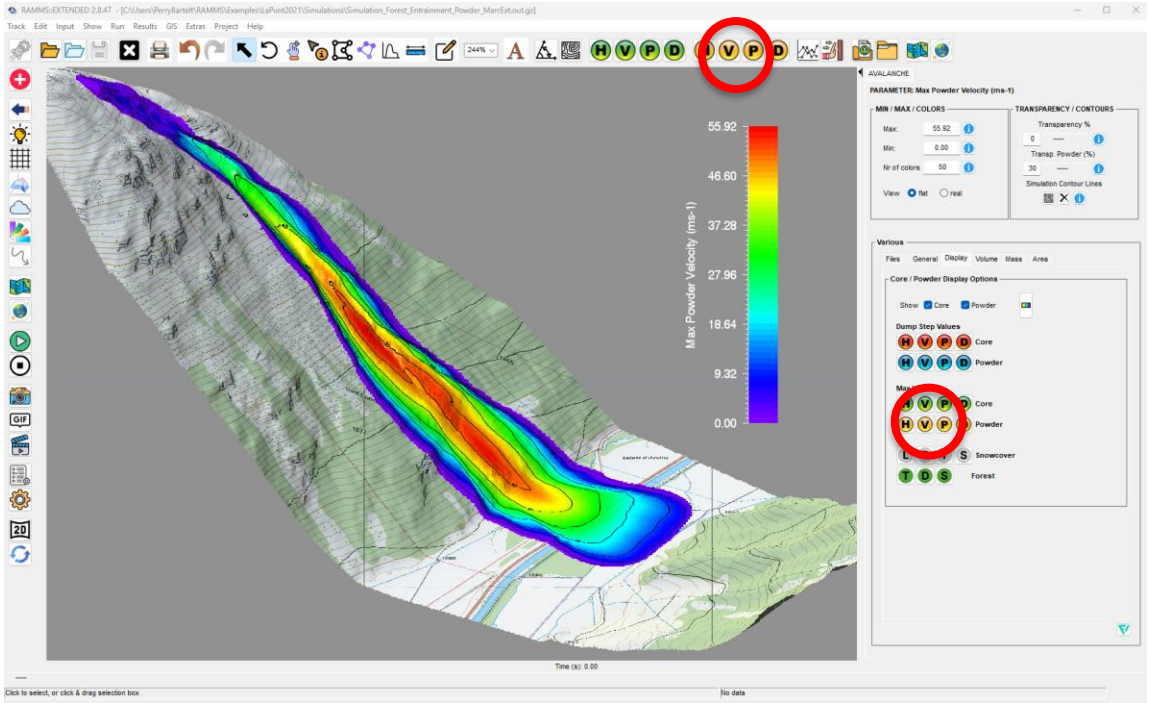


Figure 5.11: Three-dimensional depiction of max powder velocities using the yellow V button.

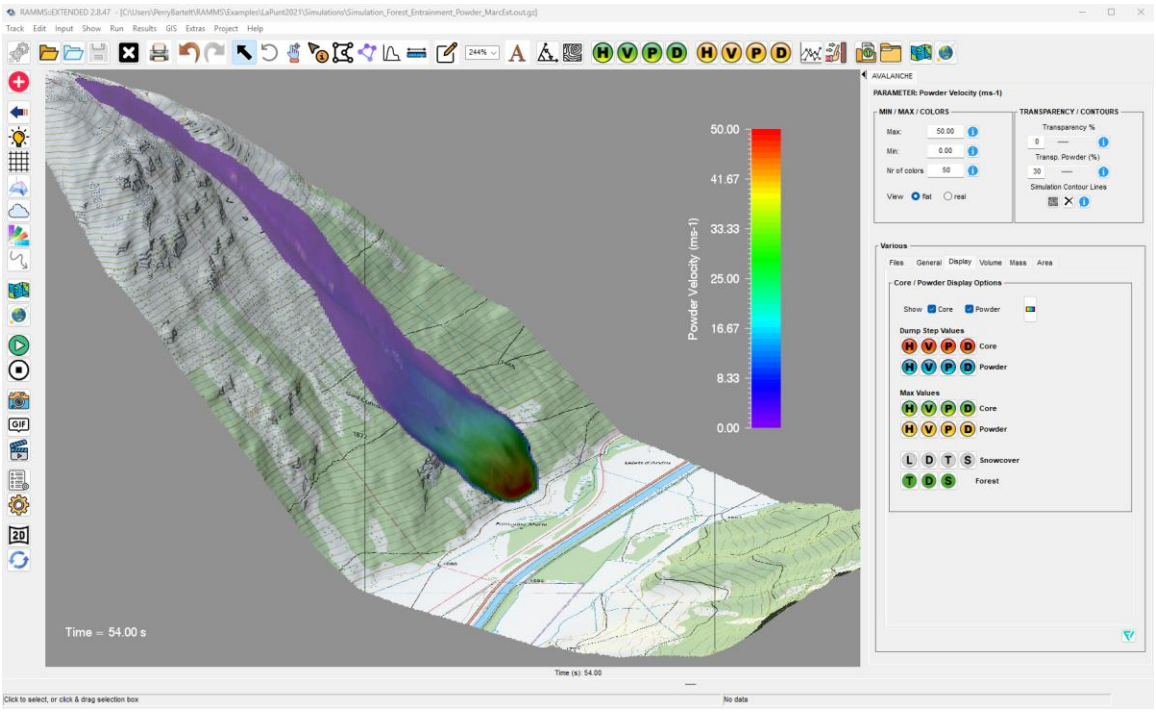


Figure 5.12: Displaying the powder velocities at a specific time using the blue V button. In this depiction it is possible to display the high velocity at the avalanche front and low velocities in the avalanche wake.

Pointwise data plots and **profile plots** are frequently utilized alongside visualizations of powder avalanche velocity to provide a more comprehensive representation of the flow dynamics.

5.9 Output: Cloud Density

For analytical and modeling purposes, avalanche mass can be conceptually divided into two components:

- **Core (Φ):** A basal, dense flow of colliding and shearing snow clods, exhibiting granular dynamics.

- **Cloud (Π):** A lofted, turbulent suspension of ice particles, driven by inertial forces and air entrainment.

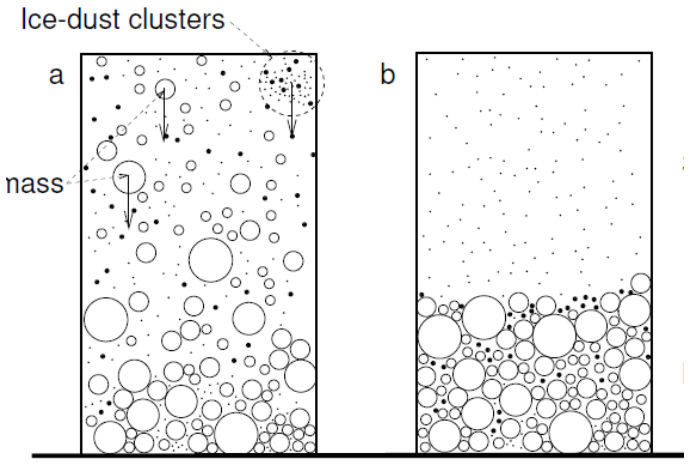


Figure 5.13: The mathematical division between the avalanche core and the suspension cloud is primarily governed by particle size. While larger snow and ice fragments experience brief uplift due to turbulence, they rapidly settle back into the dense, ground-based core. In contrast, finer ice particles remain suspended, giving rise to the turbulent powder cloud.

Particle size essentially divides the flow naturally into two components (Figure 5.13). The fundamental distinction between the avalanche core and the suspension cloud lies in the **size of the particulate mass**. Larger snow and ice particles within the core may be briefly lifted by turbulent forces but quickly return due to gravity, remaining part of the dense, basal flow. In contrast, finer particles—such as ice dust—achieve sustained suspension within the surrounding air, forming the turbulent powder cloud.

This natural segregation by particle size serves as the key mechanism that divides the avalanche into two distinct components: the dense, ground-hugging core Φ and the airborne suspension cloud Π.

Investigations have determined that the characteristic size of suspended dust particles is d ≤ 0.1 mm. Once airborne, these fine particles gradually **settle** and may **sublimate** depending on the ambient conditions.

For analysis, we assume that **air and ice particles move at the same velocity** ($\vec{u}_{\Pi}$). As a result, there is **no relative drag** between the suspended particles and the surrounding air. This assumption allows the ice particles to function as **tracers** of the airflow, providing insight into the movement and dynamics of the suspension cloud.

The volumetric fractions of ice and air are

$$\phi_i = \left[\frac{\hat{\rho}_{\Pi} - \rho_{\Lambda}}{\rho_{\mathrm{i}} - \rho_{\Lambda}}\right] \text{and } \phi_a = \left[\frac{\rho_{\mathrm{i}} - \hat{\rho}_{\Pi}}{\rho_{\mathrm{i}} - \rho_{\Lambda}}\right],$$

respectively. In these relationships ρ_i is the density of ice; ρ_Λ is the density of air; $\hat{\rho}_\Pi$ is the co-volume powder density. This gives rise to the following relationship for the mean density of the cloud,

$$\rho_\Pi = \left[\frac{\phi_i \hat{h}_\Pi}{h_\Pi + \phi_i \hat{h}_\Pi}\right]\rho_i + \left[\frac{h_\Pi}{h_\Pi + \phi_i \hat{h}_\Pi}\right]\rho_\Lambda$$

The suspension cloud is sustained by two sources of air: **the air entrained and mixed within the dense, shearing flow of the avalanche core, and the surrounding ambient air incorporated through turbulent mixing at the cloud's boundary**. Despite these two air sources, the only source of tracers is the avalanche core, which supplies the ice particles that become suspended and carried within the cloud.

The powder cloud density ρ_Π can be visualized using several tools in the **RAMM::Extended** GUI (Figure 4.11). Although it is possible to depict maximum cloud densities with the **yellow D buttons** density gradients from the front to talk of the cloud are best depicted with the **blue D button**.

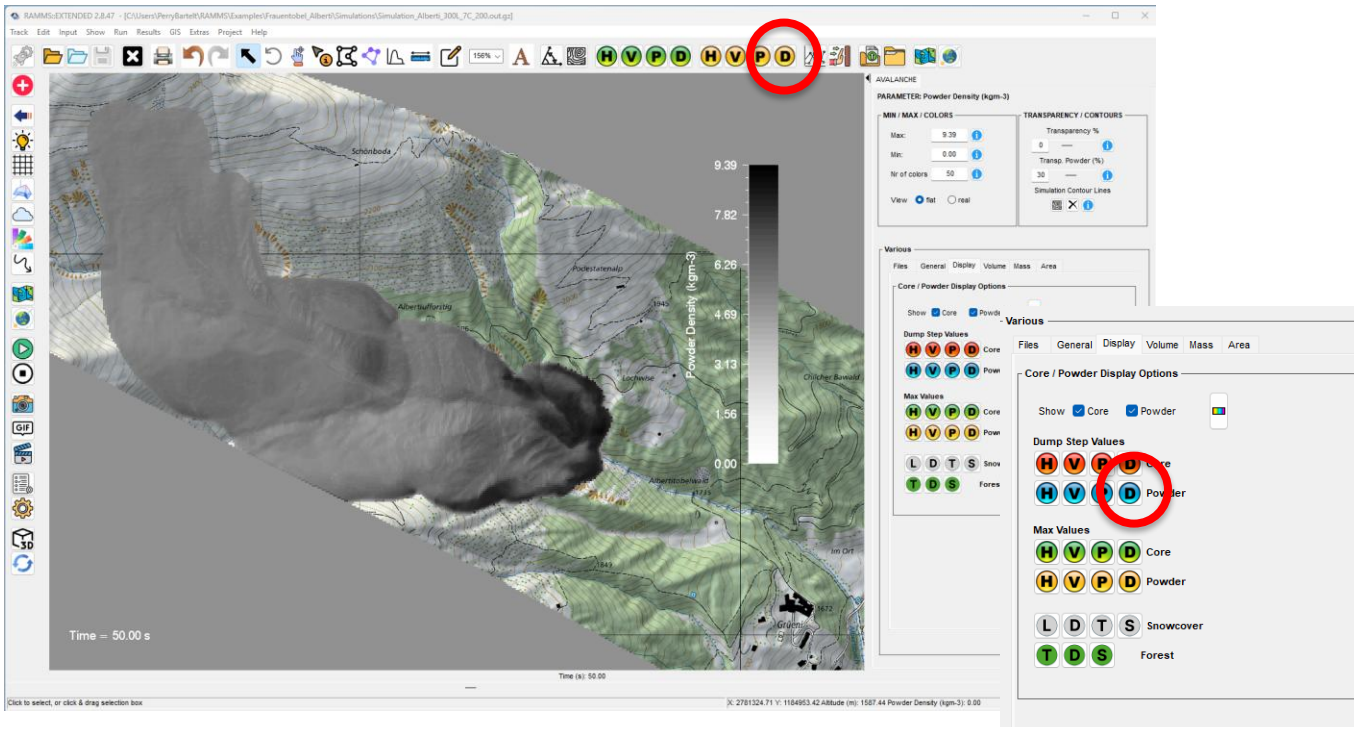


Figure 5.14: Visualizing the powder cloud density ρ_Π over time is possible using the blue D button in the display tab. The density of the cloud is depicted in a gray scale, with dark gray representing higher densities. Higher cloud densities are located at the avalanche front.

5.10 Output: Cloud Height

A distinctive feature of **RAMMS::Extended** is its ability to visualize cloud height with the avalanche core values (Figure 5.15). The cloud is depicted as a **semi-transparent shroud enveloping the avalanche core**, providing an immediate representation of its location and wider inundation area. Users can adjust the cloud's transparency using the **'Transp. Powder (%)**' slider in the TRANSPARENCY/CONTOURS panel, conveniently located on the right-hand side of the GUI. This flexibility allows for enhanced visualization of the interaction between the core and the powder cloud.

When animating the simulation results, the cloud will appear as it is formed, but will disappear before the final simulation step is reached. **The cloud disappearance/fade out begins three time steps before the final time step**.

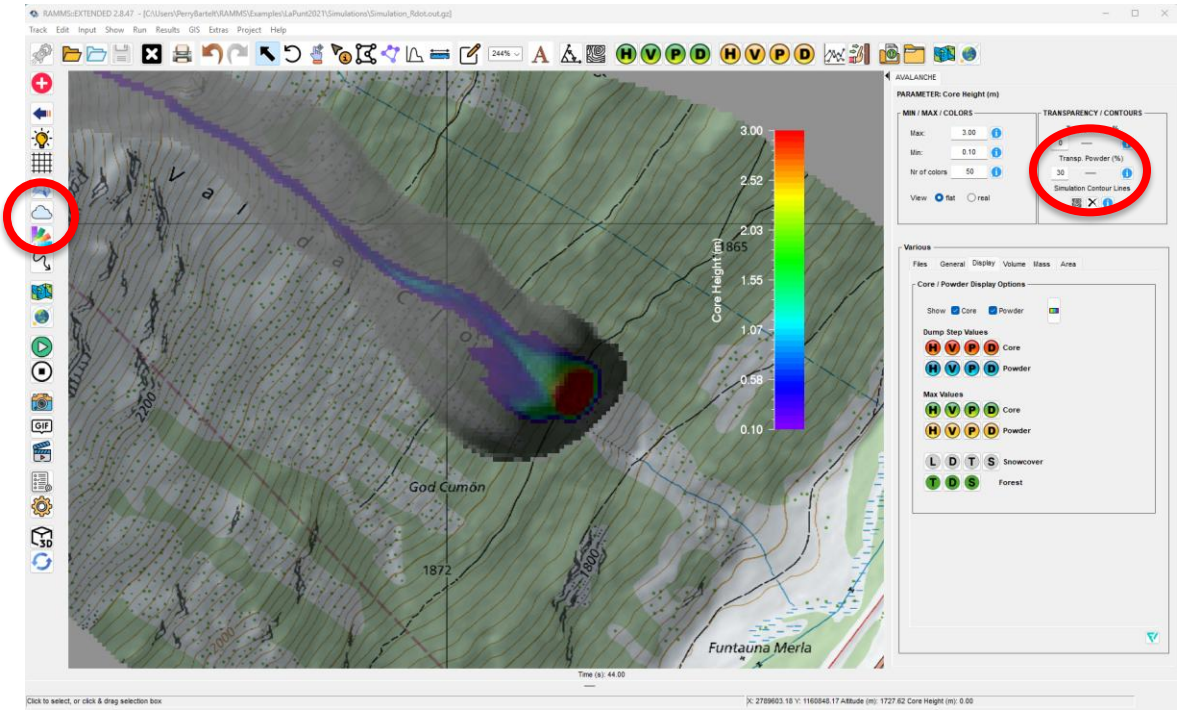


Figure 5.15: The gray region represents the powder cloud, while the colored region indicates the avalanche core height. By default, the cloud is displayed using core values but can be toggled on or off using the Cloud icon on the left-hand side of the GUI (red circle). This allows the user to visualize the relative positions and velocities of the core and cloud. Additionally, the transparency level of the cloud can be adjusted using the control on the right-hand side of the GUI (red ellipse).

Similar **yellow and blue buttons** are used to depict the maximum cloud height and the spatial evolution of the cloud height over time. The cloud height and the time-evolution of the cloud height are often depicted using the profile and point information tools (Figure 5.16).

The **cloud height is defined in from the terrain surface to the top of the cloud** in the slope-perpendicular direction. The cloud height always grows as air is incorporated in the cloud. It never settles. The cloud height often reveals large variations in height (Figure 5.16), which depend on the core dynamics.

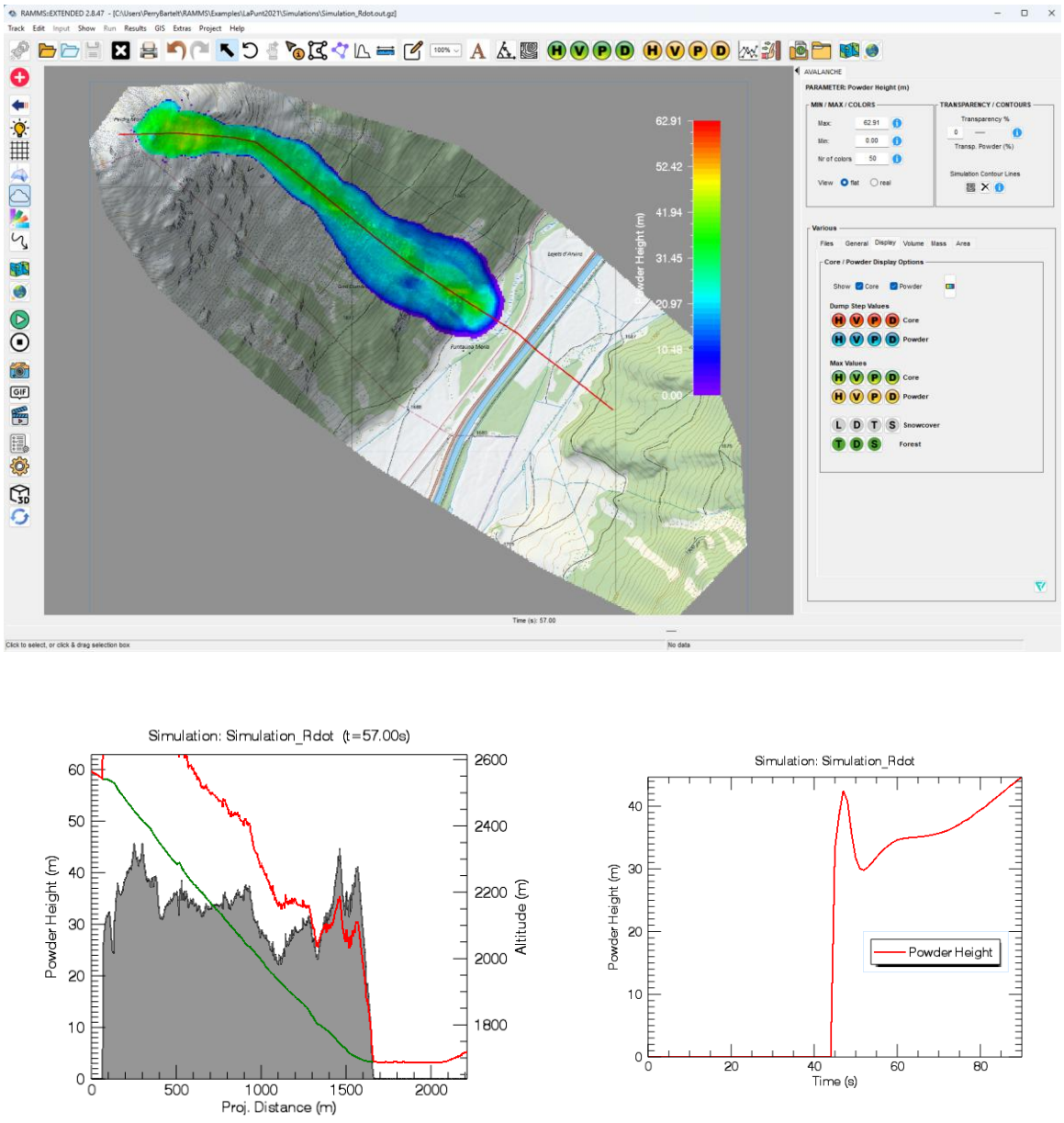


Figure 5.16: The profiling and point information tools are often used to extract simulation data and depict cloud information such as the cloud height over the track at a specific simulation time step (left) or the arrival time of the cloud, and the cloud height over time (right).

Chapter 6 Avalanche Impact Pressures

6.1 Swiss Pressure Guidelines for Hazard Mapping

RAMMS::Extended's approach to calculate **avalanche impact pressures** is closely tied to Switzerland's classification of pressure levels for avalanche hazard mapping. Switzerland employs a scenario-based hazard mapping system that categorizes terrain into distinct hazard zones based on calculated avalanche impact pressures for specific avalanche return periods. These guidelines are essential for land-use planning, infrastructure protection, and engineering design, helping to ensure that buildings and other structures can withstand potential avalanche forces.

6.1.1 Key Pressure Threshold: 30 kPa

A fundamental benchmark in Swiss avalanche hazard zoning is **30 kPa**, which is often used as a threshold for defining high-risk zones. This value is significant because:

- **30 kPa represents the pressure required to destroy an unreinforced building**, making it a practical threshold for determining areas where human life and property are at risk.
- Structures located in areas with **expected impact pressures exceeding 30 kPa** require **specialized engineering measures** or are **prohibited from construction**.
- This value has been widely validated through **historical avalanche damage assessments and structural impact studies**.

For example, Switzerland classifies avalanche-prone areas using a **color-coded hazard map system**, which defines different levels of risk based on expected avalanche impact pressures:

1. **Red Zone (High Hazard) – Impact Pressures ≥ 30 kPa**
    - Areas exposed to **high-density, high-velocity avalanches**.
    - Buildings **are not permitted**, except for avalanche-resistant structures.
    - Infrastructure must be **heavily reinforced**, with strict engineering requirements.
2. **Blue Zone (Moderate Hazard) – Impact Pressures between 3 kPa and 30 kPa**
    - Avalanche impact pressures can cause **structural damage but are not necessarily destructive**.
    - Residential buildings **may be permitted** but require **reinforced construction** (e.g., impact-resistant walls, deflection structures).
    - Infrastructure design must consider **potential dynamic forces and impact duration**.
3. **Yellow Zone (Low Hazard) – Impact Pressures < 3 kPa**
    - Areas where **light structures may experience damage** but human safety is not significantly threatened.

    - Buildings are allowed but may require **protective measures** such as avalanche fences, reinforced walls, or deflecting berms.
4. **White Zone (No Significant Hazard)**
    - No avalanche-related building restrictions.

It is crucial to recognize that for impact pressures exceeding **30 kPa**, there is minimal benefit in performing detailed force calculations—these high loads typically govern design decisions outright. However, in the intermediate pressure range of **3 kPa to 30 kPa**, precise calculations become **essential**, as the choice of design formulas can significantly influence construction costs and safety margins. Within this critical range, engineering judgment plays a pivotal role in interpreting simulation results and applying them effectively to real-world scenarios.

Most field measurements of impact pressures—such as those conducted at **Ryggfonn** and **Vallée de la Sionne**—have been recorded on steep slopes where avalanches reach extreme velocities and experience substantial gravitational forces. As a result, these measurements often fall outside the **3 kPa – 30 kPa** range, where detailed impact force calculations are most relevant. Furthermore, few research sites have deployed **strain gauges** or **force sensors** capable of measuring the internal structural forces that engineers need to assess load-bearing requirements. Historical measurements, such as those conducted in Ryggfonn during the **1980s**, made some progress in this regard, but even today, there remains a **critical gap** in our ability to translate **external impact pressures** into the **internal structural forces** that dictate a structure's response.

This distinction is vital. Engineers must not only consider the raw magnitude of impact pressures but also how these forces manifest within structural components, whether as **bending moments, shear forces, or foundation loads**. A thorough understanding of these mechanics is essential for designing avalanche-resistant structures that are both **cost-effective and resilient**.

6.1.2 Practical Considerations for Engineers

- **Obstacle Geometry Determines Impact Forces**
    - **Wide structures (e.g., walls, dams, buildings)** experience **high pile-up forces**, requiring consideration of **pile-up pressure models**.
    - **Narrow structures (e.g., pylons, bridge supports)** experience **lower pile-up but higher deflection and shear forces**, demanding a different design approach.
- **Impact Pressure Varies with Flow Properties**
    - Faster, denser avalanches **exert greater impact forces**, requiring **reinforced structures** in **high-velocity zones**. These forces are best determined with work-energy methods.
    - **Wet snow avalanches** tend to have **lower impulsive pressures** and **longer interaction durations including more pronounced shearing forces**.
- **Elevated Structures Require Powder Cloud Analysis**

    - Powder cloud pressures are usually **lower than core pressures** but can still damage **tall, lightweight structures** like transmission towers.
    - **High-altitude buildings** in avalanche-prone areas should consider **combined loading from both core and powder cloud forces**.
- **Dynamic Effects Can Amplify Stresses**
    - Tall, flexible structures like **masts and pylons** can experience **dynamic magnification**, where the natural frequency of the structure aligns with impact forces, leading to **higher-than-expected internal stresses**.
    - **Anchoring and foundation design** play a crucial role in mitigating **dynamic oscillations**.
- **Engineering Design Should Incorporate Multiple Load Scenarios**
    - A single force model may not be sufficient—**combined stagnation, deflection, and shearing loads** should be considered for comprehensive structural analysis.

6.2 Avalanche-Obstacle Interaction Forces

When an avalanche encounters an obstacle, the impact forces arise from a combination of three key mechanisms (Figure 6.1):

1. **Pile-Up Forces**
    - Occurs when snow **accumulates in front of an obstacle**, forming a compacted wedge.
    - Typically observed in **wide obstacles** such as walls, buildings, and deflection dams.
    - Results in high impact pressures as the incoming mass is rapidly decelerated.
2. **Deflection Forces**
    - Occurs when avalanche snow **moves around an obstacle** rather than stopping.
    - Common for **narrow structures** like bridge supports, pylons, or slender buildings.
    - The flow exerts **lateral forces** on the obstacle, which can induce **bending moments and shear stresses**.
3. **Shearing (Edge) Forces - Jamming and Frictional Loading**
    - Created when **deflected snow interacts with the surrounding avalanche flow**, causing localized shear stresses that are taken up by the obstacle.
    - Can significantly **increase loading** on structures with **sharp edges** or **narrow geometries**.

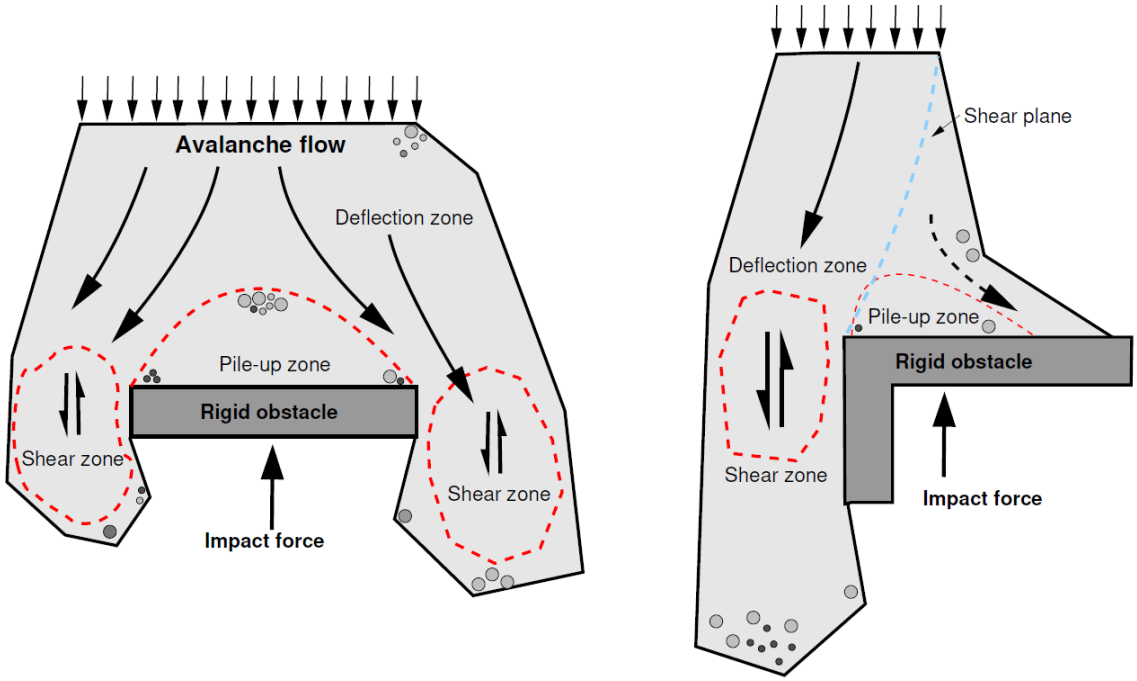


Figure 6.1: When an rigid obstacle is impacted by an avalanche, three different force are created: pile-up forces, deflection forces and shear forces.

For decades, the avalanche research community has relied on **empirical pressure formulas** to simplify the complex and dynamic interactions between avalanches and structures. These formulas take the general form:

$$p = \frac{1}{2}\rho_\Phi C_d \|\vec{u}_\Phi\|^2$$

where C_d is the pressure coefficient, a key parameter governing avalanche impact forces. This equation establishes a direct relationship between the impact pressure, the square of the avalanche velocity $\vec{u}_\Phi$ and the avalanche density ρ_Φ.

In RAMMS::Extended we consider the cases shown in Table 6.1.

Traditionally, recommended values for C_d are:

1. C_d = 1 for thin obstacles, where flow largely deflects around the structure.
2. C_d = 2 for wide obstacles, which tend to cause significant pile-up effects.

This empirical framework is also incorporated into **RAMMS::Extended**. However, in **RAMMS::Extended**, we refined the traditional approach by introducing three distinct force types:

- **Pile-up (Ξ, Xi):** Represents stagnation forces that develop when an avalanche collides with a wide structure, causing snow to compact and accumulate. We use the uppercase Greek letter Xi (Ξ) because its shape visually appears as if the upper

and lower surfaces are causing the interior plane to become smaller, reflecting the compression of avalanche material upon impact with a rigid obstacle.

- **Deflection (Υ, Upsilon):** Represents forces associated with flow deviation around thinner obstacles. We use the uppercase Greek letter Upsilon because the stem divides into two branches representing deflection and the flow of avalanche mass around the obstacle.
- **Shearing (τ, Tau):** Shearing forces refer to the long-duration, non-impulsive stresses that develop when dense, slow-moving snow flows interact with obstacles or boundaries. Unlike impact or pressure forces that arise from rapid momentum transfer, shearing forces emerge gradually as the flowing material slides, grinds, or deforms while passing along the surface of a rigid object. These forces are especially prominent at the edges of obstacles or at channel walls, where velocity gradients are strongest and flow is deflected laterally.

6.3 Voellmy's Pressure Model p_{Ξ} (Wide Obstacles)

Parts I and II of Voellmy's ***Über die Zerstörungskraft von Lawinen*** (*Schweizerische Bauzeitung*, Heft 12) appeared in March 1955. The first part (*Abschätzen der Lawinenkräfte aus ihren Wirkungen*) provides an overview of his observations from the avalanche events in Austria in 1954, primarily the destruction of buildings in the villages of Blons and Dalaas. Part II (*Art und Grössenordnung der beobachteten Druckwirkungen*) summarizes the observations, which Voellmy condenses into five primary conclusions:

- The **back-up of mass at impact** (German: *Stauwirkung* which we translate to pile-up) **is the primary contribution to the avalanche force**. Therefore, the pressure exerted by an avalanche on an obstacle can be found by considering the **stagnation pressure**.
- The **back-up of mass** (German: *stauen*) at impact creates forces that are **parallel** (horizontal direction) to the flow but also **slope-perpendicular** to the flow. The most important example being forces that are exerted in the upwards direction at impact. It is estimated that these vertical forces can be half the magnitude of the horizontal force. The overall effect of the forces is determined by the surface properties (friction) of the structure. **The maximum forces in the parallel and perpendicular directions must not occur simultaneously**.
- When avalanches **run over roofs**, the pressures acting on the roof are **hydrostatic**; that is, given by the flow height and density of the avalanche snow.
- **Hard materials** (building debris, rocks, wood) are often entrained into the avalanche and can cause considerable damage. Deformable materials (German: *Arbeitsvermögen*) should therefore be considered in new building designs.

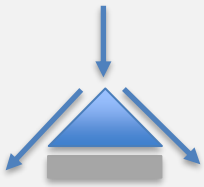
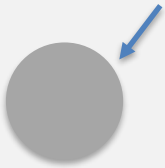
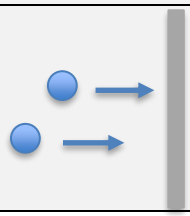
Process	Theory / Section
	**Direct impact of long wall.** Forces of stagnation and pile-up. Voellmy, work-energy and empirical formulas $C_d = 2$. **Sections 6.3 and 6.4**
	**Direct Impact of thin, flat obstacle with deadzone formation.** Forces of deflection. Work-energy and empirical formulas $C_d = 1$. **Section 6.8**
	**Direct Impact of thin, circular objects (pylons and tree stems).** Forces of deflection. Work-energy and empirical formulas $C_d = 1$. **Section 6.9**
	**Direct and oblique impact of splitting wedges.** Forces of deflection. Work-energy methods. **Section 6.11**
	**Direct and oblique impact of rectangular buildings.** Forces of deflection. Work-energy methods. **Section 6.12**
	**Shearing forces on edges, corners and lateral side walls.** Use Voellmy Ansatz **Section 6.13**
	**Particle impacts from snow/ice granules. Change of momentum/work-energy.** **Section 6.15**

Table 6.1: RAMMS::Extended considers the six impact categories involving forces of stagnation and pile-up, forces of defelction and particle impacts.

	Thin Obstacles	Wide Obstacles
RAMMS::Avalanche	C_d=1	C_d=2
RAMMS::Extended	p_{Υ}	p_{Ξ}
Process	Deflection / Flow Around	Stagnation / Pile-up

Table 6.2: Difference between RAMMS::Avalanche and RAMMS::Extended. In RAMMS::Extended we term the C_d=1 and C_d=2 as "Standard".

- The **pressures from the cloud impact** are small in comparison to the pressures from the avalanche core. These reached 5 kPa and were enough to break windows and doors. However, large pressures arising from suction effects (German: *Sogwirkung*), that would typically occur at the turbulent avalanche front, could not be conclusively identified in either Austria or Switzerland. **Voellmy** was therefore the first to recognize that the **way avalanche snow compacts during pile-up** is a primary factor influencing **the magnitude of avalanche impact pressure**, particularly the dynamic pressure on long walls and buildings. This is especially critical in scenarios where the avalanche flow cannot easily be deflected or directed around an obstacle. To determine impact pressure, Voellmy developed a **relationship between impact pressure and avalanche snow density**, which has been largely confirmed through experimentation.

The theoretical part of Voellmy's article appeared in Part III a month later in April 1955. Sections 1 (*Der Strömungsvorgang des Schnees*) and 2 (***Geschwindigkeit und Fliesshöhe der Lawinen***) appeared on April 9. These sections described what is now commonly referred to as the **Voellmy model**. They specifically present Voellmy's concept of flow friction and the well-known parameters μ and ξ. Section 3 of Part III appeared on April 23 and contained Voellmy's derivation of runout distance and **impact pressure**.

Voellmy's derivation to find the pile-up pressure p_{Ξ} begins with the assumption that high-speed compression of avalanche snow is controlled primarily by the **rate at which air escapes the pore space during compaction** (Figure 6.2). He further assumed that the deformation of the snow ice-matrix provides negligible resistance to compaction compared to the force required to expel air. He did not consider the compressibility (elasticity) of solid ice or any water present in the pore space. Instead, he treated the compaction process as entirely plastic, characterized by an **irreversible reduction in pore space**.

Section 3, Part III begins with the presentation of two important physical concepts. The first is that the **total energy line** H for a fluid flow of known (slope perpendicular) height h', velocity $\|\vec{u}_{\Phi}\|$ and avalanche density ρ_{Φ} is,

$$H = \left(h' + \frac{\|\vec{u}_\Phi\|^2}{2g} + \int_{\rho_\Lambda}^{p} \frac{dp}{\rho_\Phi} \right)$$

The third term in this equation is of interest because Voellmy considered avalanche snow to be a compressible fluid. By compressible, Voellmy meant **compactible**. That is, under a pressure p, the pore space of avalanche snow is reduced, squeezing out air, and replacing it with solid ice material. The value ρ_Λ is the reference pressure, which is the ambient air pressure. More importantly, Voellmy maintained that to compute the stagnation pressure, and therefore the force acting on the obstacle, it is **necessary to understand how avalanche snow compacts under pressure, especially the large compressive pressures at impact**.

The upper limit for the maximal impact force on a structure could be easily found directly from the entirely inelastic stagnation pressure,

$$p_\Xi = \bar{\rho}_\Xi \left(h' + \frac{\|\vec{u}_\Phi\|^2}{2g} \right)$$

In this equation $\bar{\rho}_\Xi$ represents the mean avalanche density during impact. This value is given by the on-coming avalanche flow density ρ_Φ (In **RAMMS::Extended** this value is calculated) and the dynamic, compacted density at the end of impact ρ_Ξ,

$$\bar{\rho}_\Xi = \frac{\rho_\Phi + \rho_\Xi}{2}$$

In his article, Voellmy derived a closed form solution for the compacted density ρ_Ξ. He began by defining the density of flowing avalanche snow as:

$$\rho_\Phi = \rho_\Omega[1 - \phi_\Phi] + \rho_\Lambda \phi_\Phi$$

and the density of compacted avalanche snow is:

$$\rho_\Xi = \rho_\Omega[1 - \phi_\Xi] + \rho_\Lambda \phi_\Xi$$

where ϕ_Φ represents the air content of the flowing avalanche core, and ϕ_Ξ the air content of the compacted avalanche snow, stopped by the obstacle. The value of ρ_Ω defines the maximum density for different avalanche snow types, e.g. dry, granular snow ρ_Ω=600 kg/m^3 and for wet avalanche snow ρ_Ω=1000 kg/m^3. In many ways the density ρ_Ω resembles the co-volume density of the avalanche core $\hat{\rho}_\Phi$. However, Voellmy's ρ_Ω refers to the static compacted values of dry granular snow, and not the flowing properties. In **RAMMS::Extended** we take the air pressure to be 1 bar, $p_\Lambda = 1$ bar, $p_\Lambda \approx 100$ kPa.

Using **Boyle's law** at constant temperature, Voellmy determined the rise in pressure at impact:

$$(p_\Xi + p_\Lambda)\phi_\Xi = p_\Lambda \phi_\Phi = \; p_\Lambda \left[\frac{\rho_\Omega - \rho_\Phi}{\rho_\Omega}\right]$$

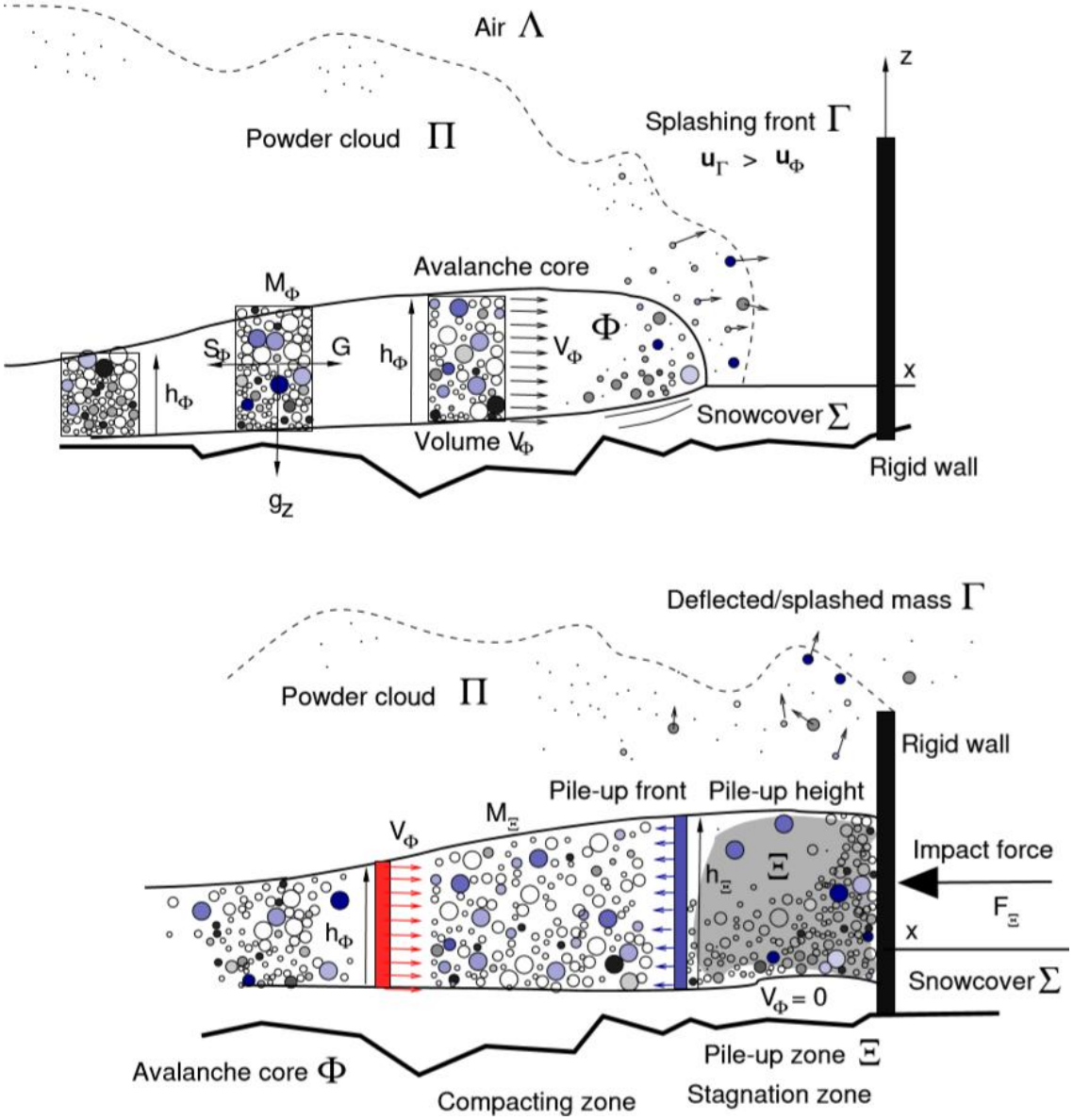


Figure 6.2: Voellmy Pile-up Forces in RAMMS::Extended. This illustration depicts the interaction between an avalanche core and a rigid wall, highlighting the fundamental mechanics of pile-up forces. Upon impact, the avalanche mass undergoes rapid compaction, coming to a sudden halt as it collides with the obstacle. During this process, air trapped within the snowpack is expelled, influencing the density and pressure distribution at the point of contact. The degree of compaction directly governs the magnitude of the impact force, making it a critical factor in understanding avalanche-structure interactions.

Boyle's Law states that **the pressure and volume of a gas are inversely proportional when temperature and the amount of gas remain constant**. Mathematically, it is expressed as, $p_1 V_1 = p_2 V_2$. This formula can be rearranged to find the volumetric content of air in the compacted snow,

$$\phi_{\Xi} = \left[\frac{p_{\Lambda}}{p_{\Xi} + p_{\Lambda}}\right]\left[\frac{\rho_{\Omega} - \rho_{\Phi}}{\rho_{\Omega}}\right]$$

The density of compacted snow in relation to flowing avalanche snow is given by

$$\rho_{\Xi} = \left[\frac{\rho_{\Phi}}{1 + \Delta\phi}\right] = \left[\frac{\rho_{\Phi}}{1 + \phi_{\Xi} - \phi_{\Phi}}\right]$$

Substituting known values, the relationship between pile-up density and stagnation pressure is (Figure 6.3):

$$\rho_{\Xi} = \rho_{\Phi}\frac{\left[1 + \frac{p_{\Xi}}{p_{\Lambda}}\right]}{\left[1 + \frac{\rho_{\Phi}}{\rho_{\Omega}}\frac{p_{\Xi}}{p_{\Lambda}}\right]}$$

The value of ρ_{Ξ} is the final compaction density for the applied pressure p_{Ξ}. The mean density $\bar{\rho}_{\Xi}$ during the impact is

$$\bar{\rho}_{\Xi} = \frac{\rho_{\Phi} + \rho_{\Xi}}{2} = \frac{\left[\rho_{\Phi} + \frac{p_{\Xi}}{p_{\Lambda}}\left(1 + \frac{\rho_{\Phi}}{\rho_{\Omega}}\right)\frac{p_{\Xi}}{p_{\Lambda}}\right]}{\left[1 + \frac{\rho_{\Phi}}{\rho_{\Omega}}\frac{p_{\Xi}}{p_{\Lambda}}\right]}$$

This result can be substituted into the pressure formula,

$$p_{\Xi} = \bar{\rho}_{\Xi}\left(h' + \frac{\|\vec{u}_{\Phi}\|^2}{2g}\right)$$

This is a quadratic equation in p_{Ξ} with the final closed-form solution:

$$p_{\Xi} = \rho_{\Omega} g\left[\sqrt{\left(\frac{q}{2}\right)^2 + H\left(\frac{p_{\Lambda}}{\rho_{\Omega}}\right)} - \frac{q}{2}\right]$$

where q is defined as:

$$q = \frac{p_{\Lambda}}{\rho_{\Phi}} - \left(\frac{H}{2}\right)\left[1 + \frac{\rho_{\Phi}}{\rho_{\Omega}}\right]$$

This formula for p_Ξ is implemented in **RAMMS::Extended under the heading of Voellmy**. In general, the formula provides lower values of impact pressure than the empirical formula with C_d=2, see Figure 6.4.

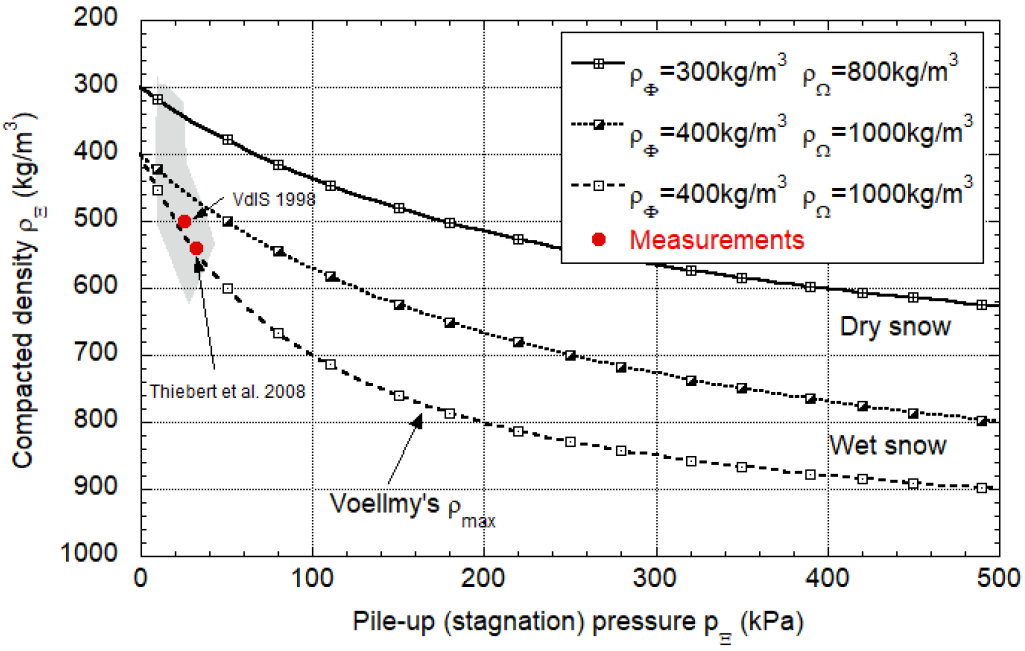


Figure 6.3: Calculated values of compacted density ρ_Ξ as a function of the pressure p_Ξ. Voellmy's formula appears to underestimate the compacted density. Voellmy also provided the maximum compaction density possible as $\rho_{max} = \left(\rho_\Phi + \rho_\Omega \frac{p_\Xi}{p_\Lambda}\right)/\left(1 + \frac{p_\Xi}{p_\Lambda}\right)$ which fits the experimental results well.

In summary, based on his observations and back-calculations of real events, Voellmy believed that the **compactibility of avalanche snow essentially defines the magnitude of the stagnation pressure at impact**. Assuming that avalanche snow is a non-compactible, elastic material leads to an over-estimation of the impact forces. The stagnation pressure is derived from **energy considerations**.

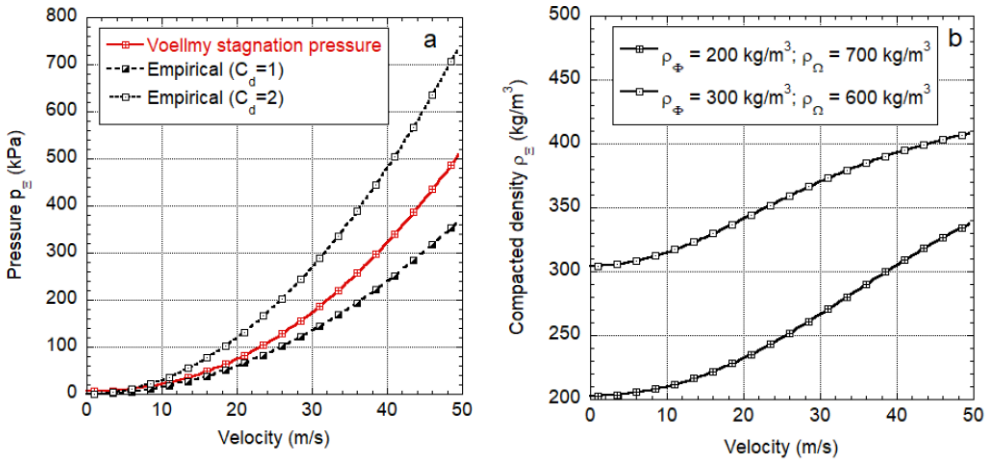


Figure 6.4: (a) Calculated values of Voellmy pressure p_Ξ in comparison to C_d=1 and C_d=2 as a function of the arrival velocity. Voellmy pressures are lower than C_d=2. (b) The mean compaction density $\overline{\rho}_\Xi$ as a function of the arrival velocity. The final compaction density is given by ρ_Ω.

6.4 Wide Obstacles: Work-Energy Method

We consider a dense avalanche Φ with velocity $V_\Phi = \|\vec{u}_\Phi\|$, height h_Φ and bulk density ρ_Φ impacting a rigid structure. The structure of width w is positioned at the position $x = 0$; the positive x-direction defining the upstream direction of the pile-up (Figure 6.5). For simplicity we assume the avalanche strikes the structure with a mean depth-averaged velocity and density; that is, both variables are constant over the flow height defined in the z-direction but can vary in the streamwise direction and therefore time. We do not consider the impact of the powder dust cloud Π. For now we assume that the height of the structure H is higher than the flow h_Φ i.e. there is no overtopping of the structure. The width of the flow is assumed to be larger than the width of the obstacle (no flow around the structure).

We describe the pile-up process by considering avalanche mass before and after the pile-up (Figure 6.5). All the incoming mass is piled-up. The avalanche is divided into "compacting" avalanche snow, the *Knautschzone* (density not yet ρ_Ξ, velocity not yet zero, time t) and "compacted" avalanche snow (region Ξ, density ρ_Ξ, no velocity, time $t + \Delta t$. The maximum possible density the snow can compact is $\rho_\Xi \leq \rho_\Omega$.

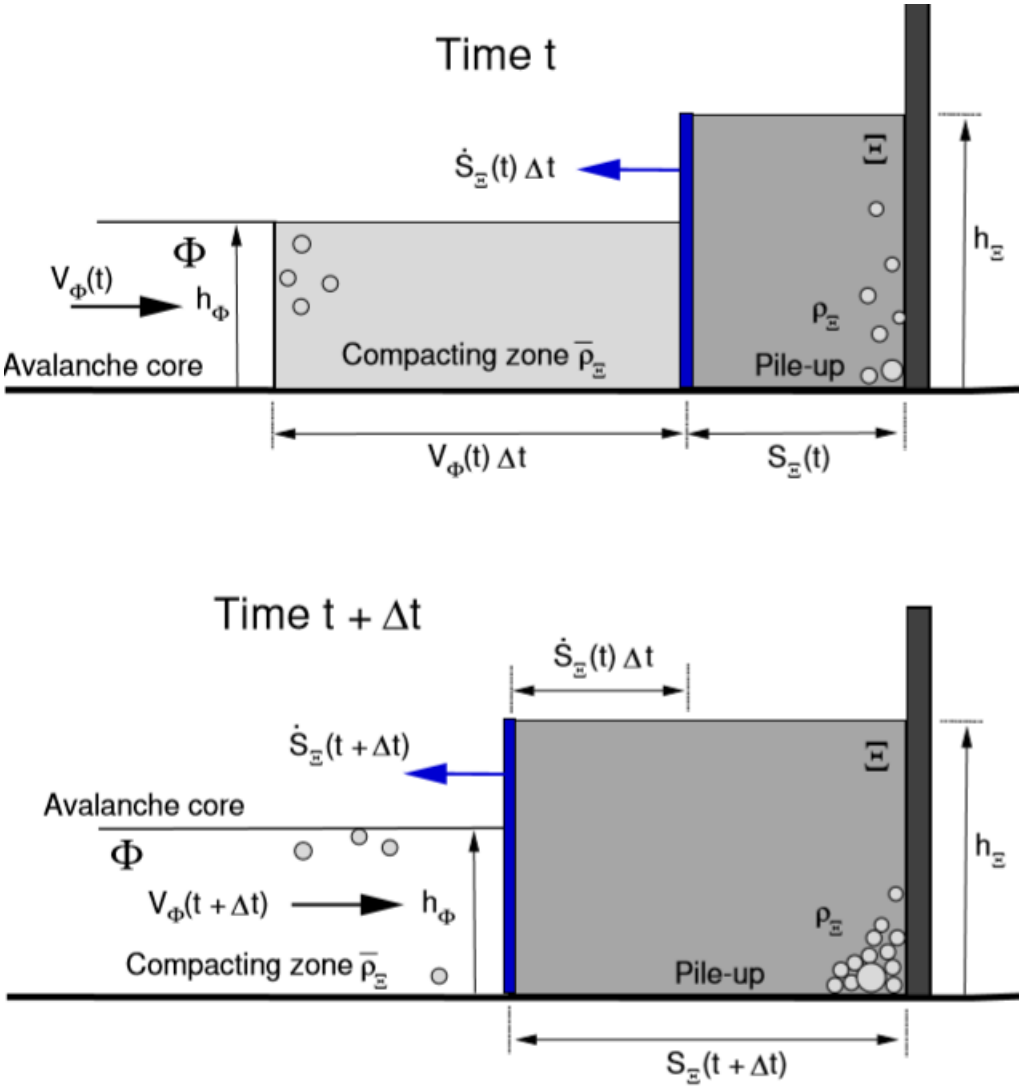


Figure 6.5: Pile-up model. An avalanche with flow height h_Φ and density ρ_Φ hits a rigid wall with velocity V_Φ. A pile-up zone Ξ with compacted snow density ρ_Ξ and height h_Ξ develops. The speed of the pile-up front is given by $\dot{S}_\Xi$. The braking length $d_{\Phi\to\Xi}$ is the distance between the location of the center-of-mass of the compacting mass and the piled-up mass.

The avalanche mass arriving at the obstacle M_Φ and stopping within the time interval Δt is

$$M_\Phi = \rho_\Phi h_\Phi w[V_\Phi \Delta t]$$

The corresponding change of kinetic energy ΔK of the avalanche is therefore

$$\Delta K = \frac{1}{2} M_\Phi \mathrm{V}_\Phi^2 = \frac{1}{2} \rho_\Phi h_\Phi w[V_\Phi^3 \Delta t]$$

The length of the compacted, pile-up zone is denoted S_Ξ, the height h_Ξ. The pile-up height might be larger than or equal to the incoming avalanche height $h_\Xi \geq h_\Phi$. Because we do not have overtopping, we assume it remains smaller than the height of the obstacle $h_\Xi < H$. The length of the pile-up zone is growing at the rate $\dot{S}_\Xi$ it is given by conservation of mass,

$$\dot{S}_\Xi = \frac{\rho_\Phi h_\Phi}{\rho_\Xi h_\Xi} V_\Phi$$

During the pile-up, the region Ξ of length $V_\Phi \Delta t$ in the x-direction compacts, increasing the length of the compaction zone S_Ξ (Figure 6.5). The difference in the locations of the center-of-mass of the compacting zone and the piled-up mass Ξ defines the braking distance $d_{\Phi\to\Xi}$ over which the incoming mass must stop,

$$d_{\Phi\to\Xi} = \frac{1}{2}\left[V_\Phi \Delta t - \dot{S}_\Xi \Delta t\right]$$

The mean force on the obstacle F_Ξ is found by equating the work-done by the braking and the change of kinetic avalanche energy in the compaction zone ΔK,

$$F_\Xi d_{\Phi\to\Xi} = [p_\Xi h_\Xi w] = \Delta K$$

or,

$$F_\Xi = \rho_\Phi h_\Phi w \left[\frac{V_\Phi}{V_\Phi - \dot{S}_\Xi}\right] V_\Phi^2$$

For now, we assume that the deceleration is constant over the **braking distance**. The impact pressure p_Ξ is found assuming the force is applied uniformly over the impact area $h_\Xi w$. Therefore,

$$p_\Xi = \rho_\Phi \frac{h_\Phi}{h_\Xi}\left[\frac{V_\Phi}{V_\Phi - \dot{S}_\Xi}\right] V_\Phi^2$$

and with the substitution of the equation for mass conservation

$$p_\Xi = \rho_\Phi \frac{h_\Phi}{h_\Xi}\left[1 - \frac{\rho_\Phi h_\Phi}{\rho_\Xi h_\Xi}\right]^{-1} V_\Phi^2$$

It is therefore possible to define a pressure factor C_d for the pile-up without deflection regime,

$$C_\mathrm{d} = 2\frac{h_\Phi}{h_\Xi}\left[1 - \frac{\rho_\Phi h_\Phi}{\rho_\Xi h_\Xi}\right]^{-1}$$

Note that the pressure factor becomes infinite when $\rho_\Phi h_\Phi = \rho_\Xi h_\Xi$. The value of C_d is plotted for various height ratios h_Φ/h_Ξ and avalanche velocities V_Φ in Figure 6.6. Of

significance is that the recommended guideline pressure factors $C_\mathrm{d} = 2$ are reproduced for a height ratio h_Φ/h_Ξ =2/3. Smaller height ratios (larger pile-up heights) result in smaller pressures. An additional feature of the work-energy method is that it can be used to provide a physical meaning to the empirical pressure coefficient,

$$C_\mathrm{d} = \frac{V_\Phi \Delta t}{d_{\Phi\to\Xi}}$$

The pressure factor is therefore the length of the *Knautschzone* relative to the braking distance $d_{\Phi\to\Xi}$.

In the foregoing derivation for the pile-up pressure p_Ξ, **the compaction density ρ_Ξ=600kg/m³ was assumed to be known and constant**. The pressure factor was plotted for different compaction densities and pile-up heights. This assumption was questioned by Voellmy who derived a relationship between the pressure p_Ξ and the mean compaction density $\bar{\rho}_\Xi$,

$$\bar{\rho}_\Xi = \frac{\rho_\Phi + \rho_\Xi}{2} = \frac{\left[\rho_\Phi + \frac{p_\Xi}{p_\Lambda}\left(1 + \frac{\rho_\Phi}{\rho_\Omega}\right)\frac{p_\Xi}{p_\Lambda}\right]}{\left[1 + \frac{\rho_\Phi}{\rho_\Omega}\frac{p_\Xi}{p_\Lambda}\right]}$$

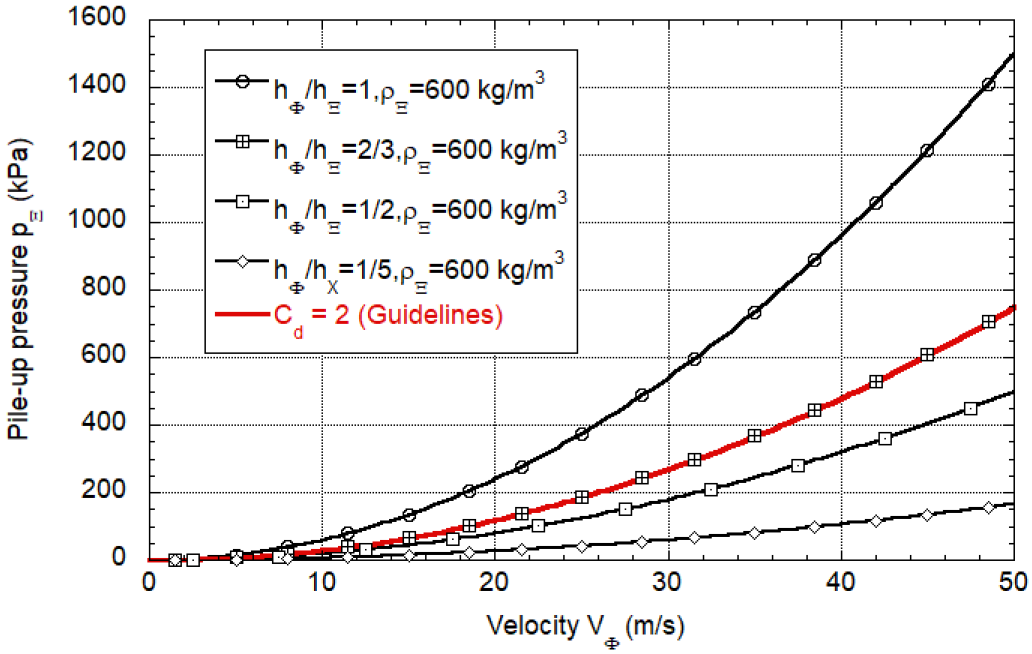


Figure 6.6: Calculated impact pressure p_Ξ for constant compaction density ρ_Ξ=600kg/m³. Variable ratios h_Φ/h_Ξ = 1.0, 2/3, 1/2 and 1/5. In red are the guideline calculations with $C_\mathrm{d} = 2$ which corresponds to h_Φ/h_Ξ = 2/3.

This relationship can now be applied in the work-energy approach by substituting it into the pressure equation. This leads to a quadratic equation for the pressure p_Ξ,

$$q_1 p_\Xi^2 - q_2 p_\Xi - q_3 = 0$$

with

$$q_1 = 1 - \frac{\rho_\Phi h_\Phi}{\rho_\Omega h_\Xi}$$

$$q_2 = \rho_\Phi \frac{h_\Phi}{h_\Xi} V_\Phi^2 - p_\Lambda \left[1 - \frac{h_\Phi}{h_\Xi}\right]$$

and

$$q_3 = \rho_\Phi \frac{h_\Phi}{h_\Xi} V_\Phi^2 p_\Lambda$$

which has the solution,

$$p_\Xi = \frac{1}{2q_1}\left[q_2 + \sqrt{q_2^2 + 4q_1 q_3}\right]$$

This equation is plotted for various height ratios h_Φ/h_Ξ and avalanche velocities V_Φ in Figure 6.7. The inclusion of a pressure dependent compaction density **slightly increases** the calculated impact pressure for the same set of assumptions (compare to Figure 6.8).

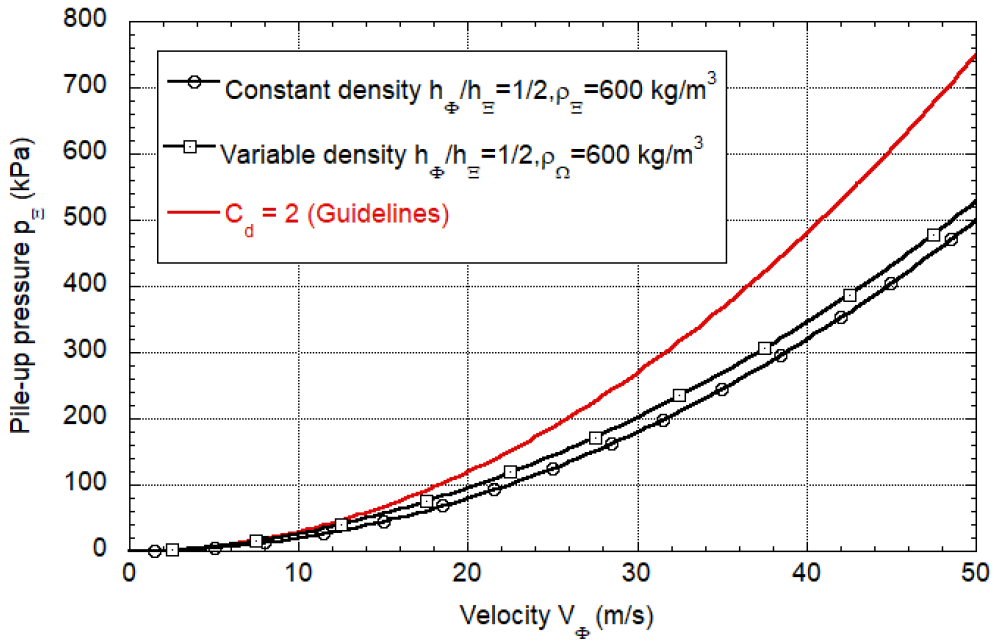


Figure 6.7: Calculated impact pressure p_Ξ for variable compaction density $\rho_\Xi = \rho_\Omega$ = 600kg/m³. Comparison to guidelines ($C_d = 2$), Voellmy and work-energy with constant compaction density (h_Φ/h_Ξ =1/2).

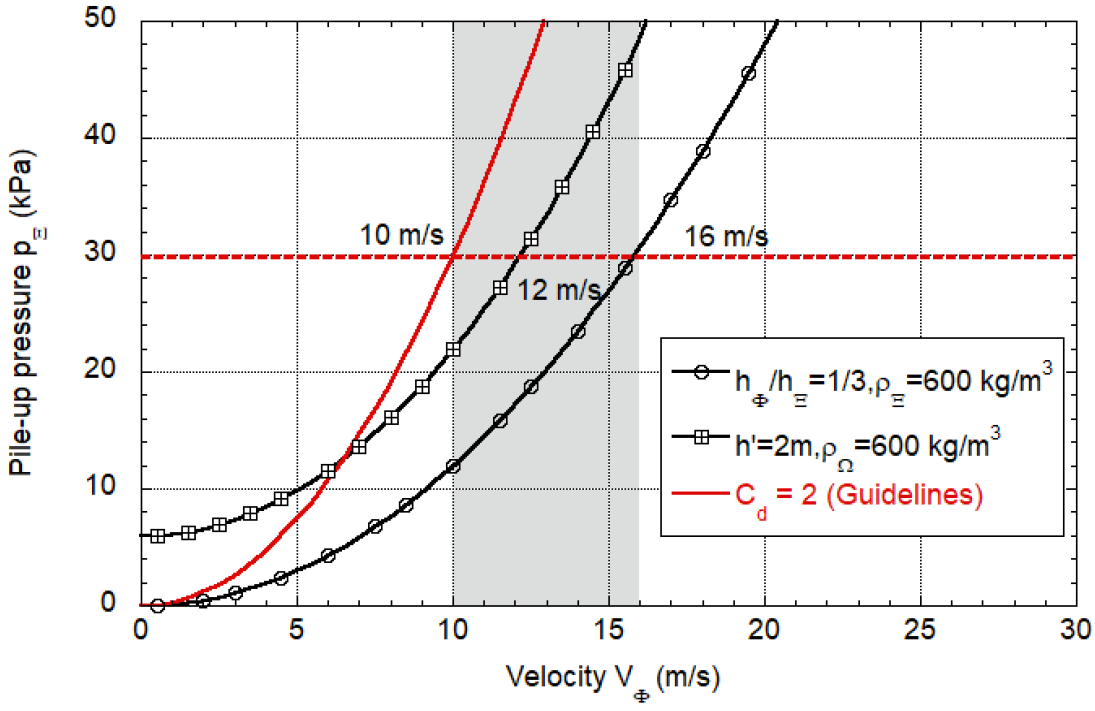


Figure 6.8: Comparison between guidelines ($C_d = 2$), Voellmy and work-energy methods as installed in RAMMS::Extended. The guidelines predict the highest pressures (30kPa at 10m/s), followed by Voellmy (30kPa at 12m/s) and the work-energy method (30kPa at 16m/s).

6.5 Wall Pressures in RAMMS::Extended

RAMMS::Extended offers three robust methods for calculating core pressures on long walls. These include the **standard method** (with $C_d = 2$, the **Voellmy method**, and the **work-energy method**, denoted as **"WorkE"**. Each approach is tailored to different modeling needs, ensuring flexibility and accuracy.

Users have two intuitive ways to access the results:

1. **The Display Tab** (see Figure 6.9): This interactive visualization tool enables users to examine the spatial distribution of maximum pressures along the wall, providing a clear and comprehensive overview of the pressure fields.

2. **The Pull-Down Menu**: Ideal for targeted analysis, this feature allows users to generate time series plots of core pressures at specific points of interest (Figure 6.10), offering detailed insights into the temporal evolution of pressures during the simulated event.

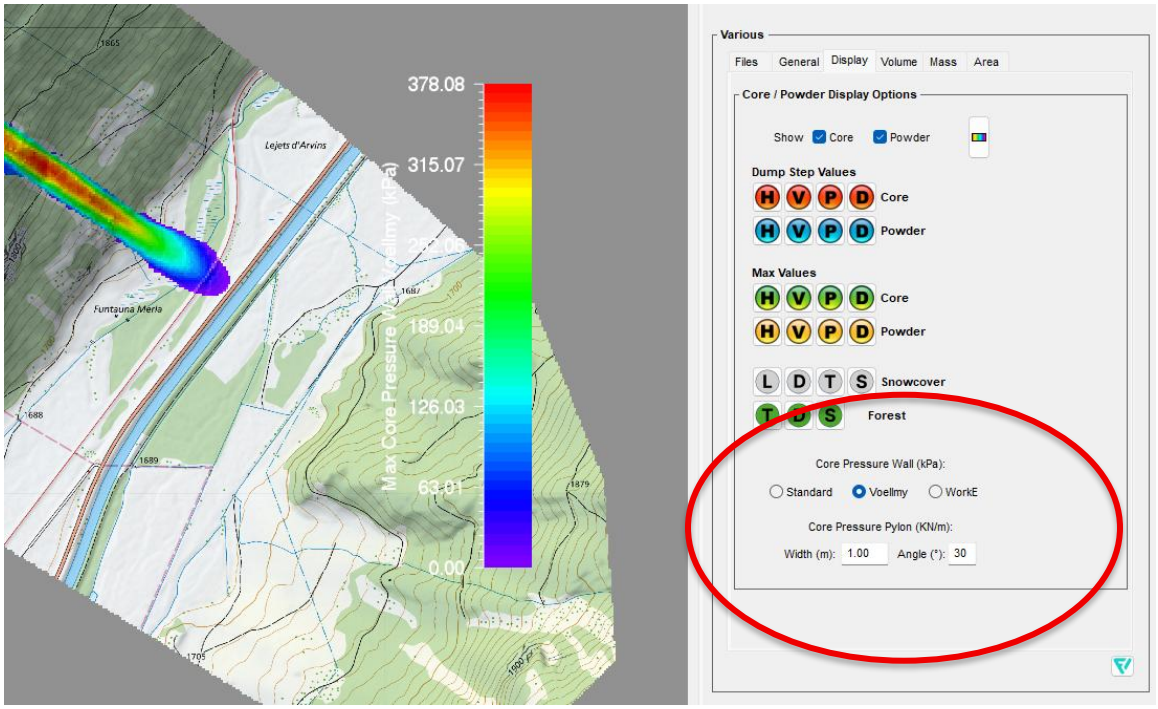


Figure 6.9: The display tab contains radio buttons and input fields to select the desired calculation method for wall pressures: Standard, Voellmy or WorkE. By selecting the green P button (max pressure) or red P button (time step), the corresponding core pressure is displayed.

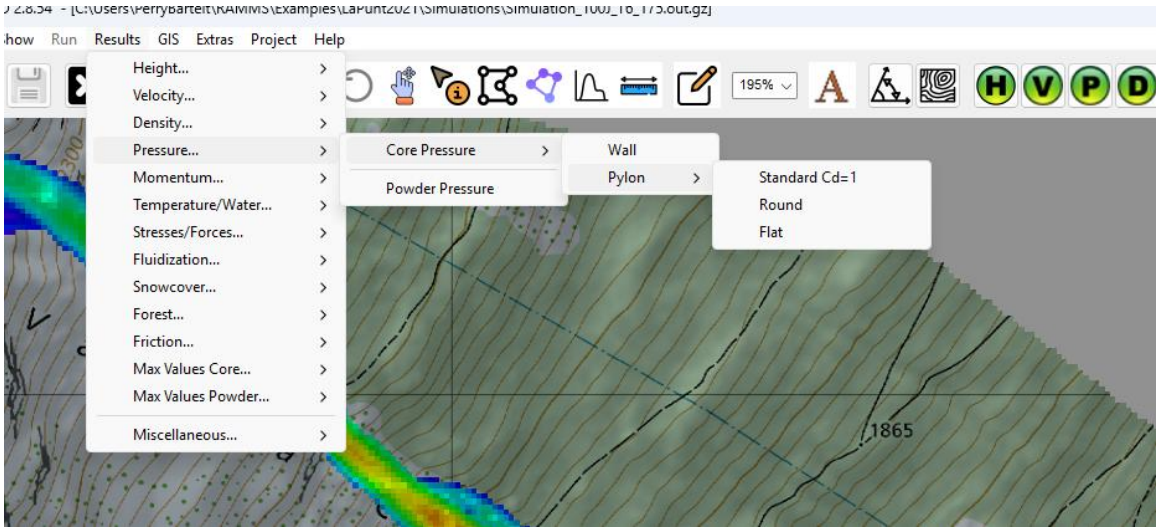


Figure 6.10: The Results pull-down menu can also be used to select either wall or pylon pressures. Wall calculation methods (Standard, Voellmy, WorkE) are selected using the Display Tab.

The maximum wall and pylon pressures can also be displayed using the **Results → Max Values Core** pull-down menu (Figure 6.11).

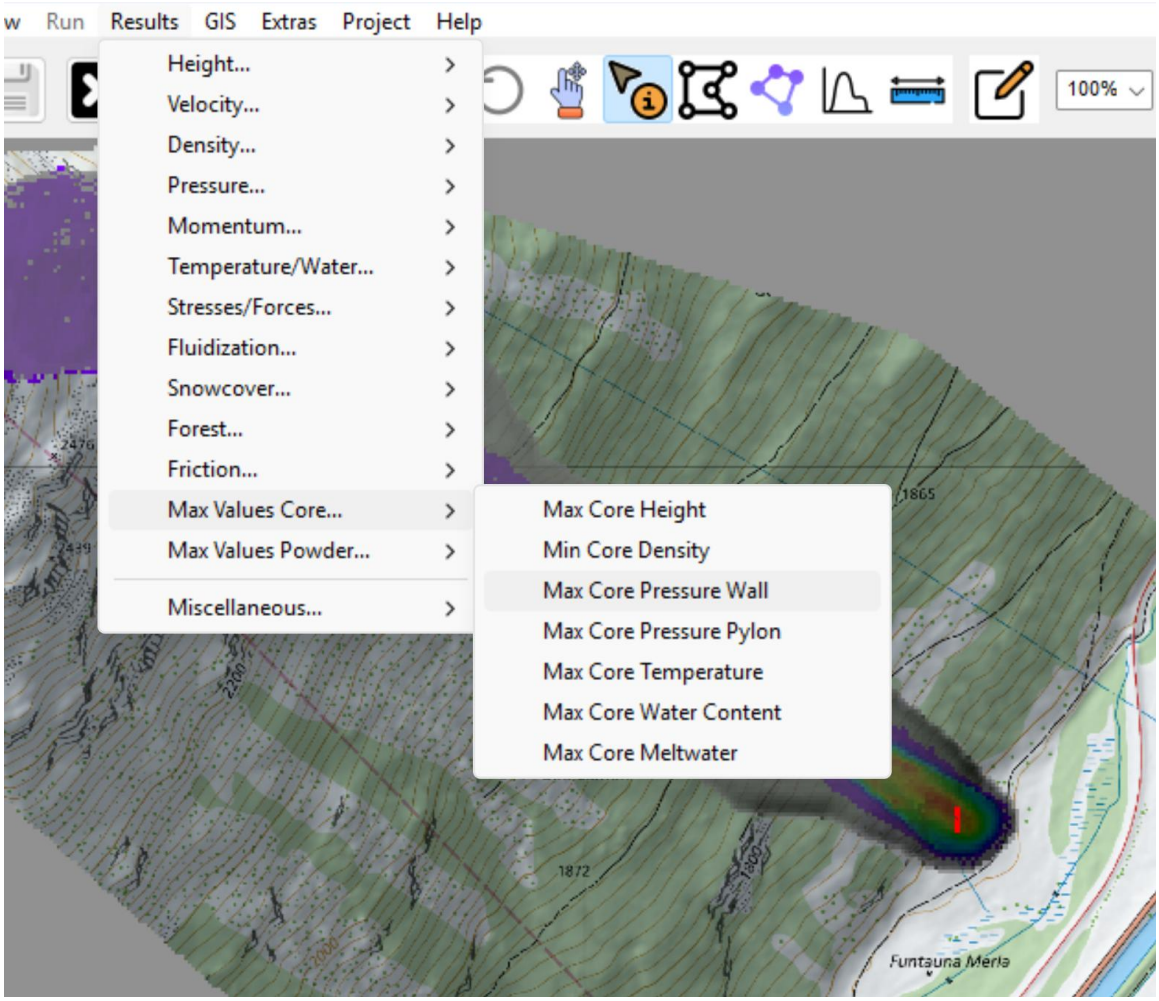


Figure 6.11: Maximum wall and pylon pressures can be depicted using the Result→Max Values Core pull down menu. Both maximum wall and pylon pressures can be selected.

These tools provide both a broad and detailed understanding of core pressure dynamics, combining spatial visualization with precise temporal data extraction.

6.5.1 Core Pressures on Walls in the Display Tab

At the lower section of the **Display Tab** (Figure 6.12), users will find three radio buttons under the heading **"Core Pressure Wall (kPa)"**. These radio buttons allow you to select the method used for calculating core pressures. The available methods are:

- **Standard** ($C_d = 2$)
- **Voellmy** (refer to Section 6.3)
- **WorkE** (refer to Section 6.4)

These buttons are ***mutually exclusive***—only one method can be selected at a time, and one selection is always required to proceed.

Figure 6.12: Selecting one of the radio buttons, Standard, Voellmy or WorkE defines which pressure will be displayed in the visualization window for wall impacts.

The selected pressure calculation method directly influences the output displayed by the **green "P" buttons**, which represent the **maximum pressure (for all time steps)**. These buttons can be found both within the **Display Tab** itself and in the **upper horizontal toolbar**. An example for the spatial distribution of Standard pressures is depicted in Figure 6.13.

In addition, the **red "P" button** output is also determined by the selected pressure calculation method, ensuring consistency in the representation of core pressures.

It is important to note that while the **Results Tab** includes a **Pressure → Core → Wall** section, it does **not** contain any subcategories for these pressure calculation methods. The selection of calculation methods (**Standard**, **Voellmy**, **WorkE**) is defined **exclusively within the Display Tab**.

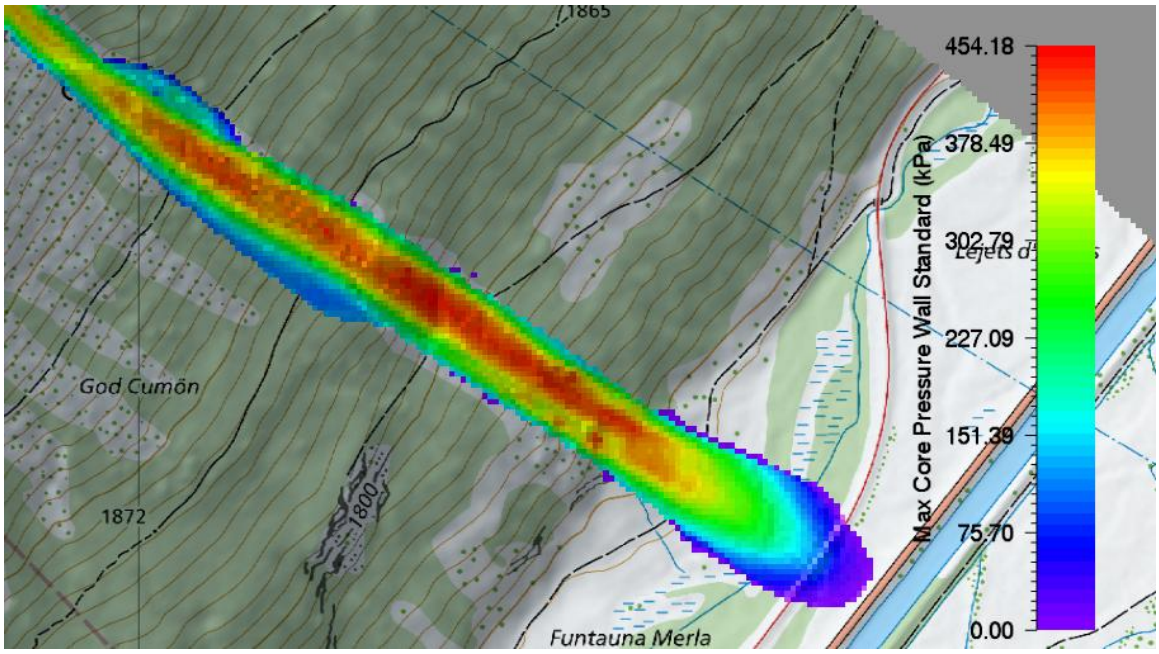


Figure 6.13: Spatial distribution of maximum core pressure, using the Standard method.

6.5.2 Numerical Pressure Values using Point-Info Icon

By leveraging the **Display** tab, users can effectively visualize **wall impact pressures**, seamlessly combining this feature with the **Point Info** tool for in-depth analysis. To extract detailed, point-specific data, simply select either the **green** or **red "P" buttons**, then pinpoint a desired location using the **Point Info icon** located on the upper horizontal toolbar.

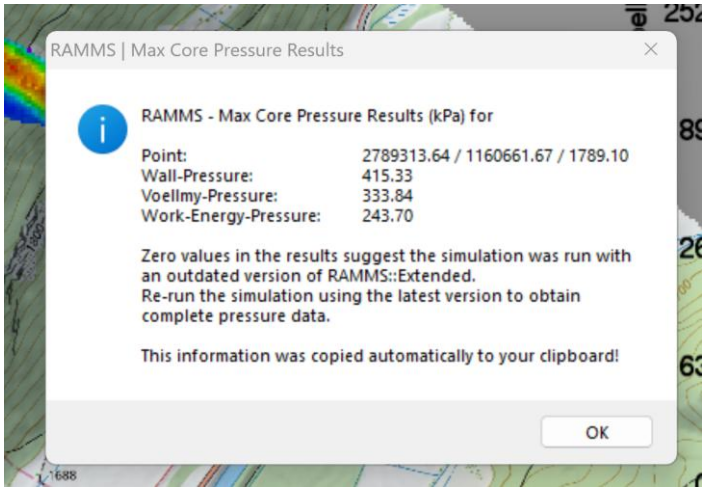


Figure 6.14: When a point is selected using the point info icon on the upper tool bar and maximum core pressures are displayed, the information window provides the coordinates of the selected points and the three different pressure values, Wall-Pressure (Standard), Voellmy and Work-Energy.

When the **green button** (which displays **maximum pressure values**) is active, a pop-up **information box** will appear. This box provides not only the **coordinates** of the selected point but also the corresponding **three distinct pressure values**, facilitating easy **numerical comparison** (refer to Figure 6.14).

Alternatively, if the **red pressure button** is activated, a **time-series plot** illustrating the **pressure evolution at the selected point** will be displayed (see Figure 6.15), offering valuable temporal insights into the pressure behavior. In this case, the three pressure values are displayed over time, but the activated method (Standard, Voellmy or WorkE) appears highlighted in red. (Recall that the resolution of the plot depends on the selected dump step interval.)

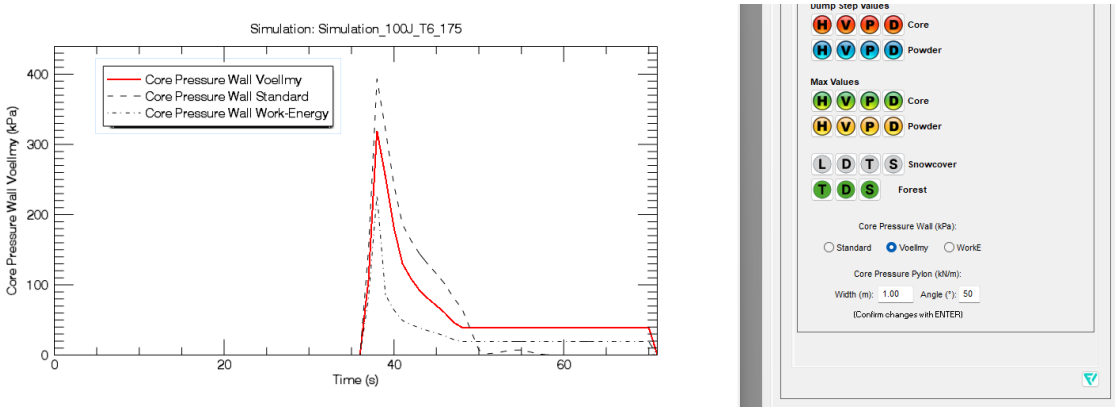


Figure 6.15: An example of a point-info pressure plot. The plot depicts the magnitude of all calculated pressures for the selected point over time. In this case Voellmy is activated in the display tab, and therefore the Voellmy pressure is highlighted in red.

6.6 Which Method for Wall Impact Pressure?

RAMMS::Extended provides three distinct methods for calculating wall impact pressures from avalanches, each tailored to specific flow regimes and design considerations. Naturally, this leads to the crucial question: **Which pressure calculation method should be applied in each scenario?**

Based on detailed analysis and practical experience, we recommend the following approaches:

1. The Voellmy Model: For Flowing Avalanches with Moderate Velocities

For flowing avalanches characterized by **bulk flow densities up to 300 kg/m³**, **flow depths exceeding 1 meter**, and **velocities below approximately 10 m/s**, the Voellmy model is the preferred choice. This method incorporates the effects of **snow compaction** upon impact and captures the influence of flow height, making it well-suited for large, cohesive flows

where compaction significantly decreases impact pressures. Our practical experience indicates that the standard approach usually over-estimates wall impact pressures.

The **Voellmy model** is especially recommended for conditions typical of the runout zone, or where the avalanche is decelerating. However, it assumes the snow is free of hard, non-compactible debris such as rocks, ice chunks, or woody material. This makes it ideal for assessing moderate pressures, up to about 30 kPa, in **clean-snow scenarios**.

2. The Standard Method (C_d = 2): For High Velocity, Dense, or Debris-Rich Avalanches

The standard calculation method (with a drag coefficient C_d = 2) produces the **highest impact pressures** and should be selected when dealing with high flow velocities (exceeding 10-15 m/s), large uncertainties, or complex debris-laden flows. It is particularly recommended in the transition zone, where velocities are highest and the flow is highly variable.

Furthermore, the standard method is especially applicable for avalanches with high flow densities—**greater than 400 kg/m³**—where the flow contains dense snow or substantial rocky and woody debris. In these cases, additional shear forces—which act parallel to the obstacle surface—become significant. Although the standard method does not directly account for these shear forces, the difference in calculated pressures between the standard and Voellmy approaches can be interpreted as reflecting these additional force components.

However, caution is warranted when applying the standard method for flow heights below 1 meter, as it does not consider particle dispersion at impact, potentially resulting in overestimated pressures.

3. The Work-Energy Method: For Dispersive, Low-Mass, Powder-Cloud Avalanches

The work-energy method is best suited for **highly dispersive, low-mass avalanches**, particularly those exhibiting saltation fronts or accompanied by powder clouds. This method captures the dynamics of avalanches with widely dispersed particles, producing the lowest pressure estimates—typically well below 30 kPa.

It is particularly useful for small avalanches with velocities under 10 m/s, where the avalanche exhibits a low-density, granular suspension behavior. The work-energy approach also provides a conservative lower-bound estimate for minimum core impact pressures.

Final Considerations: Matching Method to Scenario

The choice of calculation method depends on a combination of flow characteristics and the criticality of the infrastructure being assessed:

- **For moderate-velocity, large-volume flows of compactable snow, the Voellmy model is recommended**.

- **For high-velocity, high-density (over 400 kg/m³) or debris-rich avalanches, the standard method (C_d =2) should be used to account for higher pressures and additional shear forces**.

- **For low-density, highly dispersive flows with powder clouds and velocities under 10 m/s, the work-energy method is appropriate**.

Additionally, if the infrastructure in question is intended to shelter people, engineers should adopt a more conservative approach, potentially favoring the standard method to account for uncertainties and maximize safety. These results are summarized in Table 6.2.

Method	Flow Density	Flow Height	Flow Velocity	Debris Content	Pressure	Notes
Voellmy Model	≤ 300 kg/m³	> 1 m	< 10 m/s	Clean snow (minimal hard debris)	≤ ~30 kPa	Captures snow compaction and height effects; best for large, cohesive flows.
Standard Method (Cd=2)	> 400 kg/m³	Any (watch <1 m)	~10 m/s or higher	High debris (rocks, wood, ice)	Potentially > 30 kPa	Conservative; accounts for higher pressures and shear forces in dense/debris flows.
Work-Energy Method	Low-mass, highly dispersive	Any	< 10 m/s	Powder clouds, saltation fronts	Lower end (< 30 kPa)	Best for low-density, granular suspension flows; provides lower-bound estimates.

Table 6.2: Recommended ranges for selecting pressure method.

6.7 Uplift Forces

One of Voellmy's key insights was the recognition that buildings struck by avalanches are often not destroyed by direct horizontal impact alone, but rather by powerful *lifting forces* (German: *Auftriebskräfte*). These vertical forces arise during the pile-up phase of an avalanche, when rapidly moving snow is abruptly deflected upward, generating **intense upward pressure beneath structures** (Figure 6.16). Since most buildings—particularly their foundations and roofs—are not engineered to withstand such vertical loads, these lifting forces can effectively "uproot" the entire structure, allowing it to be carried away by the avalanche flow. Voellmy's stagnation model is mostly built around this concept: that the pile-up of snow produces secondary vertical forces, which act precisely on the most vulnerable elements of a building's design (Figure 6.17).

Figure 6.16: On January 18th, 2017, the Rigopiano Hotel was destroyed by a powerful avalanche. The force of the avalanche displaced the structure by at least 10 meters. Notably, the building appears detached from its foundation and is visibly tilting backward.

To approximate the upward pile-up velocity w_Ξ at the height z (for example, at the height of the roof), it is recommended to apply the energy formula,

$$w_\Xi = \sqrt{2g\left(h_\Phi + \frac{V_\Phi}{2g} - z\right)}$$

The lifting pressure q_Ξ is given by,

$$q_\Xi = \rho_\Xi \frac{w_\Xi^2}{2}$$

In this formula Voellmy recommended the use,

$$\rho_{\Xi} = \left(\rho_{\Phi} + \rho_{\Omega}\,\frac{p_{\Xi}}{p_{\Lambda}}\right)\Big/\left(1 + \frac{p_{\Xi}}{p_{\Lambda}}\right)$$

for the density. Thus, the pressure q_{Ξ} is indeed a function of the stagnation pressure p_{Ξ} which can be calculated either with Voellmy, work-energy methods or empirical methods.

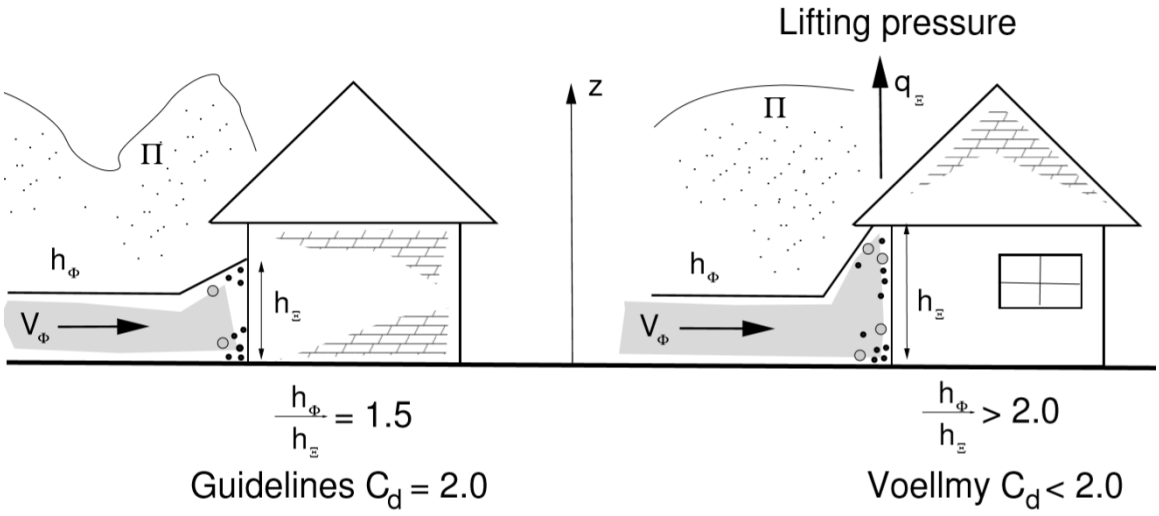


Figure 6.17: Lifting forces (Auftriebskräfte) are created during the pile-up process. The pile pressure is a function of the stagnation pressure q_{Ξ}, which in turn depends on the pile-up height h_{Ξ}. The larger the pile-up height h_{Ξ}, the smaller the horizontal pile-up pressure p_{Ξ}, but the larger the danger vertical, lifting pressures, q_{Ξ}. To obtain the pressures of the Swiss guidelines ($C_d = 2$) requires relatively small pile-up rations $h_{\Phi}/h_{\Xi} \leq$.1.5.

6.8 Thin Obstacles with Deadzone Formation

To derive the **forces of deflection**, we consider a **dense avalanche** Φ impacting **a rigid**, **thin** structure with velocity $V_{\Phi} = \|\vec{u}_{\Phi}\|$, height h_{Φ} and bulk density ρ_{Φ}. During the initial impact, a prismatic wedge of compacted snow forms on the **flat face** of the structure and the subsequent avalanche flow is **deflected around the obstacle** (Figure 6.18). The compacted snow wedge is often termed the **deadzone** in front of the obstacle. We assume symmetry (a perfect, perpendicular impact with the obstacle face). The structure of half width w_0 is positioned at x=0; the positive x direction defining the incoming avalanche direction; the y direction is defined along the width of the obstacle. The total width of the obstacle is therefore $2w_0$.

In the time interval Δt the incoming mass of the avalanche core $M_\Phi(t)$ is partitioned into two parts. The mass that is deflected $M_\Upsilon(t)$ and the mass that is piled up in a triangular wedge $M_\Delta(t)$

$$M_\Phi(t) = M_\Delta(t) + M_\Upsilon(t)$$

Figure 6.18: Example of wedge-shaped pile-up at the Vallée de la Sionne test site in 1998. A wet snow avalanche hit a 3.5m wide wall with an approximated velocity of 5m/s. The compacted snow densities in the wedge ranged from 500kg/m³ to 550 kg/m³. The wall is 7m high. The wall was later destroyed by large avalanches that occurred several months later in January and February 1999. Note the symmetry of the wedge. The wedge angle appears to approximately $\alpha_1 = 30°$ ($2\alpha_2$=60°). Picture Gauer, SLF.

We begin by considering the case when the wedge has been completely built. In this case, $M_\Delta(t)$ = 0 and all the mass approaching the obstacle is being deflected by the deadzone

$$M_\Upsilon(t) = \rho_\Phi h_\Phi w_0 V_\Phi \Delta t$$

Within the time interval Δt the avalanche snow impacts the wedge and **changes direction**. Because of the impact, **the avalanche snow also compacts**, changing its density from the flow density ρ_Φ to a somewhat **higher density** ρ_Υ. Again, because of the interaction with the wedge, the **center-of-mass of the incoming snow is deflected outwards**. At the end of the time interval Δt, the incoming avalanche mass is no longer in the shape of a

rectangular, rather a compressed triangle, with the least mass at the top of the wedge, and the most mass located near the obstacle edge (Figure 6.19).

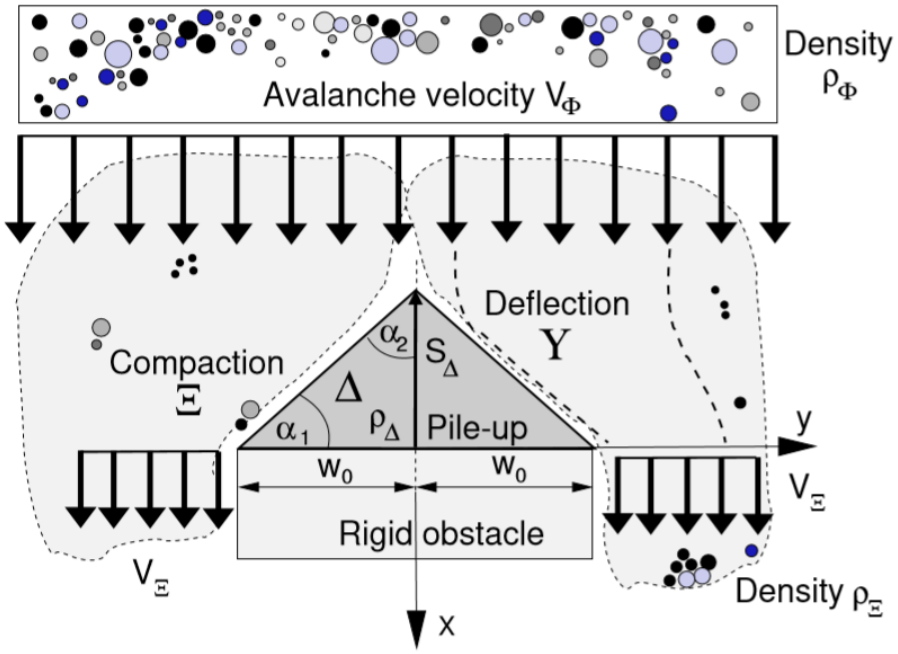


Figure 6.19: Definition of wedge-shaped pile-up with deflection plus compaction in front of the obstacle. The avalanche approaches the obstacle in the positive x -direction with velocity V_Φ. The bulk density of the flow is ρ_Φ. A pile-up wedge Δ is formed. The density of the piled-up snow is ρ_Δ. The length of the wedge in the flow (negative) x-direction is S_Δ. The wedge geometry is defined by the angle α_1 with $\alpha_2 = 90° - \alpha_1$.

The **compacted avalanche snow** moves with the velocity V_Υ

$$V_\Upsilon = V_\Phi \cos\alpha_2$$

parallel to the surface of the wedge. From mass continuity, the height of the compacted triangle is therefore,

$$l_\Upsilon = \frac{\rho_\Phi h_\Phi w_0 V_\Phi}{\rho_\Upsilon V_\Upsilon h_\Phi} = \frac{\rho_\Phi w_0}{\rho_\Upsilon \cos\alpha_2}$$

The distances d_x and d_y the center-of-mass of the incoming avalanche snow is deflected in the x and y-directions are (Figure 6.20)

$$d_x = \frac{1}{2} V_\Phi \Delta t + S_\Delta - \frac{S' \sin\alpha_1}{3} - \frac{l_\Upsilon \sin\alpha_2}{3}$$
$$d_y = \frac{1}{2} w_0 - \frac{S' \cos\alpha_1}{3} + \frac{l_\Upsilon \cos\alpha_2}{3}$$

where S' is the length of the wedge, $S' = \sqrt{{S_\Delta}^2 + {w_0}^2}$. The time Δt that is required to deflect the center-of-mass can be found by noting that at the beginning of the deflection

there is no velocity in the y and at the end of time interval, the velocity in the y-direction is $\mathrm{V_Y}\cos\alpha_1$. Therefore,

$$\Delta t = \frac{2d_{\mathrm{y}}}{\mathrm{V_\Phi}\cos\alpha_2\cos(\alpha_1-\alpha_3)}$$

The factor 2 arises because we must consider the mean speed in the y-direction during the deflection. The angle α_3 corrects for the fact that the center-of-mass velocity has slightly different flow angle than the exit velocity, which flows parallel to the wedge. This angle can be found by noting

$$\tan(2\alpha_3) = \frac{l_\Upsilon}{S'}$$

This apparently harmless angle correction is important because it leads directly to zero shear (parallel) forces on the wedge, in the absence of frictional effects. The change in kinetic energy of the deflected mass in the x and y-directions are

$$\Delta K_x = \frac{1}{2}M_\Upsilon V_\Phi^2[1-(\cos\alpha_2)^2(\sin\alpha_1)^2]$$

and

$$\Delta K_y = \frac{1}{2}M_\Upsilon V_\Phi^2[(\cos\alpha_2)^2(\cos\alpha_1)^2]$$

From which it is possible by the work energy method to determine the forces on the wedge in the x and y-directions

$$F_{\Upsilon x} = \frac{\Delta K_x}{d_{\mathrm{x}}} = \frac{\rho_\Phi V_\Phi^2}{3}\left[\frac{\mathrm{b_1}+\mathrm{b_2}}{\mathrm{b_0}}\right]w_0\csc\alpha_1$$

and

$$F_{\Upsilon y} = \frac{\Delta K_y}{d_{\mathrm{y}}} = \frac{\rho_\Phi V_\Phi^2}{3}\left[\frac{b_1+b_2}{b_0}\right]w_0\sec\alpha_1$$

with

$$\mathrm{b_0} = \left[\frac{\rho_\Phi}{3}+\frac{\rho_\Upsilon}{6}\right]$$

$$\mathrm{b_1} = \left[\rho_\Phi+\frac{\rho_\Upsilon}{2}+\left((\cos\alpha_1)^2\left(\rho_\Phi-\frac{3\rho_\Upsilon}{2}\right)-\frac{\rho_\Upsilon}{2}-\rho_\Phi\right)(\cos\alpha_2)^2\right]\sin\alpha_1$$

and

$$\mathrm{b_2} = \rho_\Phi\cos\alpha_2\,(\cos\alpha_1)^3\sin\alpha_2 = \rho_\Phi(\cos\alpha_1)^4\sin\alpha_1$$

The force of the avalanche normal to the wedge is $F_\Upsilon = \sqrt{(F_{\Upsilon x})^2+\left(F_{\Upsilon y}\right)^2}$ which can be simplified to

$$F_Y = \frac{[3\rho_Y(\cos\alpha_1)^2 + 4\rho_\Phi - 2\rho_Y]}{[2\rho_\Phi + \rho_Y]}\rho_\Phi V_\Phi^2\, w_0 \cos\alpha_1$$

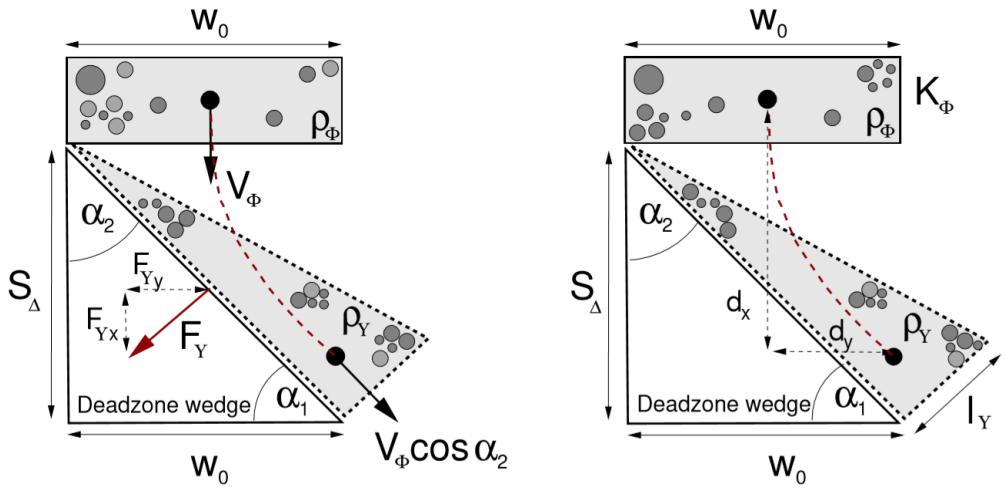


Figure 6.20: Avalanche mass is deflected around a deadzone in the shape of a triangular wedge. The wedge has height S_Δ and half-width w_0. The avalanche snow hits the wedge with velocity $V_\Phi = \|\vec{u}_\Phi\|$ and density ρ_Φ. It semi-compacts to the density ρ_Y. During the impact energy is lost; the velocity of the snow moving along the wedge side is $V_Y = V_\Phi \cos\alpha_2$.

The force parallel to the wedge is zero. If a sensor was placed behind the pile-up wedge on the face w_0 it would measure the pressure,

$$p_Y = \frac{F_Y \cos\alpha_1}{w_0} = \frac{[3\rho_Y(\cos\alpha_1)^2 + 4\rho_\Phi - 2\rho_Y]}{[2\rho_\Phi + \rho_Y]}\rho_\Phi V_\Phi^2(\cos\alpha_1)^2$$

Calculated values of p_Y for different wedge geometries are plotted in Figure 6.21(α_1=15°, 30°, 45° and 60°). Clearly, the **flatter the wedge**, **the higher the deflection pressure** (conversely, the **sharper the wedge, the lower the deflection pressure**). Of interest in this example is the fact that the guideline pressure coefficient C_d=1, corresponds to a wedge angle of α_1=45° (Figure 6.21). **Very sharp wedges have low pressure factors.**

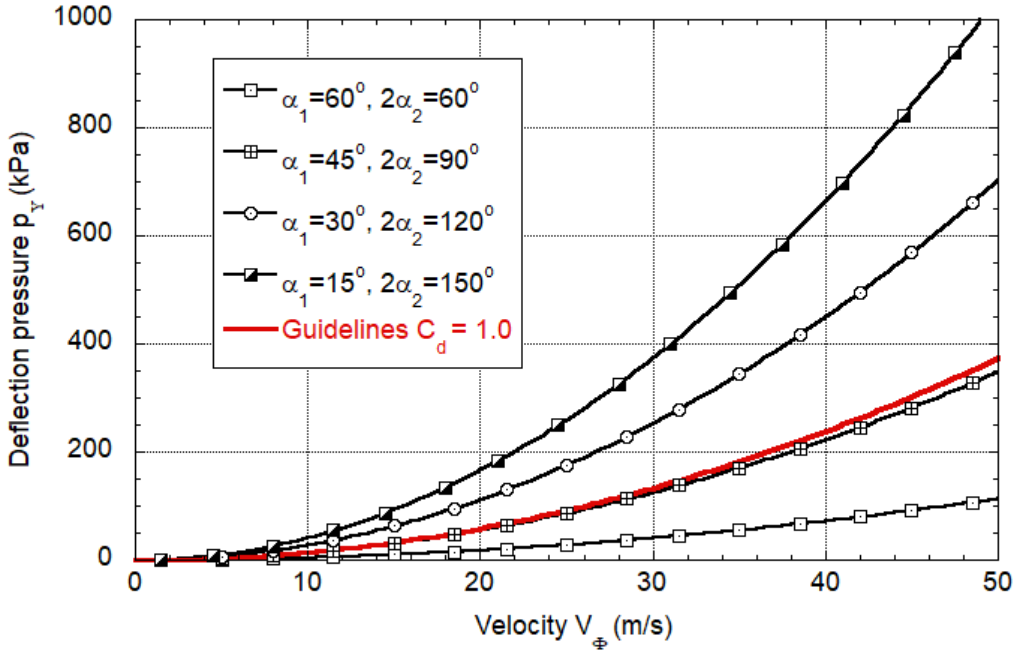


Figure 6.21: Pressure p_Y for different wedge (deadzone) angles α_1=15°, 30°, 45° and 60°. The half width of the obstacle is $w_0 = 0.5$m. The on-coming avalanche density is ρ_Φ=300kg/m³ and the compacted deflection density is ρ_Y=450kg/m³. The flatter the angle the higher the deflection forces. The guidelines C_d=1 correspond to a deflection angle of approximately α_1=45° (2α_2=90°). A sharp wedge angle can reduce the pressure on the wall.

In the derivation of the equations, we assumed that the avalanche hits the **obstacle directly and a symmetric deadzone** is formed. Because of symmetry the side forces pushing the wedge from the right- and left-hand side balance each other (Figure 6.22). When the avalanche hits the obstacle at an angle, it is not possible to construct a symmetric deadzone (unequal forces).

If we let the deflection pressure be written as

$$p_Y = \frac{1}{2}\rho'_\Phi V_\Phi^2$$

then we find an **effective density** ρ'_Φ for the deflection pressure p_Y,

$$\rho'_\Phi = \frac{2[3\rho_Y(\cos\alpha_1)^2 + 4\rho_\Phi - 2\rho_Y]}{[2\rho_\Phi + \rho_Y]}\rho_\Phi(\cos\alpha_1)^2$$

This formula can help reduce the complexity of the calculations, being easily substituted into most computational programs.

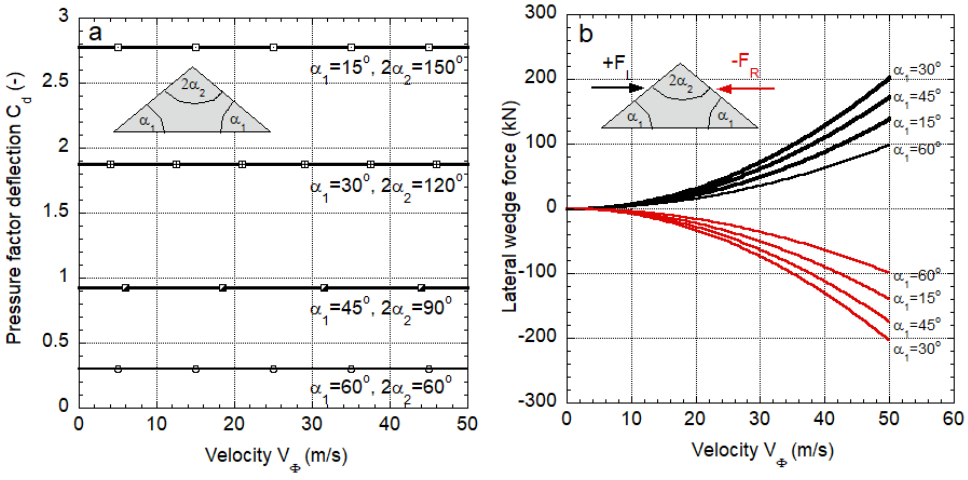


Figure 6.22: a) Calculated pressure factors for different deflection angles. The flatter the wedge, the higher the pressure factor. When the deflection angle approaches $\alpha_1 \rightarrow 0$ the pressure factor approaches $C_d \rightarrow 4.0$ in accordance with the work-energy pile-up theory with $h_\Xi = h_\Phi$. b) Lateral forces acting on the wedge. For the case of a direct impact, the forces on the left and right of the wedge are in balance. The net force is zero. This is not the case when the avalanche impacts an obstacle at an angle.

6.9 Circular, Thin Pylons

The **structural integrity of circular steel pylons**, widely used in ski lifts and aerial cableways, depends on accurately calculating avalanche loads. These pylons are typically designed with circular cross-sections. **Engineers are primarily concerned with internal bending stresses that could weaken bolted or welded connections**, making a thorough understanding of avalanche impact forces on circular sections crucial in avalanche engineering. Similar force analyses are also applied when studying tree trunk breakage, particularly in reconstructing avalanche events based on observed damage (Figure 6.23).

For circular sections with diameters of a few meters or less, avalanche forces mainly arise from the **deflection of flow around the section**, especially in low-density, high-velocity impacts. This section examines the deflection forces acting on a circular section of radius r.

The pressure factor for circular sections is **typically given as C_d= 1.17**. Here, we derive this value based on the assumption of incompressible (non-compactible) avalanche flow and explore how snow compaction generally lowers the pressure factor.

Unlike flat surfaces (e.g., walls), circular sections pose a challenge in defining pressure (force per unit area, kPa) because the impact surface continuously changes its orientation relative to the oncoming avalanche (Figure 6.24). Instead, it is more practical to express the total force per unit flow height (kN/m). Since the deflected mass is evenly distributed around both sides of the section, the net impact force is twice the force from one side and aligns with the avalanche flow vector V_Φ.

Figure 6.23: Tree impact (circular section) with wedge formation.

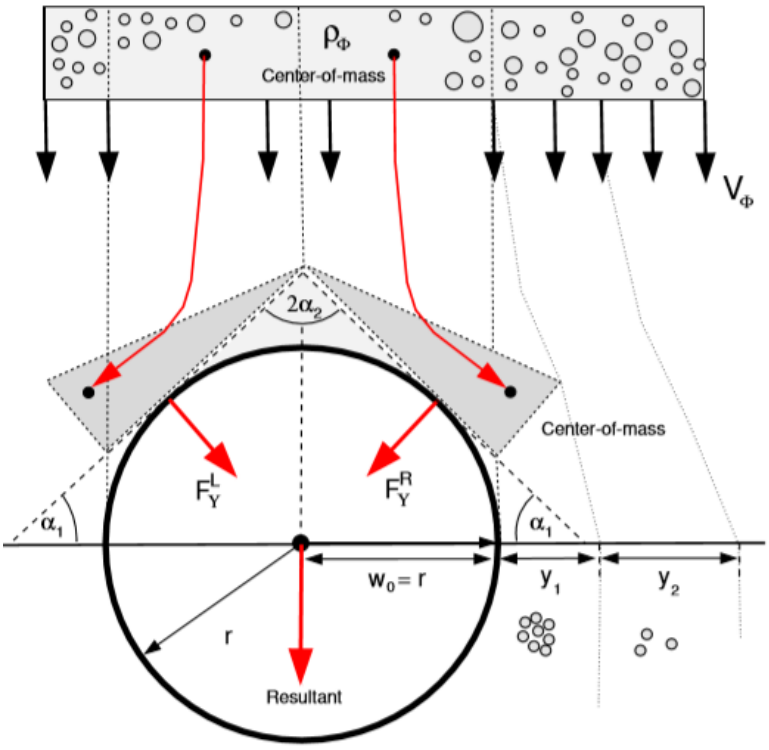


Figure 6.24: Resultant impact force on a circular section due to deflection. The model assumes a mean deflection angle $\alpha_1 = \pi/2 = 45°$, with the deflected mass determined by the section radius r.

To determine the force on one side, we analyze the center-of-mass movement around the circular arc. The deflection angle varies from $\alpha_1 = 0$ at the top to $\alpha_1 = \pi/2$ at the section's edge. Averaging these angles gives a mean deflection angle of $\alpha_1 = \alpha_2 = \pi/4$. Using this approach, we define the normal force on the section due to avalanche deflection as:

$$F_{\Upsilon}^{L} = F_{\Upsilon}^{R} = \frac{\sqrt{2}r}{2}\frac{[8\rho_{\Phi} - \rho_{\Upsilon}]}{[2\rho_{\Phi} + \rho_{\Upsilon}]}\frac{\rho_{\Phi}\mathrm{V}_{\Phi}^{2}}{2}$$

The **normal force coefficients** are given by:

$$\mathrm{b}_0 = \left[\frac{\rho_{\Phi}}{3} + \frac{\rho_{\Upsilon}}{6}\right]$$

$$\mathrm{b}_1 = \frac{\sqrt{2}}{2}\left[\frac{3\rho_{\Phi}}{4} - \frac{\rho_{\Upsilon}}{8}\right]$$

$$\mathrm{b}_2 = \frac{\sqrt{2}}{8}\rho_{\Phi}$$

These coefficients involve a combination of avalanche flow density ρ_{Φ} and compacted density ρ_{Υ}, leading to an **effective deflection density**:

$$\rho'_{\Phi} = \frac{\rho_{\Phi}}{2}\frac{[8\rho_{\Phi} - \rho_{\Upsilon}]}{[2\rho_{\Phi} + \rho_{\Upsilon}]}$$

The total force on the circular section, accounting for symmetrical deflection, is:

$$F_{\Upsilon} = (F_{\Upsilon}^{L} + F_{\Upsilon}^{R})\cos\alpha_1 = \sqrt{2}F_{\Upsilon}^{L} = \sqrt{2}F_{\Upsilon}^{R}$$

It is important to validate these theoretical results against empirical formulas commonly used in snow engineering. Standard engineering practice applies a pressure factor C_d = 1 to problems involving snow deflection around thin obstacles. The work-energy approach reproduces this result for a specific combination of densities: ρ_{Φ} = 300 kg/m³ and p_{Υ} = 400 kg/m³, as shown Figure 6.25.

For incompressible/non-compactible cases ($\mathrm{p}_{\Phi} = \mathrm{p}_{\Upsilon}$ = 300 kg/m³), **we find C_d = 1.1666**, aligning closely with the recommended value of C_d = 1.17. However, when **compaction effects are significant** (p_{Υ} = 500 kg/m³), the pressure factor drops to C_d = 0.86. **Greater compaction of avalanche snow leads to lower deflection pressures on circular sections.**

The resultant force increases with the section radius r, though the rate of increase is highly dependent on avalanche velocity (Figure 6.26). At higher velocities, the increase in force with radius is more pronounced.

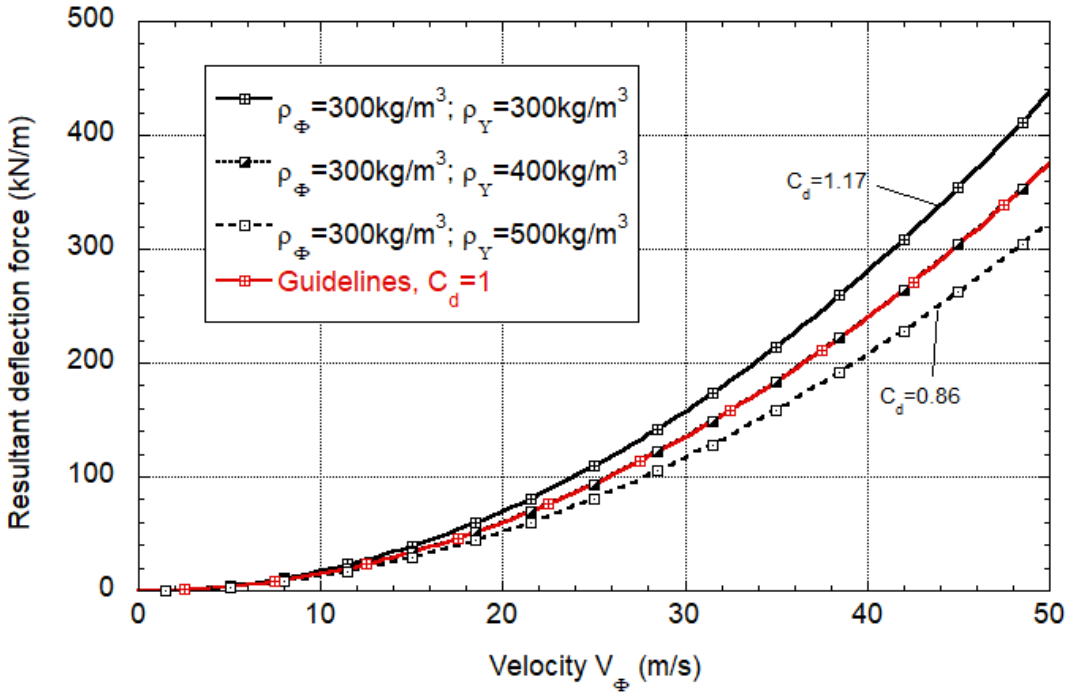


Figure 6.25: Resultant force on a circular section (r = 0.5 m) as a function of avalanche velocity and varying compaction densities $\mathrm{p_Y}$.

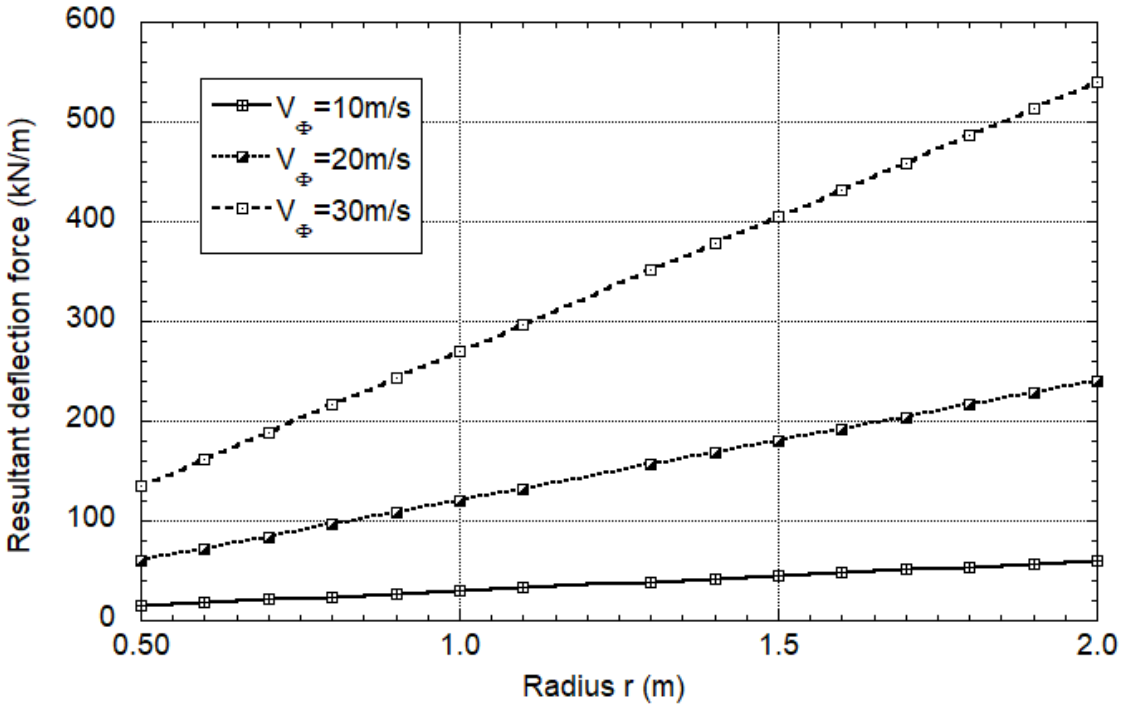


Figure 6.26: Resultant force on a circular section as a function of radius (0.5 ≤ r ≤ 2.0 m) for different arrival velocities V_Φ.

6.10 Thin Obstacles in RAMMS::Extended

RAMMS::Extended provides three distinct methods for calculating the **impact loading** (expressed in **kN/m**) on thin structures such as pylons, masts, or poles. These methods are designed to capture the complex interactions between avalanche flows and slender obstacles, each based on different flow and structural characteristics.

A fundamental distinction between **impact loading on thin structures** and **wall pressure calculations** lies in the **units and interpretation** of the results.

For **wall pressures**, **RAMMS** calculates and displays results in terms of **pressure per unit area (kN/m² or kPa)**.

In contrast, for **thin structures**, the calculations yield **loading per unit height (kN/m)**—essentially, a line load along the vertical axis of the structure. This is obtained by multiplying the local pressure by the **width of the obstacle** at the point of impact.

6.10.1 Three Calculation Methods

RAMMS::Extended offers the following three calculation approaches for impact loading:

Standard Method (C_d = 1):

- This method applies an empirical drag coefficient of C_d **= 1** to estimate impact loading on thin structures.
- It provides a general-purpose estimate but does not account for specific geometry or flow deflection effects.

Loading on Thin, Flat Obstacles with Deadzone Formation:

- For thin, flat structures such as barriers or signposts, this method account for the formation of a **deadzone wedge**—a localized build-up of material in front of the obstacle that effectively **deflects the flow around it**.
- The loading is applied over the **calculated height of flow**. We **do not** assume that the flow height increases, which would reduce the impact force.
- This model applies the **work-energy theorem** to estimate the impact loading, considering both the **obstacle's width** and the **deflection angle** of the deadzone.

- Sharper, more elongated deadzones result in **lower impact pressures**, as the material is more efficiently deflected.

Loading on Round Pylons:

- Designed for circular obstacles, this method assumes a typical **deflection angle of 45°**, representing the way flow is diverted around the pylon's curved surface.

- As with the flat obstacle method, it applies the **work-energy theorem** and considers the **diameter** of the pylon to compute the line load.

An overview is provided in Table 6.3.

6.10.2 Controlling the Display and Input Parameters

While the **Display tab** in **RAMMS** allows users to toggle between the **three impact loading options** (Standard, Voellmy, Work-Energy), the actual **display of pressures** is primarily controlled through the **Results → Pressure pull-down menu** (refer to Figure 6.27).

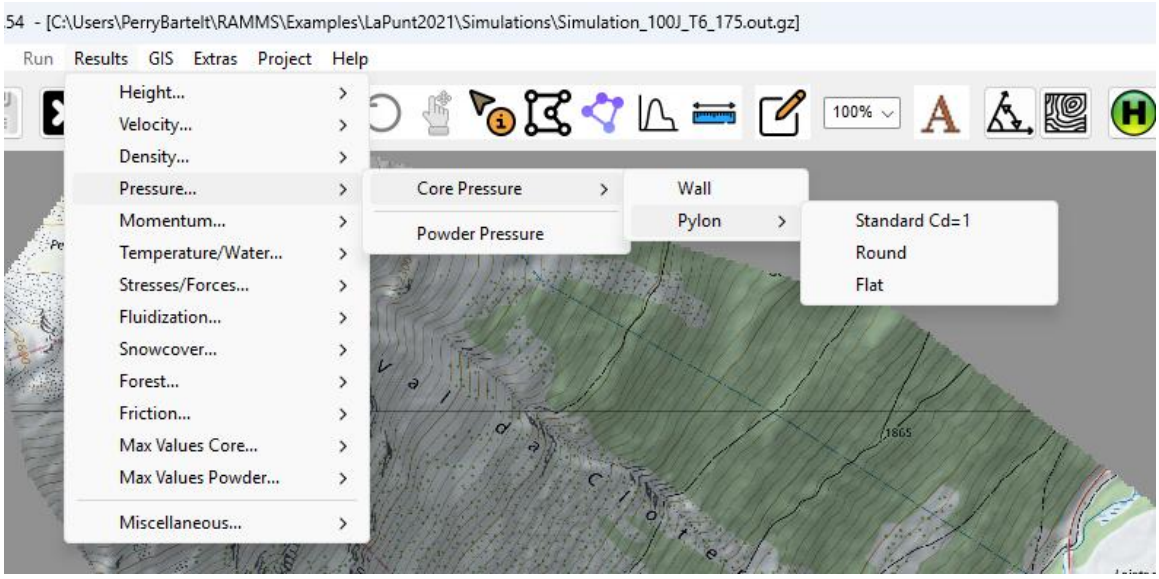


Figure 6.27: Avalanche loadings on thin obstacles are accessed via the **Results** pull-down menu.

For **thin structures**, the calculation method must also be considered:

- The **width of the obstacle** (or the **diameter** for round pylons), which directly scales the line load.

- The **deflection angle** of the flow, especially for flat structures with deadzone formation. This angle influences the **magnitude of flow diversion** and the **resulting impact load**.

These parameters — **obstacle width and deflection angle** — are entered at the **bottom of the Display tab** within the RAMMS interface (Figure 6.28).

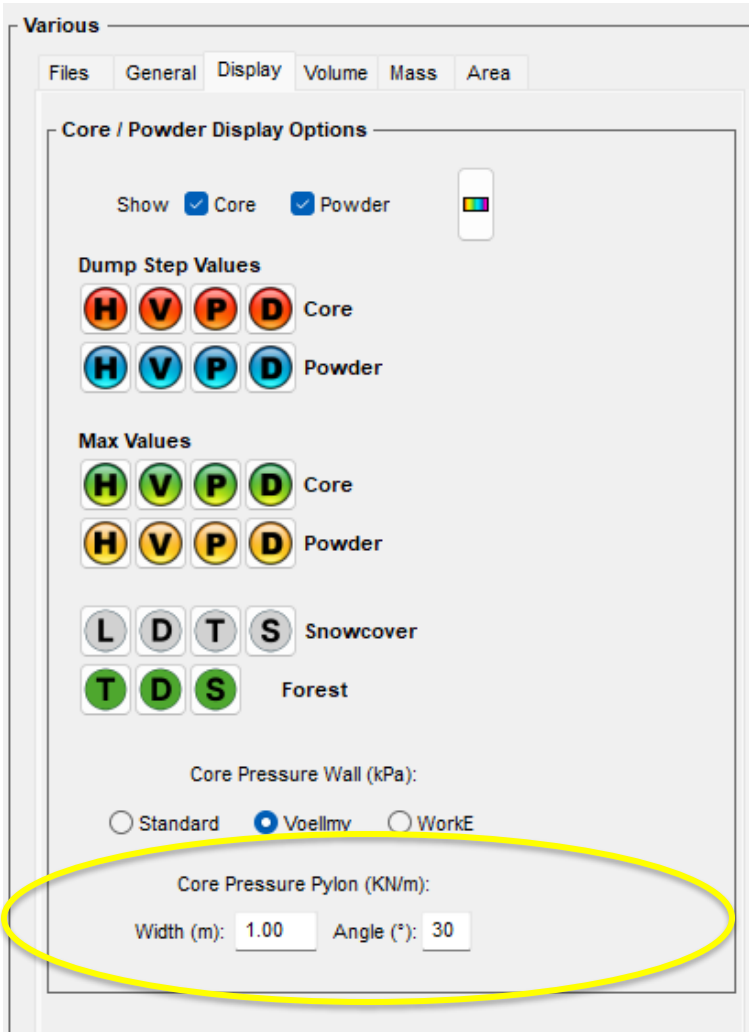


Figure 6.28: Input of thin obstacle width and deflection angle are inputted in display tab. The width is multiplied with all three calculated pressures. The **deflection angle** is only used for the **deadzone formation method**.

To make spatial depictions of impact pressures for all three calculation methods it is necessary to utilize the **Results→ Pressure** pull-down menu. The three calculation methods can be selected (Figure 6.29). The spatial depictions can be used in combination with the **time-slider to select a specific time set**. The values can be scaled with the Max/Min settings. The values can be made transparent or depicted with or without the powder cloud.

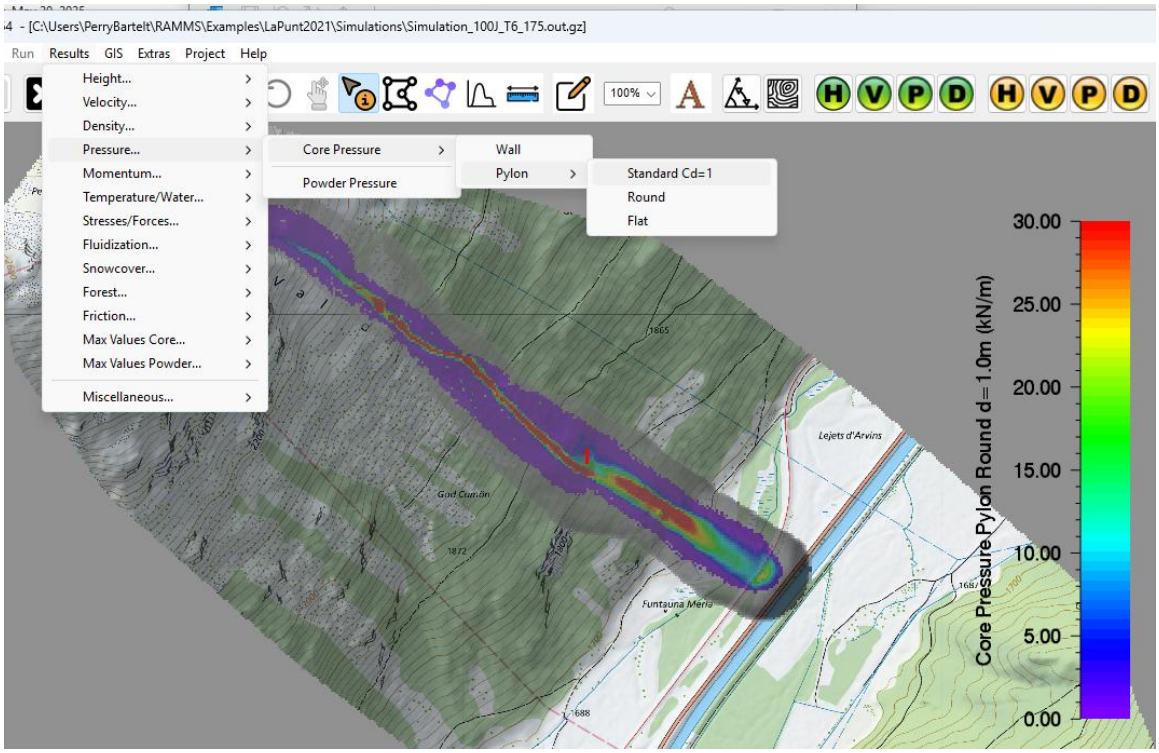


Figure 6.29: Spatial distribution of pressure on a thin object using the **Results->Pressure pull-down menu**. The values are calculated with the obstacle **width** and **deflection angle** that are input in the **Display tab**.

The spatial depictions of thin obstacle loading can also be used with the point-info button to select a specific point and generate a time plot (Figure 6.30 and Figure 6.31). These loadings will change depending on the selected obstacle width or selected deflection angle.

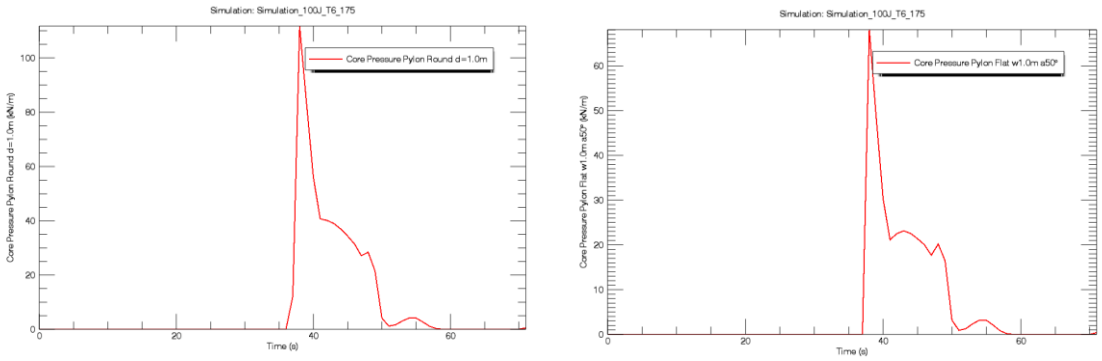


Figure 6.30: The **point-info button** can be used in combination with the **Pressure pull-down** menu to obtain impact loadings over time on thin **round** (left) or **flat** (right) pylons. In the flat case the results depend on the selected deflection angle.

Maximum core pressures for thin obstacles are **ONLY calculated** for the standard C_d=1 approach. These pressures can be visualized by using the Results->Max Values Core->Max Core Pressure Pylon **pull-down menu (**Figure 6.32**).**

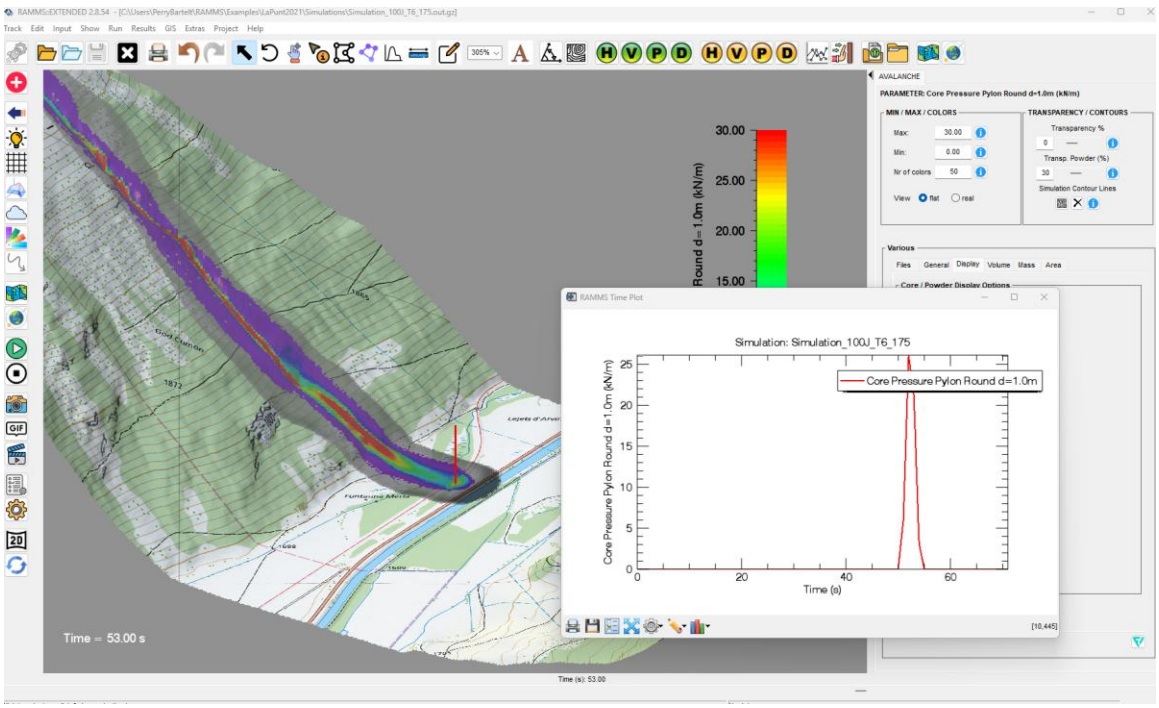


Figure 6.31: Selecting an info point and displaying the numerical values of the thin obstacle loading. The selection process is made in **2D mode**, but the results can be displayed **in 3D mode for a better dimensional visualization**. The numerical values are displayed in the time plot.

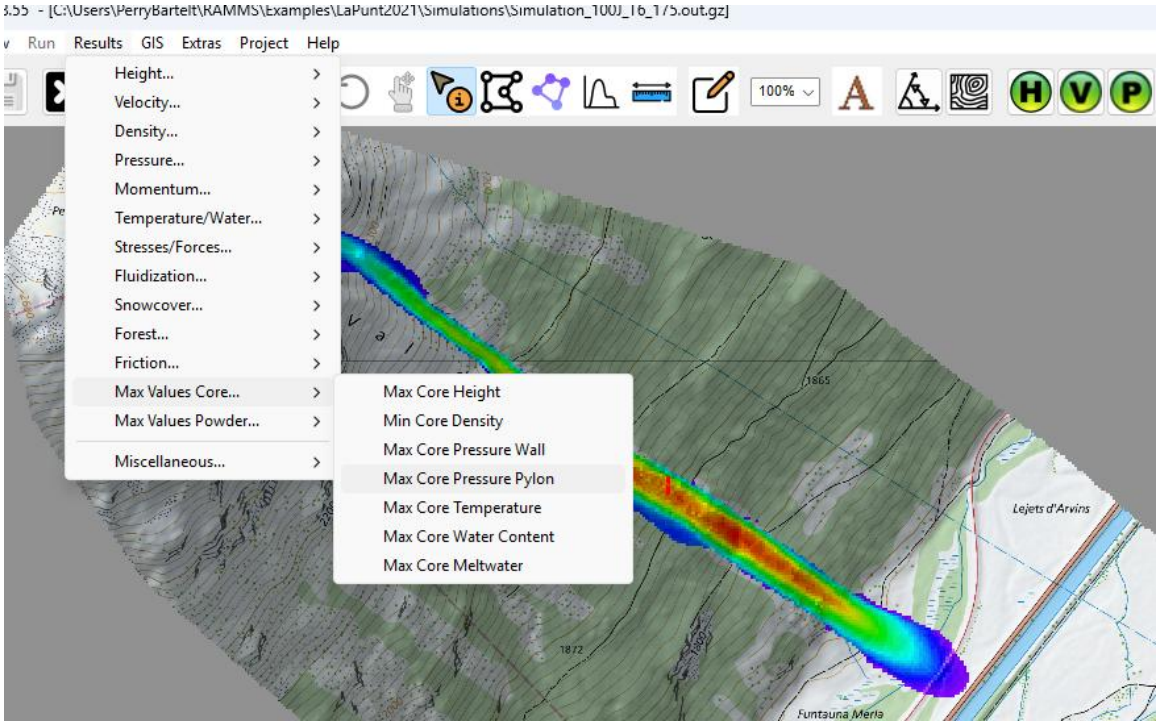


Figure 6.32: Visualization of maximum core pressures for **thin obstacles** is obtained via the **Results->Max Values Core->Max Core Pressure Pylon** pull-down menu. **ONLY** the standard method C_d=1 pressures can be visualized for maximum core pressures.

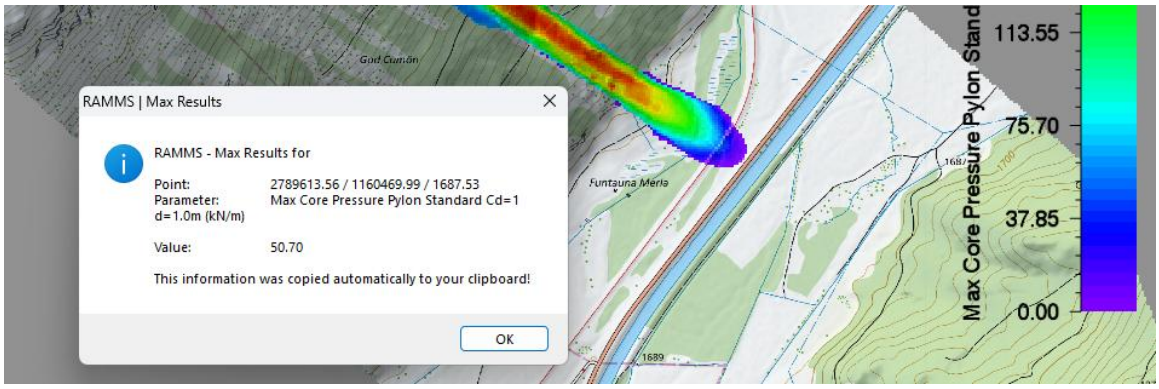


Figure 6.33: The point info button can be used to display the numerical values of the maximum core loadings acting on a thin obstacle.

Method	Obstacle	Comments
Standard Method (C_d = 1)	Thin obstacles (general)	Empirical approach, no deflection effects considered. Loading is derived directly from flow parameters with a constant drag coefficient.
Thin, Flat Obstacles with Deadzone	Flat barriers, signs, thin walls	Considers deadzone wedge formation and flow deflection. Loading depends on obstacle width and deflection angle. Calculated using work-energy principles.
Round Pylons	Circular pylons, posts	Assumes a 45° flow deflection angle around the obstacle. Loading depends on the pylon diameter. Calculated using work-energy theorem.

Table 6.3: Overview of calculation methods for thin obstacles.

6.10.3 Key Distinctions from Wall Pressure Calculations

- **Unit of Measurement**: Wall pressures are given in **kN/m² (kPa)**, whereas impact loadings for thin structures are expressed in **kN/m**.
- **Derivation**: Wall pressure calculations are generally **based on direct pressure estimates**, while impact loadings for thin structures incorporate **geometric considerations** such as width and deflection angle.
- **Methodology**: The **standard method** is empirical; the **other two methods** (for round and flat obstacles) are grounded in the **work-energy theorem** (see Sections 6.8 and 6.9 for derivation details).

An overview of the three calculation methods is provided in Table 6.3.

6.11 Direct and Oblique Impact of Splitting Wedges

6.11.1 Direct Impact

We begin by examining the normal forces acting on a splitting wedge (Figure 6.34). These forces arise from a compactive, deflection of the avalanche snow as it moves around the wedge. In our model wedge has asymmetrical sides, with different base lengths — w_o^L and w_o^R — and different deflection angles — α_1^L and α_1^R (Figure 6.35).

In this case, the avalanche flow strikes the wedge head-on. As the snow impacts and deflects along the wedge faces, it generates normal forces on both the left and right sides, denoted as F_Y^L and F_Y^R.

Due to the differing side lengths and angles, both the load distribution and average pressure vary between the two sides. Additionally, depending on the avalanche's density, shear forces may develop at the wedge corners.

Figure 6.34: A **symmetric splitting wedge** protecting a house near Davos. The cutting angle of the wedge is approximately $2\alpha_2 = 60°$.

The **deflection forces** considering the compaction of the avalanche snow can be determined directly by applying the deflection formula:

$$F_{\Upsilon}^{L} = \frac{[3\rho_{\Upsilon}(\cos\alpha_1^L)^2 + 4\rho_{\Phi} - 2\rho_{\Upsilon}]}{[2\rho_{\Phi} + \rho_{\Upsilon}]}\rho_{\Phi}V_{\Phi}^2\, w_o^L \cos\alpha_1^L$$

$$F_{\Upsilon}^{R} = \frac{[3\rho_{\Upsilon}(\cos\alpha_1^R)^2 + 4\rho_{\Phi} - 2\rho_{\Upsilon}]}{[2\rho_{\Phi} + \rho_{\Upsilon}]}\rho_{\Phi}V_{\Phi}^2\, w_o^R \cos\alpha_1^R$$

The corresponding **mean** pressures are

$$\bar{p}_{\Upsilon}^{L} = \frac{F_{\Upsilon}^{L}\cos\alpha_1^L}{w_o^L} = \frac{[3\rho_{\Upsilon}(\cos\alpha_1^L)^2 + 4\rho_{\Phi} - 2\rho_{\Upsilon}]}{[2\rho_{\Phi} + \rho_{\Upsilon}]}\rho_{\Phi}V_{\Phi}^2(\cos\alpha_1^L)^2$$

$$\bar{p}_{\Upsilon}^{R} = \frac{F_{\Upsilon}^{R}\cos\alpha_1^R}{w_o^R} = \frac{[3\rho_{\Upsilon}(\cos\alpha_1^R)^2 + 4\rho_{\Phi} - 2\rho_{\Upsilon}]}{[2\rho_{\Phi} + \rho_{\Upsilon}]}\rho_{\Phi}V_{\Phi}^2(\cos\alpha_1^R)^2$$

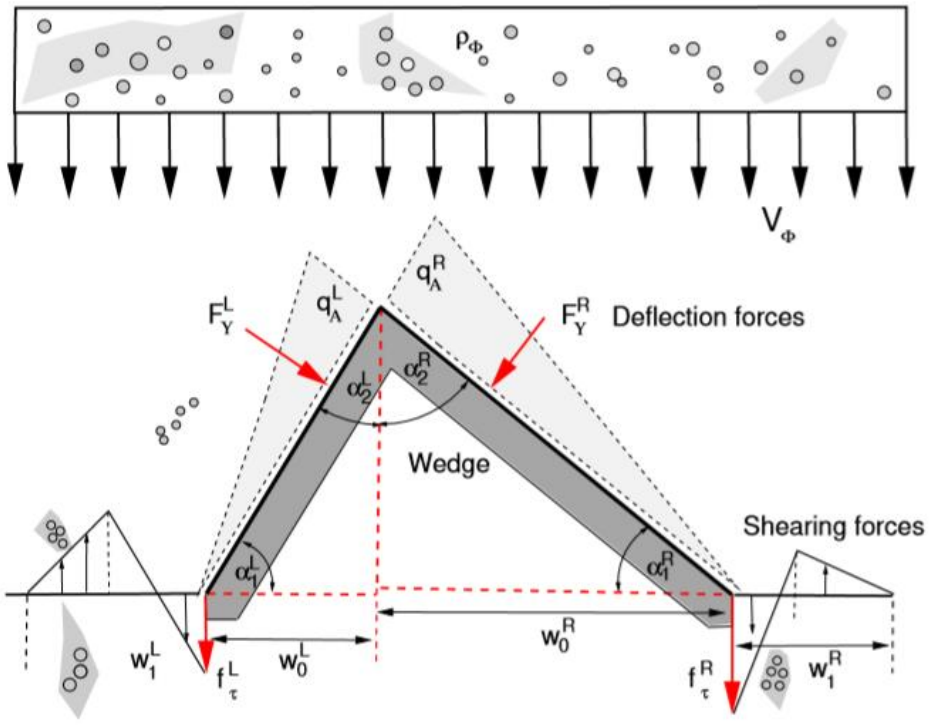


Figure 6.35: Direct impact of a non-symmetric wedge, $\boldsymbol{\alpha_1^L \neq \alpha_1^R}$. The deflective forces $\boldsymbol{F_{\Upsilon}^{L}}$ and $\boldsymbol{F_{\Upsilon}^{R}}$ act normal to the wedge faces.

Figure 6.36a illustrates the relationship between the deflection force, F_{Υ}, and the wedge angle, α_1. The trend is clear: as the wedge angle increases—meaning the wedge becomes

sharper—the resulting normal force decreases significantly. For high-velocity avalanche flows, a sharp wedge with $\alpha_1 = 60°$ can reduce the normal force by as much as a factor of six compared to a flat wedge with $\alpha_1 = 10°$. This dramatic reduction highlights the effectiveness of sharp wedge geometries in mitigating impact forces from fast-moving snow masses.

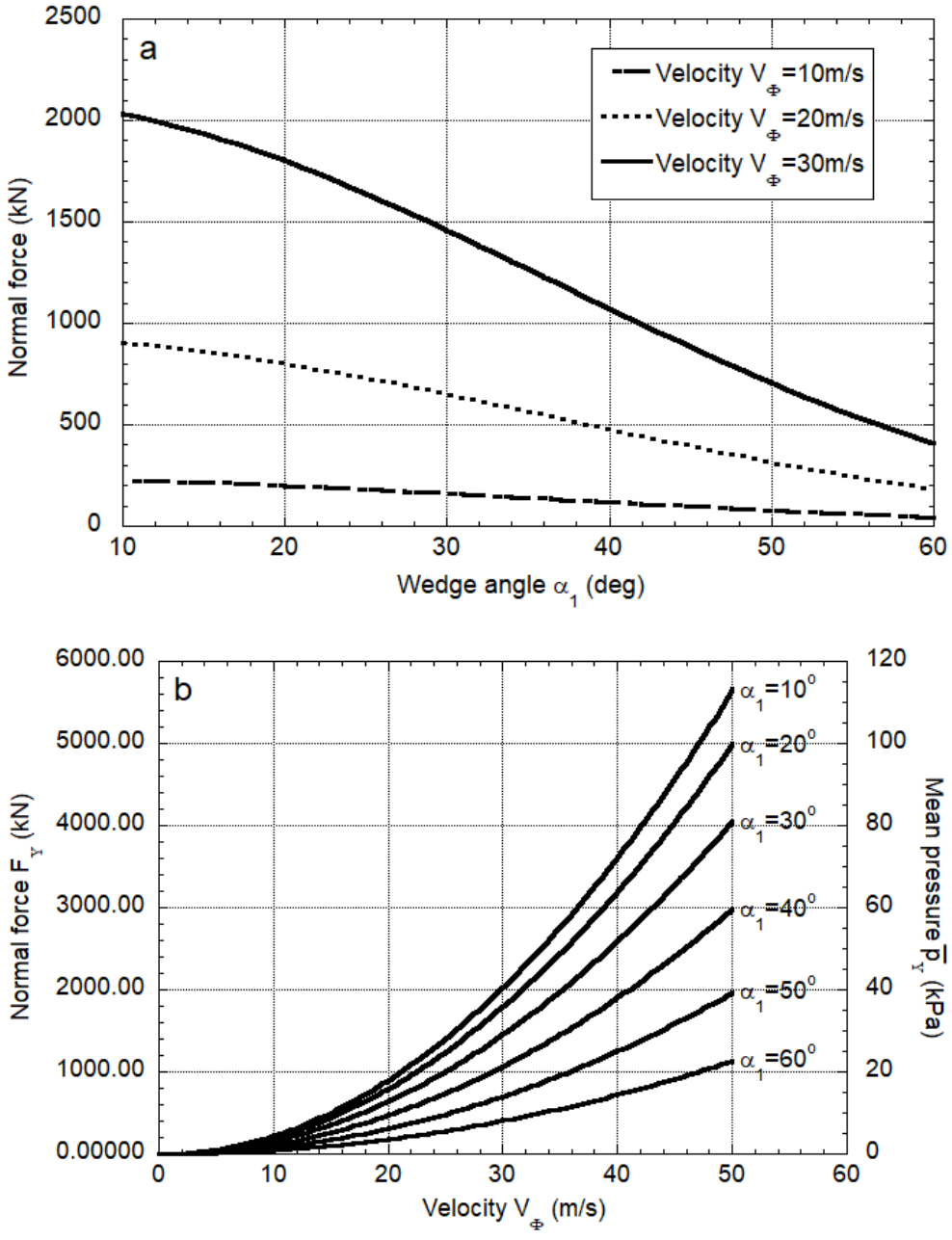


Figure 6.36: a) **Decrease in normal force with increasing wedge angle, α_1**, shown for three different impact velocities V_Φ. b) **Increase in normal force F_Y and mean impact pressure $\overline{p}_Y$ with increasing impact velocity V_Φ** for different wedge angles, α_1.

However, the benefit diminishes for slower avalanche flows, where the reduction in force is less pronounced. To further underscore this effect, Figure 6.36b presents the variation of normal force F_Υ with strike velocity V_Φ across different wedge configurations. Notably, for velocities up to 50 m/s, a sharp wedge $(\alpha_1 = 60°)$ can limit mean normal pressures to below 30 kPa—demonstrating its capacity to significantly soften avalanche impact loads.

One way to analyze the results of the deflection theory with snow compaction is to calculate the **effective impact density**

$$\rho'_\Phi = \frac{2[3\rho_\Upsilon(\cos\alpha_1)^2 + 4\rho_\Phi - 2\rho_\Upsilon]}{[2\rho_\Phi + \rho_\Upsilon]}\rho_\Phi(\cos\alpha_1)^2$$

Such that a comparison can be made to the empirical pressure formula p_Υ

$$p_\Upsilon = \frac{1}{2}\rho'_\Phi \mathrm{V}_\Phi^2$$

Figure 6.37a shows how the effective avalanche density, ρ'_Φ, varies with the wedge angle α_1. The results reveal that flatter wedges are associated with significantly higher effective densities. This occurs because the deflection equations do not account for any increase in flow height upon impact—meaning they represent conservative, upper-bound estimates of the resulting pressure.

For reference, an effective density of ρ'_Φ = 300kg/m³ corresponds to a wedge angle of approximately $\alpha_1 = 45°$. This relationship provides a useful tool for evaluating impact forces based on wedge geometry.

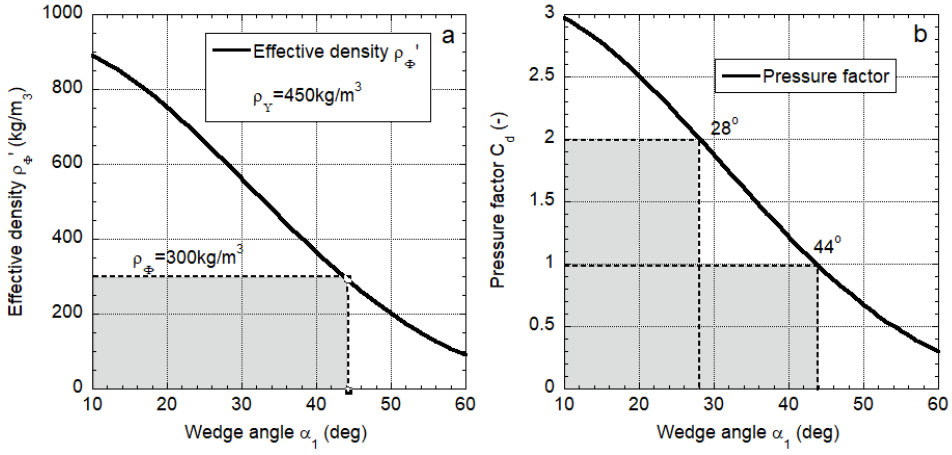


Figure 6.37: a) **Effective density ρ'_Φ as a function of the wedge angle α_1**. The flatter the wedge the larger the effective density. b) **Pressure coefficient C_d** as a function of the wedge angle α_1. The pressure factor decreases with increasing wedge angle α_1.

A similar approach can be used to assess effective pressure coefficients, as shown in Figure 6.37b. For instance, a pressure coefficient of C_d = 2 is linked to a relatively flat wedge with an angle of $\alpha_1 = 28°$, whereas a more streamlined wedge with $\alpha_1 = 44°$ yields a lower pressure coefficient of C_d = 1.

6.11.2 Oblique Impact

This section examines the effects of an **oblique avalanche impact on a splitting wedge**. The attack angle, denoted as θ, is positive when the avalanche approaches from the left and negative when approaching from the right (Figure 6.38).

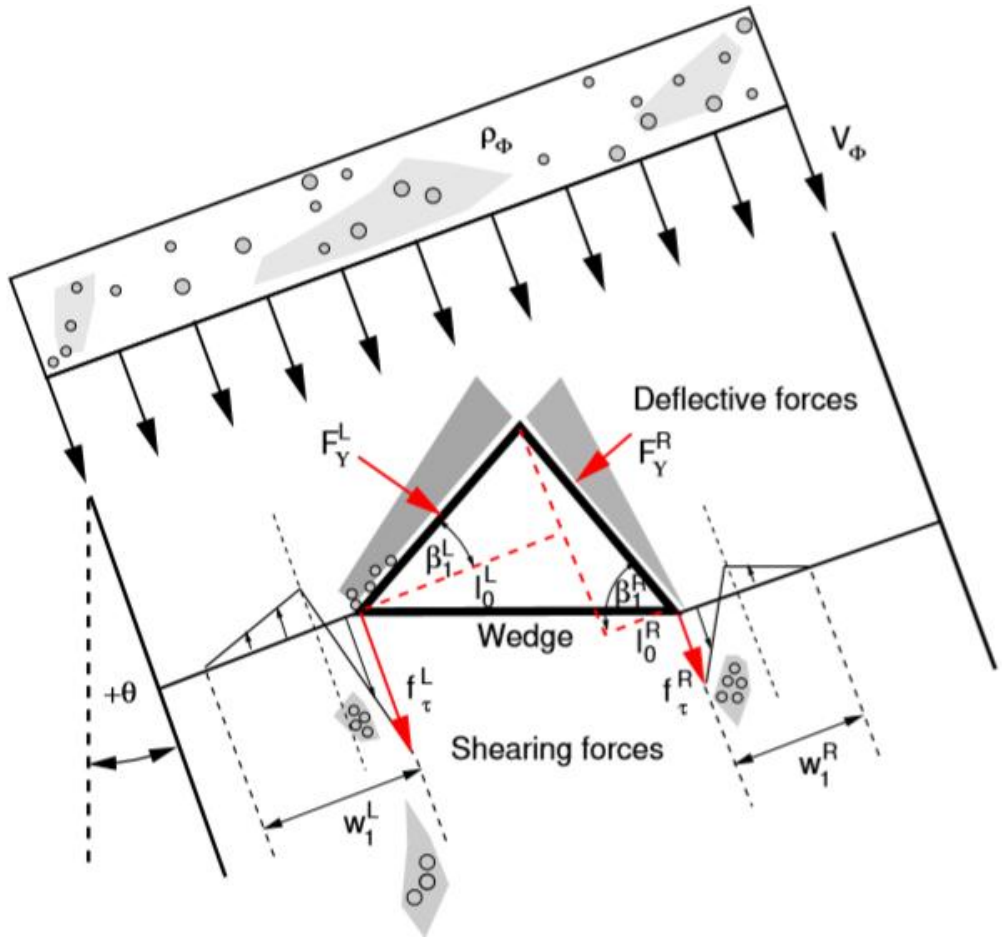


Figure 6.38: An avalanche impacts a symmetric splitting wedge at velocity $\mathbf{V_\Phi}$ and angle $+\theta$ relative to the wedge orientation. The goal is to determine the normal forces (F_Y^L, F_Y^R) acting on the wedge surfaces. The angles β_1^L and β_1^R define the calculation triangles.

The basic idea is to construct two wedges that are hit directly by the avalanche. Then the same equations derived in the previous section for deflection forces can be easily applied. The wedge is divided into two triangular sections: the left triangle ($\boldsymbol{LAM}$) and the right triangle ($\boldsymbol{RAM}$). The wedge's geometry is characterized by its internal angles, with α_1^L and

α_1^R representing the angles on the left and right sides, respectively. The apex angle of the wedge is the sum of α_2^L and α_2^R.

In the general case of a non-symmetric wedge, the bases of the two triangular sections (w_o^L and w_o^R) are not necessarily equal. The relationships between the wedge angles are given by:

- $\alpha_2^{\mathrm{L}} = \frac{\pi}{2} - \alpha_1^{\mathrm{L}}$
- $\alpha_2^{\mathrm{R}} = \frac{\pi}{2} - \alpha_1^{\mathrm{R}}$

The analysis begins by establishing a reference line, denoted as $\boldsymbol{AA'}$, which originates from the apex of the wedge ($\boldsymbol{A}$) and extends parallel to the avalanche impact angle θ (Figure 6.39). This reference line serves as the foundation for defining the interaction between the avalanche and the wedge.

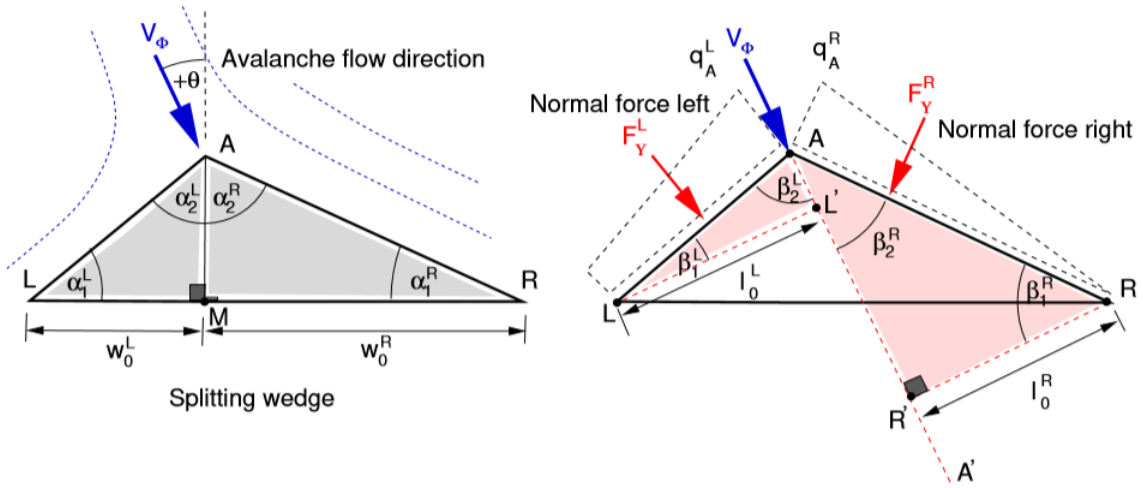


Figure 6.39: An avalanche impacts a non-symmetric splitting wedge at velocity V_Φ and angle θ. The wedge angles ($\alpha_1^L, \alpha_1^R, \alpha_2^L, \alpha_2^R$) define the original geometry, while the left and right segment lengths (w_o^L and w_o^R). To determine the normal forces on the wedge, the line AA'' is drawn parallel to the avalanche flow, while the lines *LL'* and *RR'* are constructed perpendicular to it. These define the calculation triangles (*LAL'* and *RAR'*) and their corresponding deflection angles ($\beta_1^L, \beta_2^L, \beta_1^R, \beta_2^R$) used to compute the normal forces F_Y^L and F_Y^R.

Next, two perpendicular lines, $\boldsymbol{LL'}$ and $\boldsymbol{RR'}$, are drawn from the wedge edges ($\boldsymbol{L}$ and $\boldsymbol{R}$), intersecting the reference line $\boldsymbol{AA'}$ at right angles. These lines help define the key geometric relationships essential for calculating the forces acting on the wedge.

By constructing these perpendiculars, two new calculation triangles emerge: $\boldsymbol{LAL'}$ on the left side and $\boldsymbol{RAR'}$on the right side. The angles associated with these triangles—β_1^L and β_2^L on the left, and β_1^R and β_2^R on the right—play a crucial role in determining the deflection forces exerted by the avalanche.

The relationships governing these angles are as follows:

- $\beta_1^{\mathrm{L}} = \frac{\pi}{2} - \beta_2^{\mathrm{L}}$ and $\beta_1^{\mathrm{R}} = \frac{\pi}{2} - \beta_2^{\mathrm{R}}$
- $\beta_2^{\mathrm{L}} = \alpha_2^{\mathrm{L}} + \theta$ and $\beta_2^{\mathrm{R}} = \alpha_2^{\mathrm{R}} - \theta$

These angular transformations allow us to assess how the incoming avalanche flow is redirected upon impact, influencing both the magnitude and distribution of forces across the wedge surface. The lengths of the calculation wedge can be found

$$l_o^L = \frac{\sin \beta_2^L}{\cos \alpha_1^L} w_o^L \quad \text{and} \quad l_o^R = \frac{\sin \beta_2^R}{\cos \alpha_1^R} w_o^R.$$

The **normal forces exerted by the avalanche** on each side of the wedge are derived using the **deflection force equation** derived above:

$$F_{\mathrm{Y}}^L = \frac{[3\rho_{\mathrm{Y}}(\cos \beta_1^L)^2 + 4\rho_{\Phi} - 2\rho_{\mathrm{Y}}]}{[2\rho_{\Phi} + \rho_{\mathrm{Y}}]} \rho_{\Phi} \mathrm{V}_{\Phi}^2 \, l_o^L w_0 \cos \beta_1^L$$

$$F_{\mathrm{Y}}^R = \frac{[3\rho_{\mathrm{Y}}(\cos \beta_1^R)^2 + 4\rho_{\Phi} - 2\rho_{\mathrm{Y}}]}{[2\rho_{\Phi} + \rho_{\mathrm{Y}}]} \rho_{\Phi} \mathrm{V}_{\Phi}^2 \, l_o^R w_0 \cos \beta_1^R$$

The **corresponding mean pressures** on the wedge surfaces are:

$$\bar{p}_{\mathrm{Y}}^L = \frac{F_{\mathrm{Y}}^L \cos \beta_1^L}{l_o^L} = \frac{[3\rho_{\mathrm{Y}}(\cos \beta_1^L)^2 + 4\rho_{\Phi} - 2\rho_{\mathrm{Y}}]}{[2\rho_{\Phi} + \rho_{\mathrm{Y}}]} \rho_{\Phi} \mathrm{V}_{\Phi}^2 (\cos \beta_1^L)^2$$

$$\bar{p}_{\mathrm{Y}}^R = \frac{F_{\mathrm{Y}}^R \cos \beta_1^R}{l_o^R} = \frac{[3\rho_{\mathrm{Y}}(\cos \beta_1^R)^2 + 4\rho_{\Phi} - 2\rho_{\mathrm{Y}}]}{[2\rho_{\Phi} + \rho_{\mathrm{Y}}]} \rho_{\Phi} \mathrm{V}_{\Phi}^2 (\cos \beta_1^R)^2$$

Figure 6.40 illustrates how normal forces and pressures change with the impact angle θ. As θ increases, the force F_{Y}^L rises on the impact side, while F_{Y}^R decreases on the shadow side. When F_{Y}^L reaches 30°, the avalanche flow is parallel to the right side, making F_{Y}^R=0.

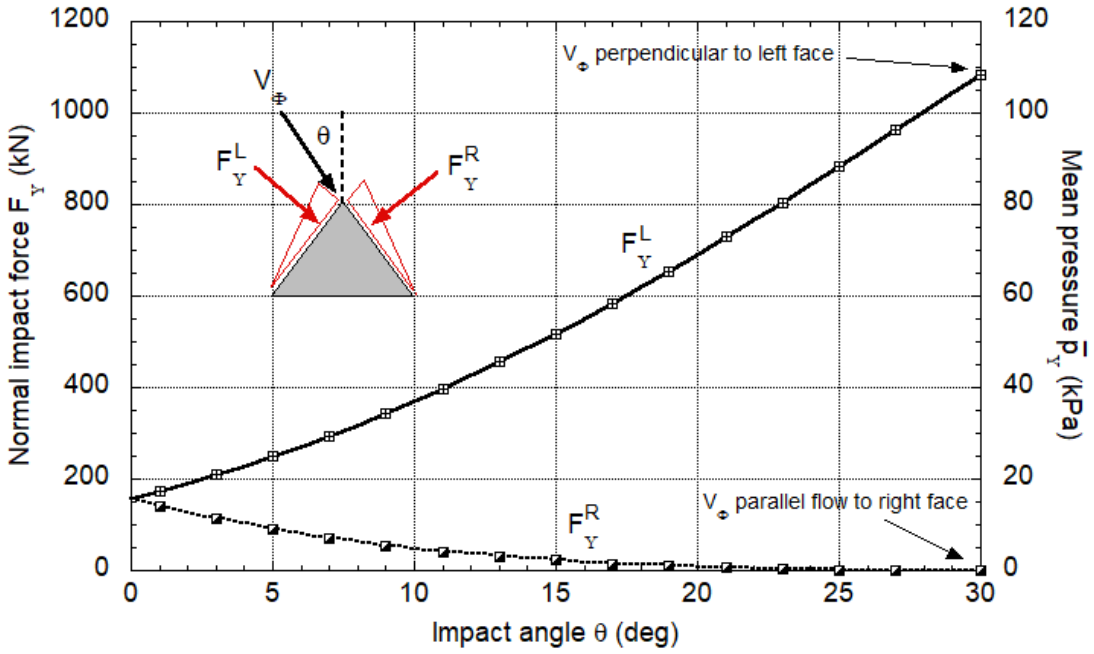


Figure 6.40: Deflection forces on a splitting wedge hit at different angles θ. The avalanche impacts the wedge with a velocity of V_Φ=20m/s and density ρ_Φ=300kg/m³. Compaction density ρ_Y=500kg/m³. The half-base width is w_0 = 5m. Normal forces F_Y^L increase on the impact side and decrease on the shadow side F_Y^R. At an impact angle of θ = 30° the flow is parallel to the right side F_Y^R=0.

6.12Oblique Impact of Buildings

We now examine the loading scenario illustrated in Figure 6.41, where an avalanche **collides with a building at an oblique angle**. The structure in question has a rectangular footprint with unequal side lengths. Upon impact, the avalanche flow is divided at the initial point of contact, commonly referred to as the **impact corner**. If the building can withstand the force of the impact, the avalanche is split into **two separate flow paths** that continue along the length of each adjacent side. **Consequently, deflecting forces act on both walls.**

The magnitude of these deflecting forces depends predominantly on the velocity and density of the avalanche, but additional factors—such as the length of the impacted sides and the **strike angle** θ at which the flow meets the structure—also play a crucial role. When the avalanche is sufficiently wide, the deflected snow exerts shearing forces at the exposed building corners, which can further influence the structural load.

In essence, the interaction between the avalanche and a rectangular building closely resembles that of a splitting wedge, where the incoming flow is forcibly redirected, **creating distinct dynamic forces** that must be accounted for in structural design and hazard mitigation planning.

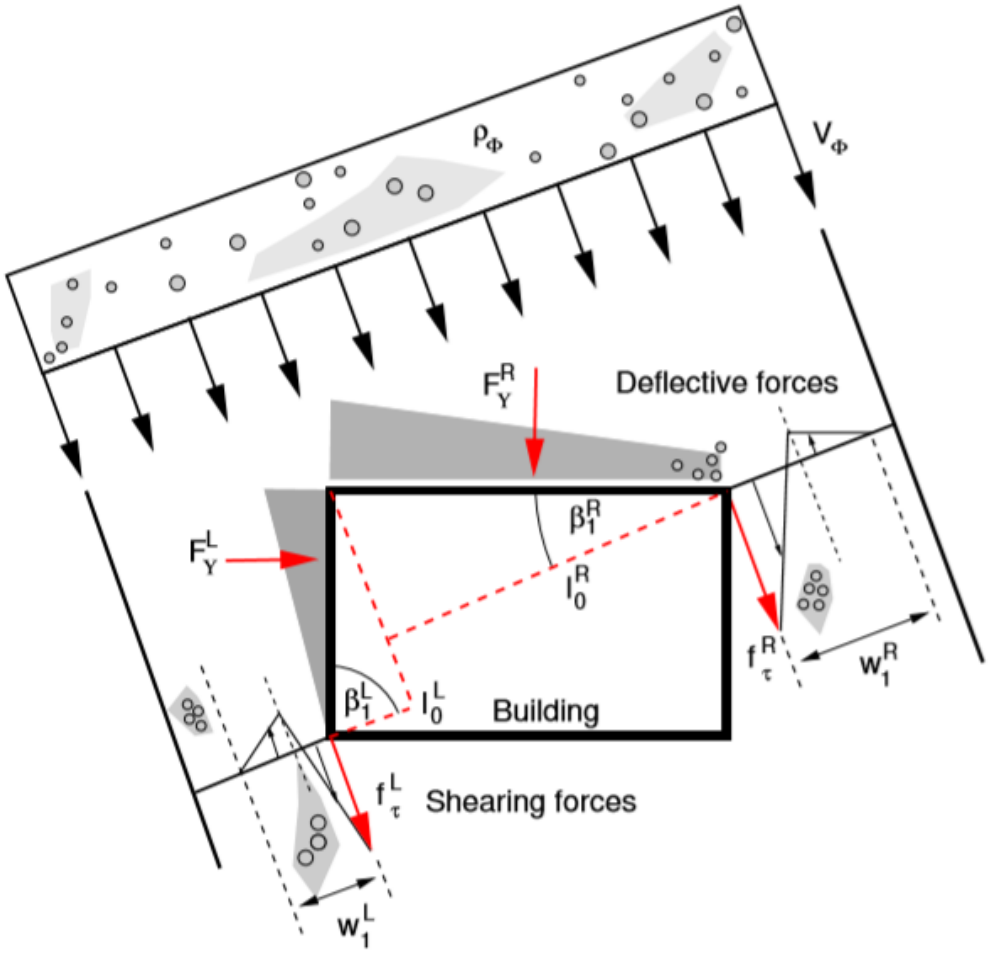


Figure 6.41: Oblique avalanche impact of a rectangular building. The avalanche approaches the building at speed V_Φ at the angle $\pm\theta$ relative to the upper left corner. The problem is to find the normal forces F_Y acting on the building sides.

Let the **impact corner** be denoted **A** (Figure 6.42). The avalanche flow is thus deflected along the walls $\boldsymbol{AR}$ (right side, length S_R) and $\boldsymbol{AL}$ (left side, length S_L). Buildings are a specific type of splitting wedge with $\boldsymbol{AR} \perp \boldsymbol{AL}$, which is to say that the building corners are right angles. The calculation angles for the deflection forces are always,

- $\alpha_1^L = \alpha_2^L = \frac{\pi}{4}$
- $\alpha_1^R = \alpha_2^R = \frac{\pi}{4}$

for both the left and right sides of the building (Figure 6.42). The reference configuration (strike angle $\theta = 0°$ is thus when the avalanche hits the building at angle of $\pi/4$. This creates two initial triangles, $\Delta\boldsymbol{ARM}^R$ and $\Delta\boldsymbol{AM}^{\boldsymbol{L}}\boldsymbol{L}$. The base length of each triangle is simply,

$$w_o^L = \frac{S_L}{\sqrt{2}}$$

$$w_o^R = \frac{S_R}{\sqrt{2}}$$

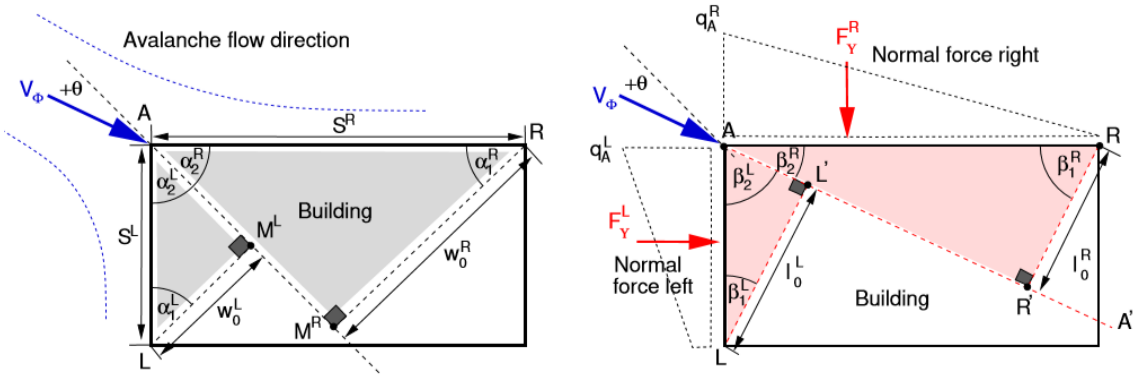


Figure 6.42: An avalanche impacts a rectangular building with the speed V_Φ at the angle $\pm\theta$ relative to the wedge orientation. The problem is to find the normal forces F_Υ acting on the building sides. The problem is treated as a non-symmetrical wedge. The angles β_1^L and β_1^R become the calculation angles.

With this information the deflection forces F_Υ and mean normal pressures p_Υ for the case $\theta = 0°$ could be found directly by applying deflection formulas derived in the previous section for splitting wedges. This leads to the force and pressure formulas,

$$F_\Upsilon^L = \frac{S_L}{2}\frac{[8\rho_\Phi - \rho_\Upsilon]}{[2\rho_\Phi + \rho_\Upsilon]}\left[\frac{\rho_\Phi V_\Phi^2}{2}\right]$$

$$F_\Upsilon^R = \frac{S_R}{2}\frac{[8\rho_\Phi - \rho_\Upsilon]}{[2\rho_\Phi + \rho_\Upsilon]}\left[\frac{\rho_\Phi V_\Phi^2}{2}\right]$$

$$p_\Upsilon^L = p_\Upsilon^R = \frac{1}{4}\frac{[8\rho_\Phi - \rho_\Upsilon]}{[2\rho_\Phi + \rho_\Upsilon]}[\rho_\Phi V_\Phi^2]$$

We note that the pressures on the left and right side of the building are equal (Figure 6.43a) whereas the forces depend on the side length (Figure 6.43b). Similar to the deflection loading of a circular cross section, the as C_d coefficient depends on the assumption of the compactibility density of the avalanche snow ρ_Υ in relation to the avalanche flow density ρ_Φ. The pressure factors will typically range between $0.86 \leq C_d \leq 1.17$. Engineers should decide whether the application means pressures or distribute the normal force less homogeneously on each side wall.

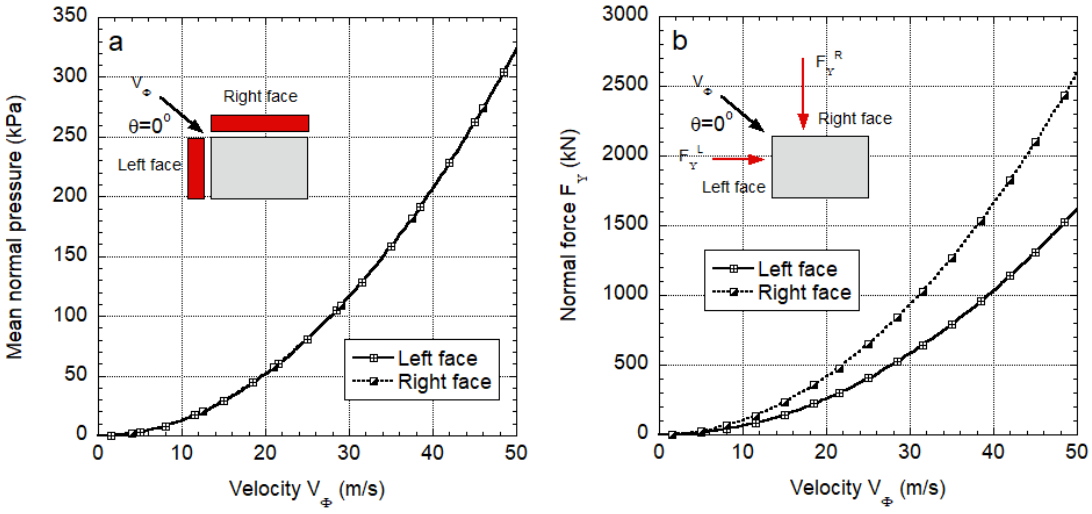


Figure 6.43: An avalanche impacts a 5m (left) x 8m (right) building. Deflection force F_Y as a function of impact velocity V_Φ. The avalanche hits the upper left corner at an angle of 45° ($\theta = 0°$). The mean deflection pressures are the same on both sides (a), but the mean forces are higher on the right side (b), the larger of the two side walls.

When the avalanche strikes the building at an $\theta \neq 0°$, the calculation procedure is slightly modified. Calculation triangles are introduced, like the analysis of splitting wedges. These triangles are, as before triangles $\Delta \boldsymbol{ARR}'$ and $\Delta \boldsymbol{AL'L}$. This leads to the calculation angles,

- $\beta_1^L = \frac{\pi}{4} - \theta$ and $\beta_1^R = \frac{\pi}{4} + \theta$
- $\beta_2^L = \frac{\pi}{4} + \theta$ and $\beta_2^R = \frac{\pi}{4} - \theta$

The lengths of the calculation wedges can be found as before

$$l_o^L = S_L \sin \beta_2^L \quad \text{and} \quad l_o^R = S_R \sin \beta_2^R$$

This leads to the force formulas for the left and right sides,

$$F_Y^L = \frac{[3\rho_Y(\cos\beta_1^L)^2 + 4\rho_\Phi - 2\rho_Y]}{[2\rho_\Phi + \rho_Y]} \rho_\Phi V_\Phi^2 \, l_o^L \cos\beta_1^L$$

$$F_Y^R = \frac{[3\rho_Y(\cos\beta_1^R)^2 + 4\rho_\Phi - 2\rho_Y]}{[2\rho_\Phi + \rho_Y]} \rho_\Phi V_\Phi^2 \, l_o^R \cos\beta_1^R$$

$$\bar{p}_Y^L = \frac{F_Y^L \cos\beta_1^L}{l_o^L} = \frac{[3\rho_Y(\cos\beta_1^L)^2 + 4\rho_\Phi - 2\rho_Y]}{[2\rho_\Phi + \rho_Y]} \rho_\Phi V_\Phi^2 (\cos\beta_1^L)^2$$

$$\bar{p}_Y^R = \frac{F_Y^R \cos\beta_1^R}{l_o^R} = \frac{[3\rho_Y(\cos\beta_1^R)^2 + 4\rho_\Phi - 2\rho_Y]}{[2\rho_\Phi + \rho_Y]}\rho_\Phi V_\Phi^2(\cos\beta_1^R)^2$$

With a non-zero strike angle the pressures on the left and right-side walls are no longer equal. This is demonstrated in Figure 6.44a where an avalanche impacts a building strike angle $\theta = -10°$ from the reference axis. This increases the pressures on the right wall and decreases the pressures on the left wall (Figure 6.44a). The calculated impact forces are provided in Figure 6.44b. The engineer can distribute the forces on the building side according to the specific problem.

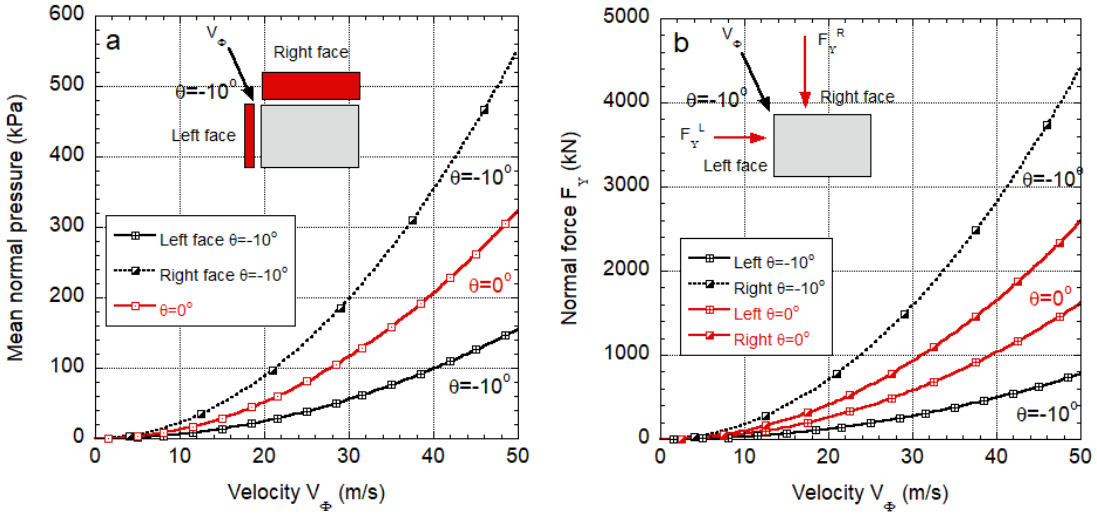


Figure 6.44: An avalanche impacts a 5m (left) x 8m (right) building at an angle $\theta = -10°$. The mean wall pressures are no longer equal (a). The oblique impact increases the force on the right wall but decreases the force on the left wall (b).

6.13 Shearing Forces

Shearing forces f_τ refer to long-duration, non-impulsive stresses that develop when **dense, slow-moving snow flows interact with obstacles or boundaries**. Unlike impact or pressure forces that arise from rapid momentum transfer, shearing forces emerge gradually as the flowing material slides, grinds, or deforms while passing along the surface of a rigid object. These forces are especially prominent at the **edges of obstacles** or at **channel walls**, where velocity gradients are strongest and flow is deflected laterally.

Shearing forces are characteristic of **dense avalanche regimes**, where particle interactions and frictional contacts dominate flow behavior. In these regimes, the flow is often slower and more confined, allowing for sustained contact with terrain features. The development of shearing stress in these zones is primarily due to **differential motion within the flow**—that is, layers or regions of snow moving at different velocities relative to each other or relative to an immobile surface.

Because these forces act tangentially, **they serve as a balancing mechanism against gravity-driven downslope motion**. On inclined terrain, shear stress resists gravitational

acceleration and regulates flow thickness, runout, and deposition. Therefore, **shearing forces are strongly slope-dependent**: steeper slopes result in higher driving forces and, consequently, higher shear stress at the base and boundaries of the flow.

Importantly, however, **shearing forces can also develop on flat terrain**, where the gravitational driving force approaches zero. In such cases, shear arises solely from **internal momentum within the flow**—momentum that continues to carry the mass forward even in the absence of a downslope component. At flow margins and near boundaries, this residual movement produces shear, which then acts to **decelerate the material**. Thus, even in low-slope or depositional zones, shear stress plays a critical role in dissipating energy and determining the final distribution of deposits.

The present version of **RAMMS::Extended** does not include a calculation method for shear stresses on obstacle edges. The primary reason is the lack of experimental data to validate a calculation procedure. An edge-shear pressure formula will appear in future versions of **RAMMS::Extended**.

6.14 Pressures on Long Deflection Walls (Old Theory)

In avalanche protection design, long walls (wider than the avalanche flow) experience deflective pressures calculated using:

$$p_n = \rho_\Phi V_\Phi^2 (\sin\alpha)^2 p_n = \rho_\Phi V_\Phi^2 (\cos\alpha_1)^2$$

where $\boldsymbol{\alpha}$ is the deflection angle, see Figure 6.45. Note that in **RAMMS::Extended** deflection theory the deflection is given by the wedge opening angle $\boldsymbol{\alpha_1 = \alpha - 90°}$. This formula is recommended by Swiss and other guidelines. The associated pressure coefficient is:

$$C_d = 2(\sin\alpha)^2$$

with $\boldsymbol{C_d = 2}$ at $\boldsymbol{\alpha = 90°}$ ($\boldsymbol{\alpha_1 = 0°}$) (direct impact). For high-velocity, fluidized avalanches, this value is generally sufficient. The deflection formula—which accounts for snow compression but no pile-up—predicts higher pressures at intermediate angles ($\boldsymbol{90° \leq \alpha \leq 60°}$). At $\boldsymbol{\alpha = 45°}$, methods converge to $\boldsymbol{C_d = 1}$. Shear pressure is given by:

$$p_s = \mu p_n$$

with μ =0.3 for **snow/snow or snow/concrete** and μ =0.4 for **snow on rough walls**.

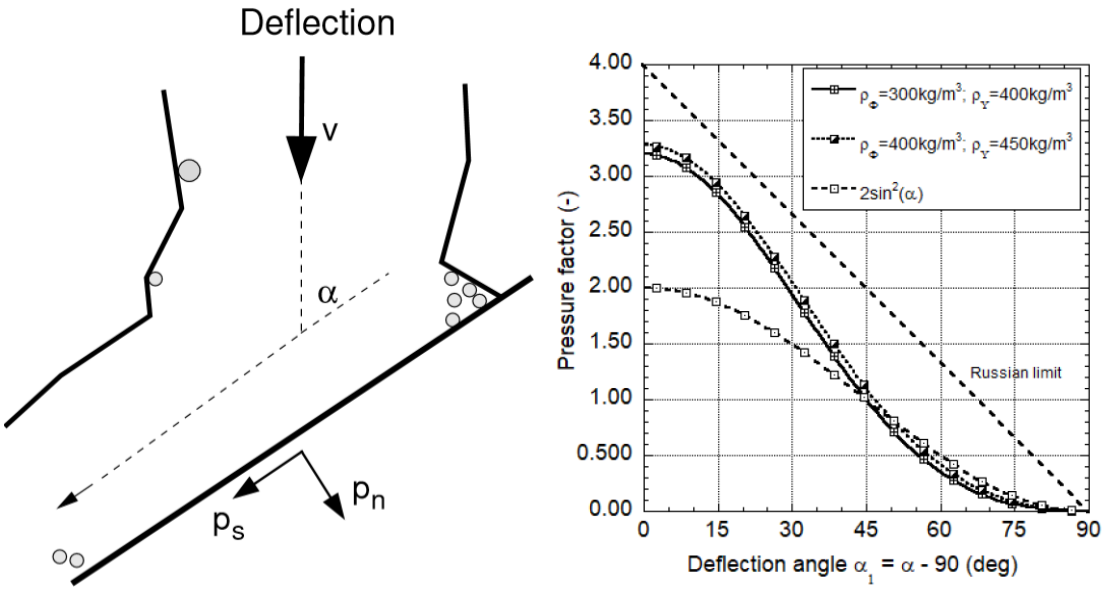


Figure 6.45: Normal p_n and shear p_s pressures on a long wall. The guidelines recommend using $C_d = 2(\sin\alpha)^2$ to calculate the normal pressure. For direct impact $\alpha = 90°$, $C_d = 2$. Larger C_d pressure factors are obtained with the deflection formula (plot right) due to the compactibility of avalanche snow. Russian experiments in the 1970s and 80s placed an upper limit on the values.

6.15 Short Duration Particle Impacts p_π

When a **snow particle impacts** an obstacle, it generates a localized impact over a very short time interval, leading to significant pointwise force. Understanding these impact forces is often crucial in analyzing the behavior of snow avalanches, especially the destructive force of **saltation/splashing fronts**. We use the lowercase Greek letter π to represent "particle" and to indicate that particle impacts are often associated with mixed flowing/powder avalanches Π with splashing fronts Γ.

The pressure exerted by an individual particle impact can be derived using fundamental principles of mechanics, particularly the work-energy theorem and momentum considerations. Structural engineers are interested in both the point loading force F_π and the associated pressure p_π,

The pressure due to a **single particle impact** (see Figure 6.46) can be obtained from the work-energy theorem:

$$p_\pi = \frac{F_\pi}{A_\pi} = \frac{1}{2}\frac{M_\pi {V_\Pi}^2}{A_\pi s_\pi} = \frac{4}{3}\rho_\pi {V_\Pi}^2$$

where:

- $A_\pi = \pi r_\pi^2$ is the particle's **cross-sectional area**,
- $M_\pi = \frac{4}{3}\rho_\pi \pi r_\pi^3$ is the **mass** of the snow clod,
- $s_\pi = r_\pi/2$ is the **braking/stopping distance** of the particle.

Here, it is assumed that the particle moves at the velocity of the surrounding powder cloud, $V_\Pi = \|\vec{u}_\Pi\|$, implying that aerodynamic drag is negligible at the moment of impact. Experimental results indicate that the local impact pressure can be very high, reaching values up to p_π =2000kPa—the highest pressures ever recorded in impact experiments.

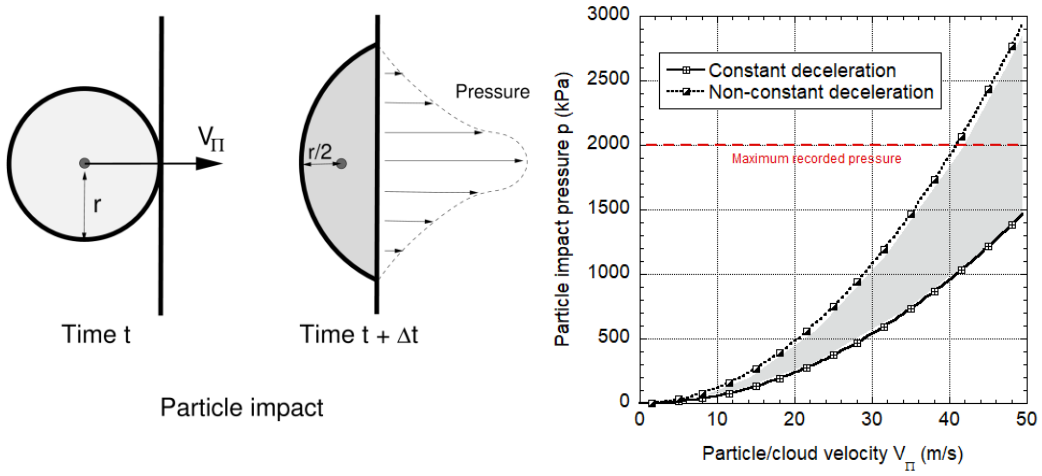


Figure 6.46: A particle with radius r_π and density ρ_π =450 kg/m³ impacts a rigid wall at velocity V_Π. The particle comes to rest after traveling the braking distance s_π. (Right) The resulting pressure p_π as a function of impact velocity V_Π.

An alternative derivation of the impact pressure can be obtained directly from the momentum change principle:

$$p_\pi = \frac{F_\pi}{A_\pi} = \frac{M_\pi V_\Pi}{A_\pi \Delta t} = \frac{4}{3}\rho_\pi {V_\Pi}^2$$

where the impact duration is estimated as $\Delta t = r_\pi/V_\Pi$. This approach leads to the same pressure formula, reinforcing the assumption of a constant deceleration over the braking distance s_π.

Assuming constant deceleration a_π over the braking distance s_π, the kinematic equations yield:

$$a_\pi = -\frac{V_\Pi}{\Delta t} \qquad v(t) = V_\Pi + a_\pi t \qquad s(t) = V_\Pi t + \frac{1}{2}a_\pi t^2$$

The corresponding velocity and braking distance curves are depicted in Figure 6.47.

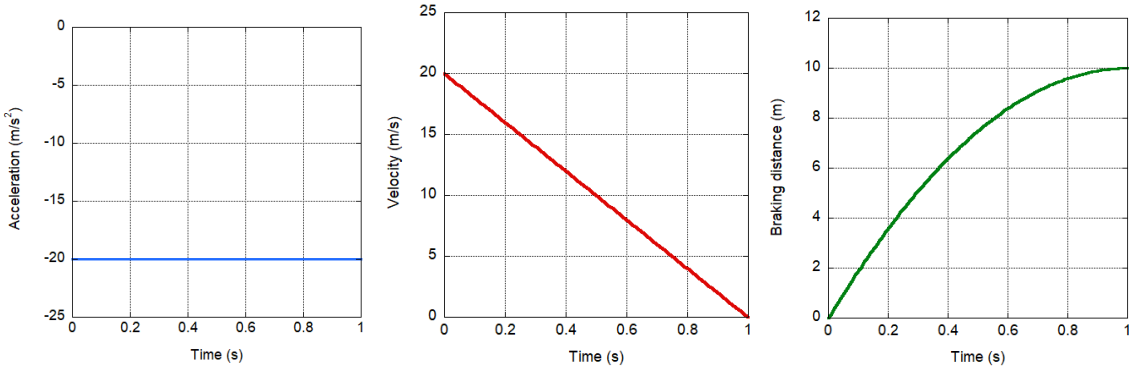


Figure 6.47: (a) Constant deceleration, (b) linear velocity decrease, and (c) quadratic increase in braking distance. The peak force corresponds to a deceleration of 20g. The incoming velocity is V_Π =20m/s.

In reality, deceleration during impact is not necessarily constant. Depending on material properties and impacting surface, deceleration may:

- Peak immediately upon impact,
- Gradually increase and reach a maximum at the midpoint of the collision,
- Start low and increase continuously as the particle compacts against the obstacle.

To model these scenarios, the concept of **jerk** j_π (the rate of change of acceleration) is introduced:

$$j_\pi = \pm \frac{a_{max}}{\Delta t}$$

Assuming a constant jerk, the velocity and displacement relations become:

$$v(t) = V_\Pi + a_\pi t + \frac{1}{2} j_\pi t^2 \qquad s(t) = V_\Pi t + \frac{1}{2} a_\pi t^2 + \frac{1}{6} j_\pi t^3$$

The implications of these variations are shown in Figure 6.48.

The variation in deceleration profiles suggests that impact forces are highly dependent on how the braking process unfolds (Figure 6.49). Specifically:

- **Peak forces are significantly larger when deceleration is non-uniform.**
- **The measured impact force cannot be directly correlated with avalanche velocity alone.**
- **The assumption of constant deceleration is often questionable, particularly in natural materials such as snow, ice, or rock.**

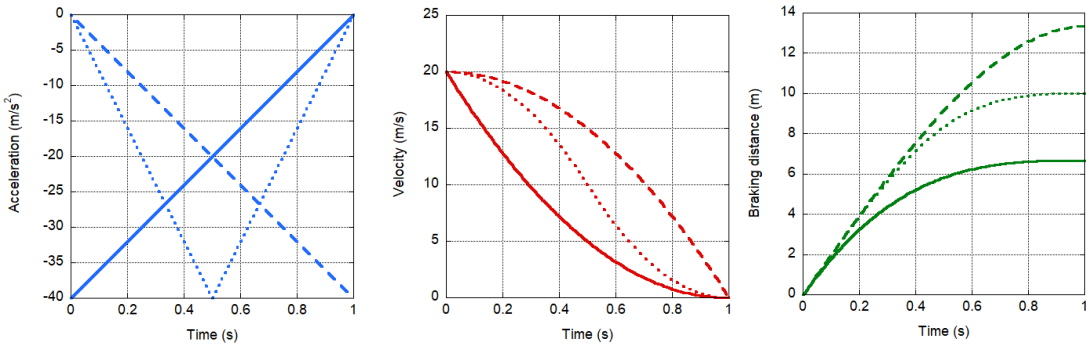


Figure 6.48: Impact force behavior under different deceleration profiles: (a) varying deceleration, (b) nonlinear velocity changes, and (c) adjusted braking distances. Peak impact forces can reach 40g, doubling those for constant deceleration. The incoming velocity is V_Π =20m/s.

These findings have profound implications for the design of building components (such as reinforced windows) against avalanches, emphasizing the need for detailed impact force characterization beyond simplified constant acceleration models. It also explains in part the difficulty of interpreting pressure cell measurements.

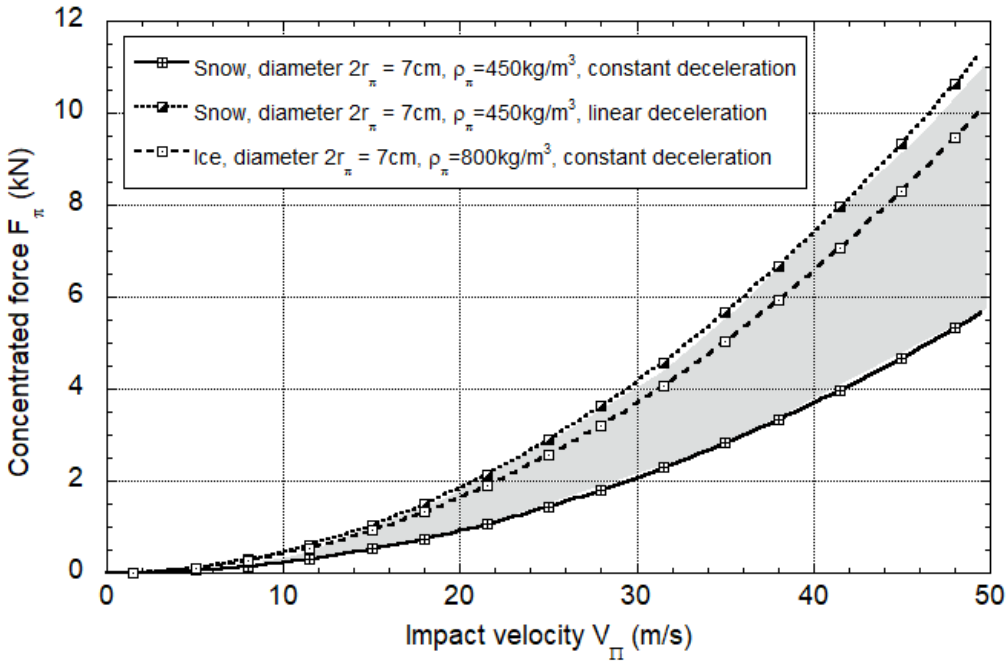


Figure 6.49: Particle impact force F_π as a function of the avalanche velocity V_Π. The calculated force depends on the particle density (ice, snow) and the assumption of how it decelerates at impact.

6.16 Woody Debris

During descent, avalanches often **destroy forest stands, entraining woody, rocky, and earthen debris** (Figure 6.50). Trees may be snapped or uprooted, and once inside the flow, branches and stems are further broken down by internal shear forces. By the time the

avalanche reaches the transition or runout zone, **it may contain a substantial volume of woody debris**—estimated in some cases to exceed 30% of the flow volume (Figure 6.50).

Woody debris fundamentally changes how avalanches interact with obstacles, particularly by increasing the potential for **jamming** and **elevated shearing forces**. Although the densities of common mountain tree species (Table 6.4) are comparable to compacted snow, wood behaves very differently upon impact. Unlike snow, it **cannot compact significantly**. This lack of compactibility **shortens braking distances during stagnation and deflection, resulting in higher impact forces**.

The effect is most pronounced in **shearing zones**, where long, rigid tree stems can become **lodged or jammed against obstacles**. This jamming concentrates force along narrow contact regions, amplifying shearing stresses beyond those typical in snow-only flows. As a result, even moderate amounts of woody debris can significantly raise shearing forces, particularly in slow to moderate flows where braking effects dominate.

Wood Type	Density (kg/m^3)
Spruce	480-780
Norway Spruce	430
Scotts Pine	510
Larch	500-560
European Birch	670
Fir	530

Table 6.4: Typical densities of tree species found in mountainous terrain. Except for birch, most values are close to the co-volume density of compacted snow. Accordingly, in RAMMS::Extended, the flow density is kept constant, while compaction densities are reduced to account for the limited compressibility of woody debris when calculating impact pressures.

Crucially, this behavior is not driven by an increase in the flow density itself, but by a reduction in **the compaction density**—the density the material reaches when piling up against an obstacle. Unlike snow, wood cannot compact significantly. Consequently, the closer the flow density is to the (lowered) compaction density, the shorter the braking distance and the higher the resulting impact force.

In **RAMMS::Extended**, this effect is captured by reducing the compaction densities — ρ_Ξ for stagnation and shearing pressures, and ρ_Υ for deflection pressures — based on the presence of woody debris. The flow density remains unchanged. The default categories are:

- **No woody debris:** ρ_Ξ, ρ_Υ = 600 kg/m³
- **Moderate woody debris:** ρ_Ξ, ρ_Υ = 550 kg/m³
- **Extreme woody debris:** ρ_Ξ, ρ_Υ = 500 kg/m³

The effect of shearing and jamming is considered by increasing the ratio of flow height h_Φ to pile-up height h_Ξ:

- **No woody debris:** $h_\Phi / h_\Xi = 1/3$
- **Moderate woody debris:** $h_\Phi / h_\Xi = 1/2$
- **Extreme woody debris:** $h_\Phi / h_\Xi = 1/1$

Figure 6.50: Woody debris in the Val Preada avalanche, Engadine. Woody debris increases the impact pressures on obstacles by reducing the compactibility of avalanche snow.

These lower compaction densities reduce braking distances and lead to increased impact pressures, especially in the runout zone, where debris is fully entrained and contributes primarily through increased mass and reduced **compactibility** (Figure 6.51).

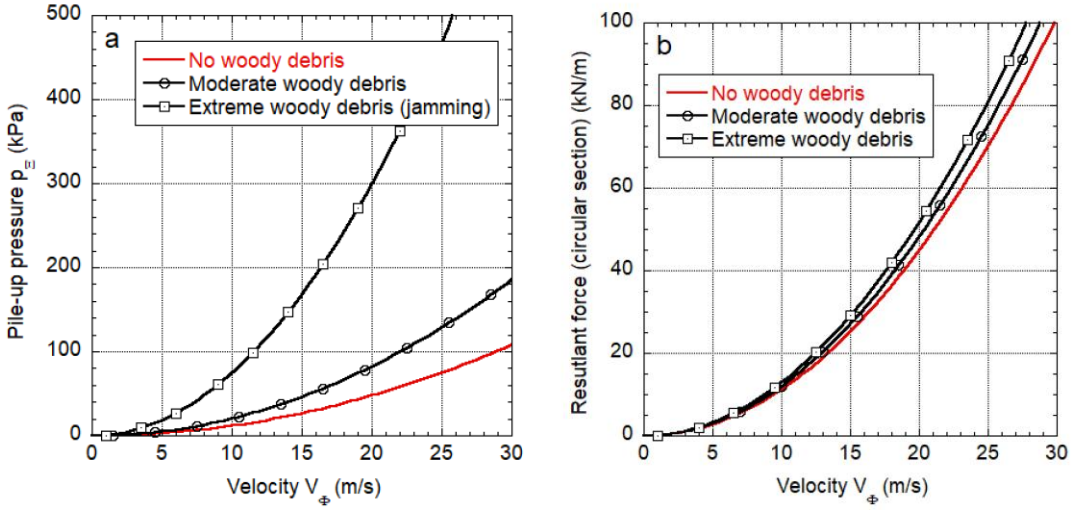


Figure 6.51: a) Influence of woody debris on stagnation/pile-up pressures p_{Ξ} in RAMMS::Extended. Stagnation pressures increase from the cases no woody debris to extreme woody debris. b) Influence of woody debris on deflection pressures p_{Υ} areound circular pylon, case $r = 0.5$m

Chapter 7 Forests

7.1 Introduction: Forests in Avalanche Dynamics

Traditionally, forests have not been incorporated into avalanche dynamics calculations for two primary reasons**. First, large and high-energy avalanches — especially those originating above the treeline — can easily devastate entire forest stands**. In such events, the momentum and destructive power of the avalanche far exceed the mechanical resistance offered by the trees. Second, **forests are dynamic and vulnerable ecosystems**, often subject to natural disturbances such as windthrow, bark beetle infestations, and disease. Given these unpredictable and long-term changes**, forests are generally excluded from hazard zoning and risk assessments for rare, high-return-period avalanches** (e.g., 100- or 300-year events).

However, in recent years there has been a growing emphasis on understanding the behavior of **small to medium-sized avalanches—events where protective forests are not obliterated** but instead act as effective natural barriers. In such cases, **forests play a significant role in reducing avalanche impact pressures and limiting runout distances**. As a result, forest effects are now increasingly considered in simulations for more frequent, lower-magnitude avalanches, typically associated with 10- or 30-year return periods.

Beyond their protective function, forests have gained renewed attention as **natural indicators of avalanche forces**. Trees, due to their structural simplicity, behave as cantilever beams with well-characterized breaking thresholds. This makes them **ideal passive sensors for back-calculating avalanche impact pressures** and flow velocities based on observed tree damage. It is even possible to quantify dynamic magnification factors, indicating the activation of inertial forces during the avalanche-tree impact. In **RAMMS::Extended**, one of the primary motivations for integrating forest interactions is precisely this: **to reconstruct the dynamics of powder cloud formation and destruction patterns**, thereby enabling robust model calibration and validation.

For this reason, substantial effort has been devoted to implementing a comprehensive forest model within **RAMMS::Extended**. This includes not only the mechanical influence of trees on avalanche flow, but also a dedicated forest module capable of visualizing the extent of tree destruction. These additions provide a practical framework for incorporating forest interactions into avalanche modeling, enabling more realistic simulations and supporting improved interpretation of forest-related avalanche effects.

7.2 Forest Interaction in RAMMS::Extended

RAMMS::Avalanche uses a **drag-based approach** to model the effect of forests, while **RAMMS::Extended**, introduces a **detrainment-based model** — a fundamental shift in how

forests are treated in the different RAMMS modules. In the **drag-based approach**, forest effects are represented by increasing the **Voellmy friction parameters**, thereby simulating the deceleration of avalanche flow within vegetated terrain.

In contrast, the **detrainment model** characterizes forest interaction through a **mass detrainment rate**, denoted as $\dot{M}_{\Phi\to\Psi}$. This term quantifies the **mass removed** from the avalanche core (Φ) due to obstruction by trees. The corresponding **momentum loss** is given by $\left[\dot{M}_{\Phi\to\Psi}\right]\vec{u}_{\Phi}$, where $\vec{u}_{\Phi}$ is the velocity of the core. The symbol Ψ is used as a subscript to represent the forest, chosen for its visual resemblance to a tree.

$$\left(\hat{h}_{\Phi}\right)_t + div\left(\hat{h}_{\Phi}\vec{u}_{\Phi}\right) = \frac{\rho_{\Sigma}}{\hat{\rho}_{\Phi}}\dot{M}_{\Sigma\to\Phi} - \dot{M}_{\Phi\to\Psi} - \frac{\hat{\rho}_{\Pi}}{\hat{\rho}_{\Phi}}\,\dot{M}_{\Phi\to\Pi}$$

$$\left(\hat{h}_{\Phi}\vec{u}_{\Phi}\right)_t + div\left(\hat{h}_{\Phi}\vec{u}_{\Phi}\otimes\vec{u}_{\Phi} + p_{\Phi}I\right) = \vec{G}\hat{h}_{\Phi} - \frac{\vec{u}_{\Phi}}{\|\vec{u}_{\Phi}\|}S_{\Phi}(R_{\Phi}, T_{\Phi}, m_{\Phi})$$

$$-\left[(1+r_{\Gamma})\frac{\rho_{\Sigma}}{\hat{\rho}_{\Phi}}\dot{M}_{\Sigma\to\Gamma} + \dot{M}_{\Phi\to\Psi} + \frac{\hat{\rho}_{\Pi}}{\hat{\rho}_{\Phi}}\dot{M}_{\Phi\to\Pi}\right]\vec{u}_{\Phi}$$

The **detrainment concept** is based on physical observations that trees **trap snow** and reduce avalanche intensity **not by slowing it down gradually**, but by **halting part of the avalanche mass immediately**. Unlike in **RAMMS::Avalanche**, this is **not parameterized with friction corrections** but is instead a **calculated sink**. By removing mass directly, the random kinetic and internal energies are likewise removed from the avalanche.
A comparison between the two models is provided in Table 7.1**.**

The mass detrainment rate $\dot{M}_{\Phi\to\Psi}$(m/s) is governed by a single parameter K_{Ψ} (Pa)

$$\hat{\rho}_{\Phi}\dot{M}_{\Phi\to\Psi} = \frac{K_{\Psi}}{V_{\Phi}}$$

The volume of detrained snow is inversely proportional to the avalanche flow velocity V_{Φ}; that is, **higher flow velocities result in lower detrainment rates**. Conversely, when the avalanche **moves more slowly, it more readily deposits snow within forested areas**, making tree stands more effective at capturing and retaining avalanche snow.

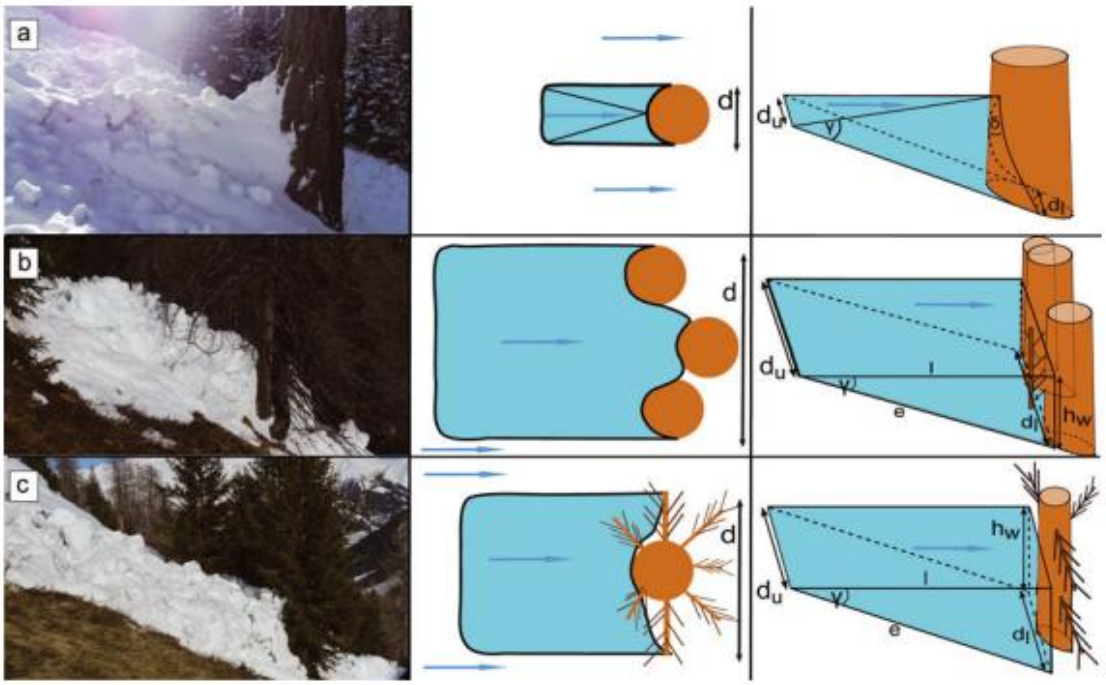


Figure 7.1: Figure taken from Feistl et al. (2014). Photographs showing avalanche snow deposition behind trees. The central column depicts snow accumulations within the forested area. The rightmost column provides an approximation of the volume of avalanche snow that was deposited behind the trees.

In **RAMMS::Extended** there are **18 categories (A-R) of K_Ψ,** ranging from K_Ψ = 44Pa to K_Ψ = 5Pa.

Field observations of avalanche mass deposition in forested areas reveal that detrained snow tends to accumulate behind tree stands. **Notably, the detrainment process is highly heterogeneous, exhibiting significant spatial variability** (see Figure 7.1)**.** In the **RAMMS::Extended** model, we represent this process by modeling the **mean detrainment heights**, thereby avoiding the need to specify the exact locations of individual tree stands. Additionally, the forest detrainment coefficient K_Ψ captures the **average detrainment rate** rather than the **maximum detrainment height** occurring directly behind trees (Figure 7.2).

As the value of $\dot{M}_{\Phi\to\Psi}$ represents the mean detrainment rate, the total detrained mass, represented by $M_{\Phi\to\Psi}$ is the entire snow mass caught by the forest. The total mass stopped by the forest can be **output in RAMMS::Extended**. Measured total detrainment volumes were used to calibrate the K_Ψ values.

Drag (Friction) Approach RAMMS::Avalanche	Detrainment Model RAMMS::Extended
Uses **indirect parameters** (increased friction).	**Directly models** snow stopping and jamming effects.
Less accurate in simulating **observed deposition patterns.**	Better reproduces **deposition shapes** and **runout shortening**, especially for small avalanches.
Cannot capture **jamming or momentum sink** processes.	Momentum extraction better reflects **real forest effects** (e.g., snow piling behind trees).
Often leads to **overestimated runouts** in these conditions.	Works well on **steep slopes** where traditional friction fails to stop the avalanche.
No direct link to **physical forest characteristics.**	Allows **calibration via parameter K**, linking to measurable forest features.
Slows flow gradually but not always realistically.	**Effective stopping** of avalanche flow (matches observations).

Table 7.1: Comparison between friction approach and detrainment forest approach.

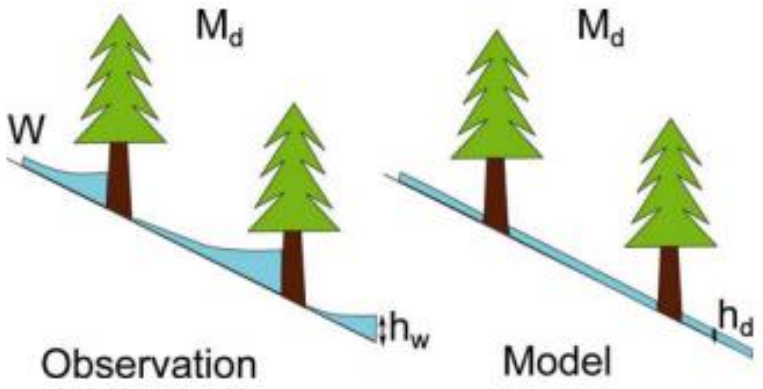


Figure 7.2: The forest detrainment model calculates the average detrainment height, not the maximum detrainment heights behind individual tree stands.

Table 7.2 and Figure 7.3 overview the detrainment values used in RAMMS::Extended. In summary, the **RAMMS::Extended** detrainment approach:

- Models **momentum extraction** due to snow jamming between trees.
- More **physically representative** than simply increasing friction.
- Calibrated through **field data** and deposition patterns.
- Shows strong agreement with **real-world avalanche deposition and runout lengths**.
- Only valid when **forests are not destroyed** by the avalanche (i.e., trees remain as obstacles). Therefore **RAMMS::Extended as options to reduce the value of K_Ψ if the tree stands are destroyed by the avalanche.**

The forest interaction module in **RAMMS::Extended** is tailored for simulating small to medium-sized avalanches, where forest stands remain largely intact and contribute to flow resistance. Field studies by Feistl (2014) show that mean detrainment height—the depth of snow retained in forests—rarely exceeds 0.5 meters. Although snow can temporarily pile up around individual trees, the spatial average stays below this threshold (Figure 7.2). To reflect these dynamics, the **RAMMS::Extended detrainment algorithm** halts once the mean detrainment height reaches 0.5 meters, maintaining consistency with observed forest-avalanche interactions.

forest type	crown coverage*	roughness**	K-value***	Code
evergreen / mixed	dense (> 70%) coverage	rough (stumps/shrubs/saplings); height > 100 cm	48	A
		knobby (scree/stepped/seedlings); height 20 - 100 cm	38	B
		smooth (grass/leaves/smooth rock); height < 20 cm	28	C
	scattered, grouped (40% - 70%) coverage	rough (stumps/shrubs/saplings); height > 100 cm	43	D
		knobby (scree/stepped/seedlings); height 20 - 100 cm	33	E
		smooth (grass/leaves/smooth rock); height < 20 cm	23	F
	open (20% - 40%) coverage	rough (stumps/shrubs/saplings); height > 100 cm	38	G
		knobby (scree/stepped/seedlings); height 20 - 100 cm	28	H
		smooth (grass/leaves/smooth rock); height < 20 cm	18	I
larch / deciduous trees	dense (> 70%) coverage	rough (stumps/shrubs/saplings); height > 100 cm	35	J
		knobby (scree/stepped/seedlings); height 20 - 100 cm	25	K
		smooth (grass/leaves/smooth rock); height < 20 cm	15	L
	scattered, grouped (40% - 70%) coverage	rough (stumps/shrubs/saplings); height > 100 cm	30	M
		knobby (scree/stepped/seedlings); height 20 - 100 cm	20	N
		smooth (grass/leaves/smooth rock); height < 20 cm	10	O
	open (20% - 40%) coverage	rough (stumps/shrubs/saplings); height > 100 cm	25	P
		knobby (scree/stepped/seedlings); height 20 - 100 cm	15	Q
		smooth (grass/leaves/smooth rock); height < 20 cm	5	R

Table 7.2: K_Ψ -value codes for detrainment model. The codes depend on forest density and ground roughness in the forest.

Figure 7.3: **The detrainment coefficient K_Ψ is based on both the density (stem density) of the forest, as well as the surface roughness. The pictures above help the user define the correct category. They are contained in the Forest Detrainment Tab**

7.3 The Forest Detrainment Tab (Input)

The **Forest Tab** is organized into **four key input areas** (Figure 7.4), each designed to capture the essential characteristics of forest stands that influence avalanche behavior.

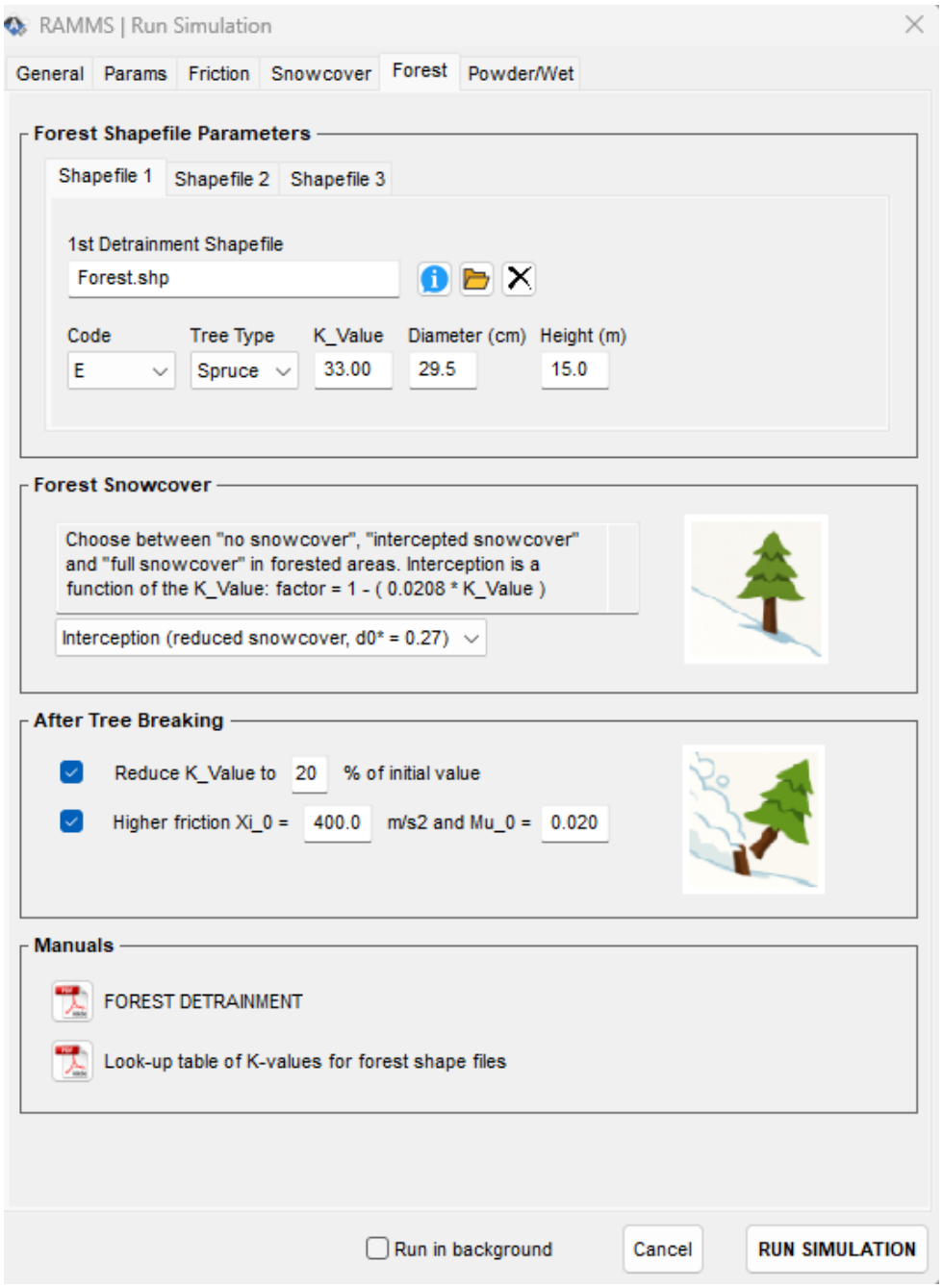
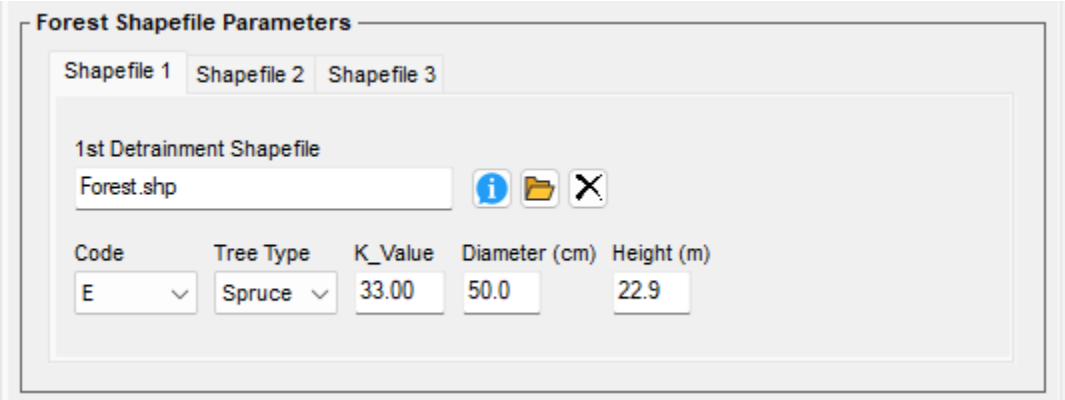


Figure 7.4: The Forest Tab contains four sections: Forest Parameters, Forest Snowcover, After Tree Breaking and On-line Manuals.

7.3.1 Forest Shapefile Parameters

At the top of the interface, you'll find the **forest shapefile menu**, where you can define **up to three forest polygon shapefiles**.

7.3.1.1 Detrainment Shapefiles



These polygons represent distinct forested zones and can be:

- **Drawn manually** using the built-in shapefile tool, or
- **Imported** via the shapefile import tool.

Remember: a single shapefile can include multiple polygons, each representing a different forest patch.

The user can provide three **individual polygon shapefiles** and then define the shapefile parameters. Alternatively, the user can define a **polygon shapefile** that already contains all three key parameters within its **attribute table**. Make sure these exact attribute names are used: **K_Value**, **Tree_Type**, and **BHD**.

7.3.1.2 Assigning the Detrainment Coefficient (K_Value)

Each shapefile must be linked to a **detrianment coefficient (K_Value)**, which reflects the forest's capacity to obstruct snow flow. This value can be:

- Selected from a **reference PDF** (Table 7.2).

The **K_Value** is essential for calculating how much snow mass is "detained" by the forest and influences flow height and momentum reduction. When the K_Value code (A-R) is selected, the numerical value appears in the K_Value tab.

7.3.1.3 Specifying Tree Type

Next, you must define the **dominant tree species** within each polygon. Choose from: **Spruce, Pine:, Larch and Birch.**

Tree Type	Breaking Stress (MPa) σ_Ψ
Spruce	72
Pine	93
Larch	109
Birch	134

Table 7.3: Tree breaking stress values used in RAMMS::Extended.

The tree type is critical—each species has a **distinct fracture or breaking stress** (see Table 7.3), which determines how resistant it is to avalanche impact. This stress threshold defines the load level at which trees are damaged or destroyed during an avalanche event. The breaking stress threshold is compared to the bending stress at the base of the tree arising from both core σ_Φ^b and powder impact:

$$\sigma_\Phi^b = \frac{M_\Phi\, y}{I}$$

$$\sigma_\Pi^b = \frac{M_\Pi\, y}{I}$$

where $(M_\Phi\,, M_\Pi\,)$ are the bending moments arising from the input forces. We assume the trees break at the outer edge $y = d/2$ and the trees are circular sections $I = \pi d^4/64$. **The diameter d of the trees is given by the diameter at breast height DBH**. Trees break when

$$\left(\sigma_\Phi^b, \sigma_\Pi^b\right) > \sigma_\Psi.$$

The bending moments at the tree base are governed by $M_\Phi = F_\Phi\, l_\Phi$ and $M_\Pi = F_\Pi\, l_\Pi$ where $(F_\Phi\,, F_\Pi\,)$ are the total core/powder forces acting on the tree stem and $(l_\Phi\,, l_\Pi\,)$ are the corresponding moment arms. The total forces $(F_\Phi\,, F_\Pi\,)$ are given by the integration of the pressure distributions $(p_\Phi\,, p_\Pi\,)$ over the height and width of the tree stem.

RAMMS::Extended distinguishes between the core and powder cloud loads, even using different impact widths to account for the effect of branches extending from the primary tree stem.

For the **avalanche core $\boldsymbol{\Phi}$** , we assume

- The **impact pressure** p_Φ is given by the empirical pressure formula with $C_d = 1$

$$p_\Phi = \frac{1}{2}\rho_\Phi C_d \mathrm{V}_\Phi^2 = \frac{1}{2}\rho_\Phi \mathrm{V}_\Phi^2$$

  The assumption $C_d = 1$ is in good agreement with values from the deflection work-energy method around circular sections (see chapter 6).

- The trees are assumed to grow vertically from the slope (not slope perpendicular); therefore, we must take the $\cos\gamma$ component of the impact pressure.

- The **density $\boldsymbol{\rho_\Phi}$** of the avalanche depends on the state of fluidization and is $\rho_\Phi \leq \hat{\rho}_\Phi \leq 450$kg/m^3. The density profile is constant over the height of impact.

- The **impact width** is the stem diameter d.

- The **height of the impact** is not only the height of the avalanche core but increases during the impact/stagnation process. The energy equation is used to determine the height of impact,

$$h = h_\Phi + \frac{\mathrm{V}_\Phi^2}{2g\lambda} b(d, h_\Phi)$$

  with λ =1.5 and $b(d, h_\Phi)$ =0.1. This formula was proposed by Salm and used by Feistl et al. (2015). We take the moment arm height to be $l_\Phi = h/2$.
- The **pressure distribution $\boldsymbol{p_\Phi}$ is constant** over the height h.
- A **dynamic magnification factor** is applied $D = 2$. We therefore assume that inertial forces in the tree are excited by the impulsive loading of the core.

With these assumptions the breaking stress for the avalanche core is

$$\sigma_\Phi^b = 2\,\rho_\Phi \frac{8\mathrm{V}_\Phi^2}{\pi d^2} h^2 \cos\gamma$$

The calculation of the bending stress for the avalanche core is straightforward because we assume a constant density and velocity profile.

For the **avalanche cloud $\boldsymbol{\Pi}$** , we assume a linear density profile, and a parabolic velocity profile,

$$\rho_{\Pi}(z) = (2\bar{\rho}_{\Pi} - \rho_{\Lambda})\left(1 - \frac{z}{h_{\Pi}}\right) + \rho_{\Lambda}$$

$$V_{\Pi}(z) = \frac{3}{2}\bar{V}_{\Pi}\left(1 - \frac{z^2}{{h_{\Pi}}^2}\right)$$

The velocity $V_{\Pi}(z)$ is the laminar velocity. To this velocity we add the turbulent velocity $V_{\Pi}^{T}(z)$

$$V_{\Pi}^{T}(z) = \sqrt{\frac{2R_{\Pi}(\mathrm{z})}{3\rho_{\Pi}(z)}}$$

to calculate the powder avalanche impact pressure. The profile of turbulent energy $R_{\Pi}(\mathrm{z})$ is described in section 5.6 of the manual. It is a function of air entrainment at the upper surface of the cloud, internal shearing in the cloud and the transfer of fluctuation energy from the core. **The factor 3 in the denominator arises because we take only 1/3 of the turbulent energy in the direction of the avalanche velocity $\boldsymbol{V_{\Pi}(z)}$.** The total powder pressure is calculated as

$$p_{\Pi}(z) = \frac{1}{2}\rho_{\Pi}(z)[V_{\Pi}(z) + V_{\Pi}^{T}(z)]^2$$

To calculate the bending moment in the tree, we determine the integral

$$M_{\Pi} = \left[\int_0^H p_{\Pi}(z) z dz\right] w \cos\gamma$$

and apply this value to determine the bending stress at the base of a circular tree stem,

$$\sigma_{\Pi}^{b} = \frac{M_{\Pi}\ {}^{d}/_{2}}{I} = \frac{32\ M_{\Pi}}{\pi d^3}$$

The upper bound of the integral is the tree height H if the height of the powder cloud is higher than the tree, $h_{\Pi} \geq H$; however, the upper bound is equal to the powder cloud height, if the cloud is smaller than H.

To accurately model and calculate the **tree loading width**, we use the following empirical formula derived from Indermühle (1978) for the crown width (Figure 7.5):

$$w = 0.415 + 0.045d$$

where:

- w is the **loading width** in meters

- d is the **tree diameter** in centimeters

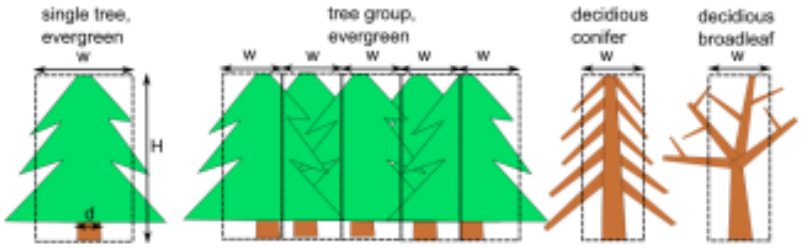


Figure 7.5: Default values of tree width for pressure calculations.

This formula serves as the **baseline** and is primarily calibrated for **Spruce** trees. It is adjusted to account for other tree types, depending on the winter foliage, see Table 7.4.

Tree Type	Loading Width w	Loading Height H
Spruce	Default	Default
Pine	Default	Default
Larch	50% of Spruce	Default
Birch	50% of Spruce	70% of Spruce

Table 7.4: Default values of tree width and tree height for pressure calculations.

When the **Turbulence option is switched OFF**, the same procedure is applied except that in this case $V_{\Pi}^{T} = 0$. **In general, tree bending stresses – and therefore tree destruction - are much smaller when the turbulence option is OFF.**

7.3.1.4 Tree Diameter (BHD) and Tree Height

The final required input is the **tree diameter at breast height (BHD)**, entered in **centimeters (cm)**. This metric serves as a proxy for tree maturity and mass. It is an important variable in the impact calculations, especially when the powder cloud height is higher than the tree height and the tree is fully loaded.

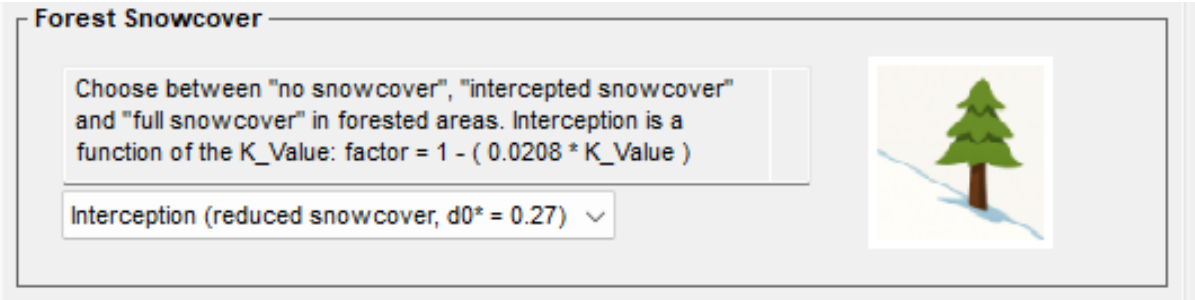
Using the inputted **BHD value**, the **tree height (H)** is automatically calculated using the formula:

$$H = Tree\ height = (BHD)^{0.8}$$

Note: BHD is in **cm**, and the resulting height H is in **meters (m)**. The height of birch trees is reduced, whereas the height of spruce, pine and larch are all calculated from this formula (see Table 7.4).

7.3.1.5 Snowcover Entrainment in Forested Areas

Entrainment of the snowcover (see Chapter 4) plays a critical role in **RAMMS::Extended** simulations, particularly when modeling avalanche behavior through vegetated terrain. **To account for snowcover variations within forested zones**, **RAMMS::Extended** provides three configurable options that can be specified by a pull-down menu. **Each pull-down menu contains the height of the corresponding erodible layer.**



1. **No Snowcover in Forested Area**
   This option assumes that there is no entrainable snow beneath the forest canopy. It should be used when the snowcover under trees is considered heavily compacted, icy, or otherwise resistant to entrainment due to canopy interception, repeated melting and refreezing, or ground-level consolidation. Selecting this setting effectively excludes any snow mass contribution from the forest floor within the polygon.

2. **Interception**
   This intermediate option accounts for partial snow interception by the forest canopy. The available snowcover is scaled by a factor of $(1 - 0.0208K)$, where K is the **detrainment coefficient**. This **scaling reduces the entrainable snow mass** within the forest polygon as determined by the standard RAMMS::Extended snowcover distribution algorithm (see Chapter 4). Since snow depth is inherently dependent on **slope angle** and **elevation**, applying this scaling produces a spatially variable, non-uniform snowcover beneath the forest canopy — more closely resembling natural conditions.

3. **Full Snowcover in Forested Area**
   This option applies no correction within the forest polygon. The snowcover distribution remains **fully determined by the standard RAMMS::Extended algorithm**, which considers both slope and elevation to assign snow depth. It is appropriate in situations where snow accumulation and entrainment potential

beneath the forest is **assumed to be equivalent to that in open terrain. That is, for very sparse vegetation**.

7.3.2 After Tree Breaking

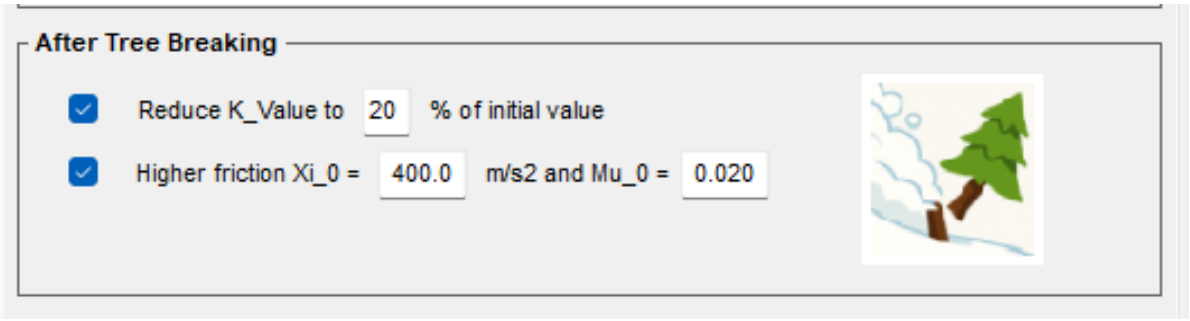
In **RAMMS::Extended**, trees are considered broken when the calculated bending stresses—σ_{Φ}^{b} and σ_{Π}^{b} —exceed the tree's breaking stress $\left(\sigma_{\Phi}^{b}, \sigma_{\Pi}^{b}\right) > \sigma_{\Psi}$. In other words, when the stress on a tree surpasses its structural limit, it breaks.

This raises the important question: **what should happen once the trees are broken?** There are two competing mechanisms to consider:

1. **Reduced Detrainment:** Once trees are broken, the detrainment coefficient K_{Ψ} can no longer remain the same—it must be reduced to reflect the decreased ability of the forest to trap material.

2. **Increased Flow Resistance:** The broken tree stems can jam within the avalanche flow, creating additional resistance.

RAMMS::Extended allows users to account for either or both effects. These options can be configured independently or together using the two input fields found under the Forest Tab, as shown below.



1. Reduce K_value (K_{Ψ})

When this option is enabled, users can specify the **K_Value percentage** to be applied *after* tree breakage. By default, the K_Value is reduced to **20%** of its original value.

For example, if the initial K_Value was **40**, it would drop to **8** once the trees are broken—reflecting the reduced detrainment capacity of the damaged forest.

This setting allows for flexible modeling of how much the forest's ability to interact with avalanche flow is diminished after structural failure.

2. Use higher friction

When this option is enabled, the user can define custom values for ξ_0 and μ_0.

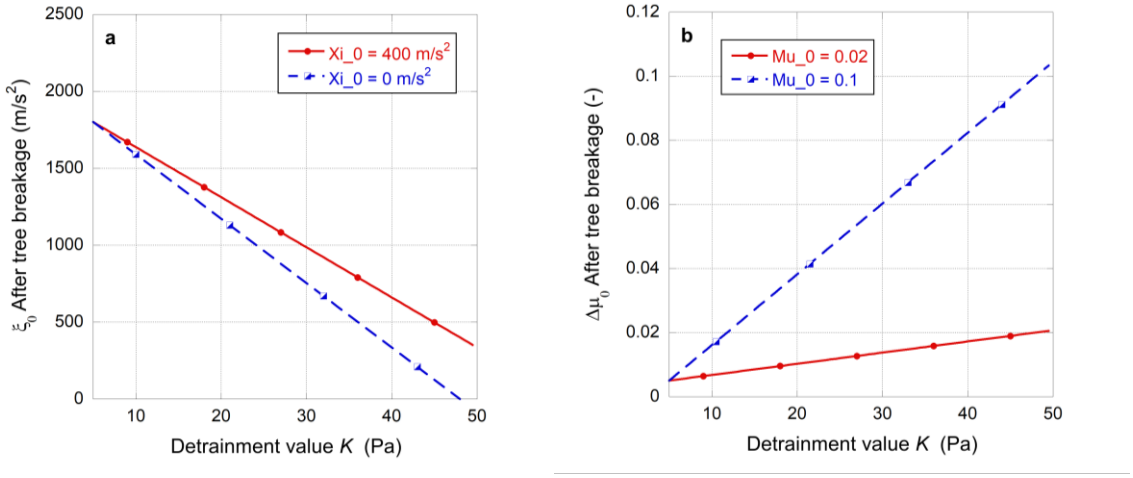


Figure 7.6: Relationship between the detrainment coefficient K_Ψ and the friction coefficients $\xi_0(K_\Psi)$ and $\Delta\mu_0(K_\Psi)$. Recommended values for Xi_0 are between 400m/s2 and 0m/s2. Recommended values for Mu_0 are between 0.02 and 0.1.

The parameter **Xi_0** represents the ξ_0-value corresponding to the **maximum K_Value** of 48 (Pa). From there, the Xi-value increases linearly up to **1800**m/s² as the **K_Value decreases to its minimum of 5** (Pa).

The equation for the ξ_0-value valid after tree breakage is

$$\xi_0(K) = Xi\_0 + \frac{1800 - Xi\_0}{5 - 48}(K - 48)$$

where Xi_0 is the input value. In Figure 7.6 we depict two possible Xi_0 values, Xi_0=400m/s² and Xi_0=0m/s². The default value is Xi_0=400m/s².

In this configuration, the **Xi-value becomes a dynamic function of the K_Value**, allowing for a smooth, continuous representation of how K weakens after tree breakage.

A similar approach is applied to **Mu_0**. This parameter defines the **delta value** $\Delta\mu_0$ **associated with the maximum K_Value of 48 (Pa)**. As the K_Value decreases—representing reduced forest interaction—**the** $\Delta\mu_0$**- value decreases linearly**, reaching a minimum of **0.005 when K_Value = 5**. The equation for the $\Delta\mu_0$-value valid after tree break is

$$\Delta\mu_0(K) = Mu\_0 + \frac{0.005 - Mu\_0}{5 - 48}(K - 48)$$

The value of $\Delta\mu_0(K)$ is added the μ_0 defined in the friction tab $\mu_0' = \mu_0 + \Delta\mu_0(K)$.

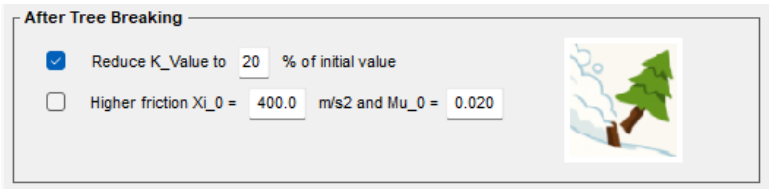
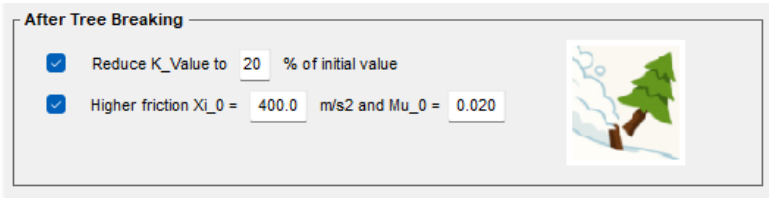
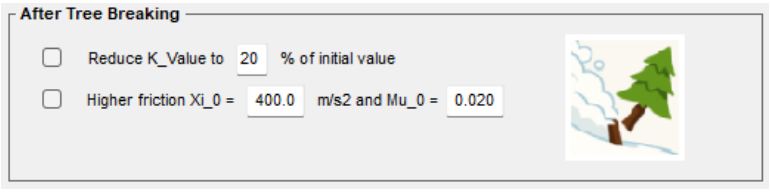
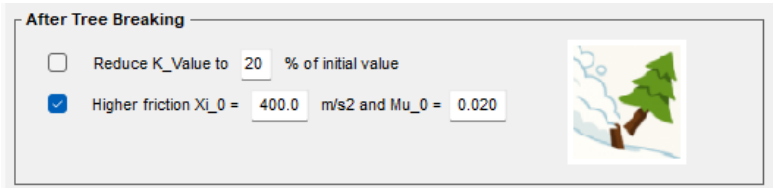
	**Least Friction**
	**Mixed Friction**
	**Mixed Friction**
	**Most Friction**

Table 7.5: Combinations of reduce K_value (K_Ψ) and Increase Friction that cause the least and the most friction.

Recommended values for Xi_0 are between 400m/s2 and 0m/s2. Recommended values for $\Delta\mu_0$ are between 0.02 and 0.1. These combinations of Reduce K_value and Higher Friction cause the least to most friction (Table 7.5).

The two middle options introduce **competing processes (mixed friction).** The decrease in K_Ψ (less detrainment) works against the increase in friction. Or, alternatively, a non-reduced K_Ψ helps stop the avalanche, but without an increase in friction (Table 7.5).

7.3.3 Visualization of Forest Interaction

RAMMS::Extended offers **two distinct methods** for visualizing forest interaction data during avalanche simulations. Users can explore four primary datasets that describe the dynamic interaction between avalanche flow and forest stands:

1. **Forest Detrainment Heights**

2. **Stem Bending Stress**

3. **Maximum Stem Bending Stress**

4. **Tree Destruction Status**

These datasets can be accessed in two ways:

- Via the **Display Tab** on the right-hand side of the graphical user interface (see Figure 7.7), or

- Through the **Results Menu** under *Forest* → ..., where the desired output field can be selected directly (see Figure 7.8).

Each dataset is defined mathematically as follows:

1. **Detrainment Height**: $M_{\Phi \to \Psi}$, representing the time-dependent snow mass deposited beneath the forest canopy. Detrainment heights are reported in units of meters (m), see Figure 7.9.

2. **Stem Bending Stress**: $\left(\sigma_{\Phi}^{b}, \sigma_{\Pi}^{b}\right)$, the dynamic mechanical stress exerted on tree stems by avalanche impact.

3. **Maximum Stem Bending Stress**: The peak value of $\left(\sigma_{\Phi}^{b}, \sigma_{\Pi}^{b}\right)$ recorded over the simulation duration, see Figure 7.10.

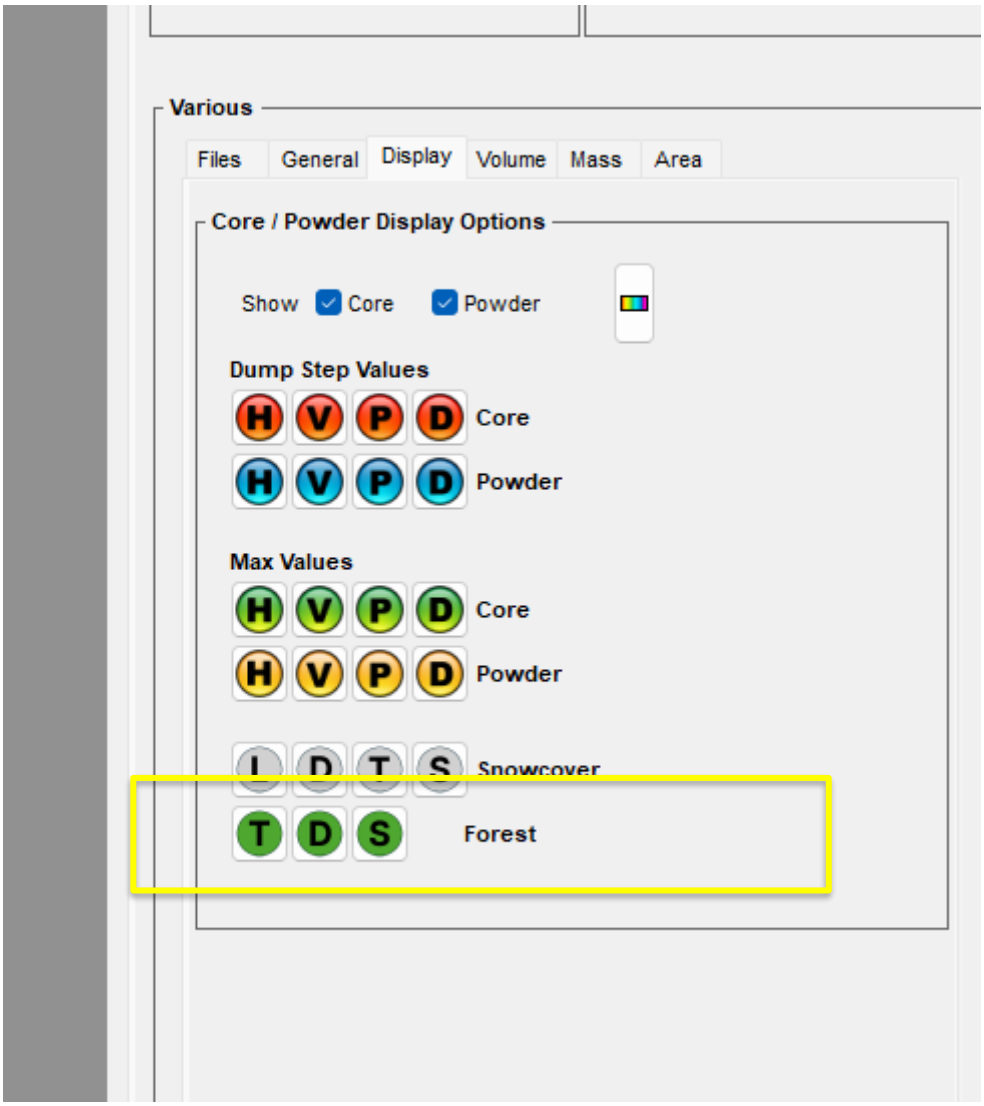


Figure 7.7: The Display tab on the right-hand side of the GUI. At the bottom, three green buttons—T (Tree Destruction), D (Detrainment Heights), and S (Maximum Bending Stress)—allow quick access to forest interaction outputs. This interface enables users to easily toggle between visualization layers. Note that Tree Destruction and Detrainment Heights are time-dependent fields; to view meaningful results, the time slider must be moved forward from $t = 0$.

4. **Tree Destruction**: A categorical output (Figure 7.11 and Figure 7.12) based on the comparison of stem stress to the breaking threshold σ_Ψ:
    - **Value 1:** Destruction by avalanche core pressure $\sigma_\Phi^b > \sigma_\Psi$
    - **Value 2:** Destruction by powder cloud pressure $\sigma_\Pi^b > \sigma_\Psi$

- **Value 3:** Both core and powder cloud pressures exceeded the breaking stress $\left(\sigma_{\Phi}^{b}, \sigma_{\Pi}^{b}\right) > \sigma_{\Psi}$

Of these four outputs, three—**detrainment height**, **stem stress**, and **tree destruction**—are **time-dependent** and evolve throughout the simulation. To properly visualize their progression, the **time slider** must be used. While users are often interested in the final state of the forest, this can be quickly accessed by pressing the **"Go to End of Simulation"** button (below the animate or play icon on the left-hand side of the GUI) or manually advancing the time slider to the desired moment.

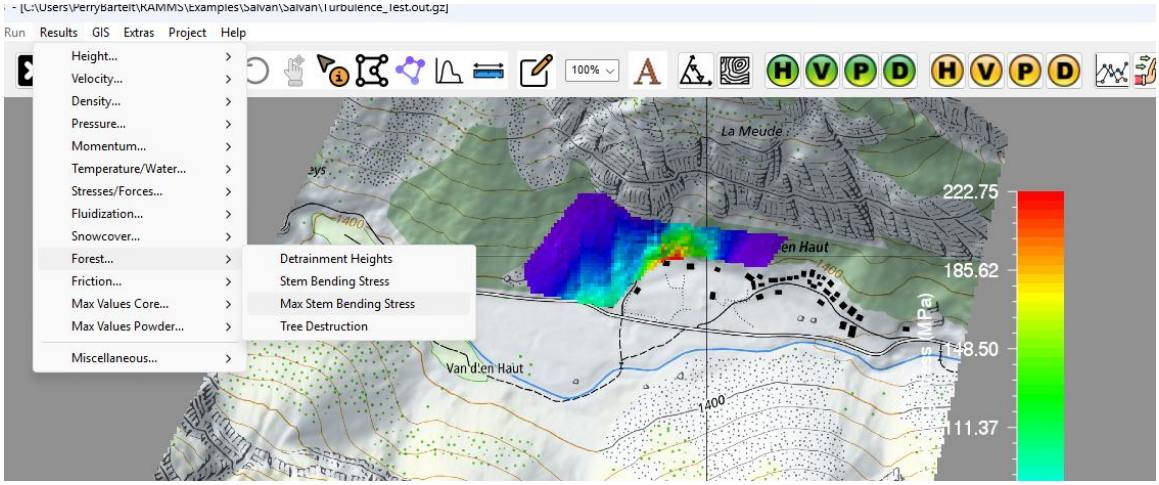


Figure 7.8: Forest interaction output can also be visualized via the Results menu under Forest.

Time-dependent outputs can also be animated, making them particularly valuable for creating instructional videos or post-event visualizations. Animating the **tree destruction** layer, for example, reveals the sequence in which trees are impacted—providing valuable insight into the spatial and temporal evolution of avalanche-forest interactions.

7.3.4 Detrained Snow

The amount of snow deposited in the forest can be visualized using the detrained snow visualization option (Figure 7.9). Detrained snow heights are never over the maximum of 0.5m, which represents an average value detrained in the cell. Higher detrainment values are found immediately in front of the trees.

Snow is detrained both at the lateral edges of the flow and in the runout zone. Snow is only detrained within the specified forest polygon.

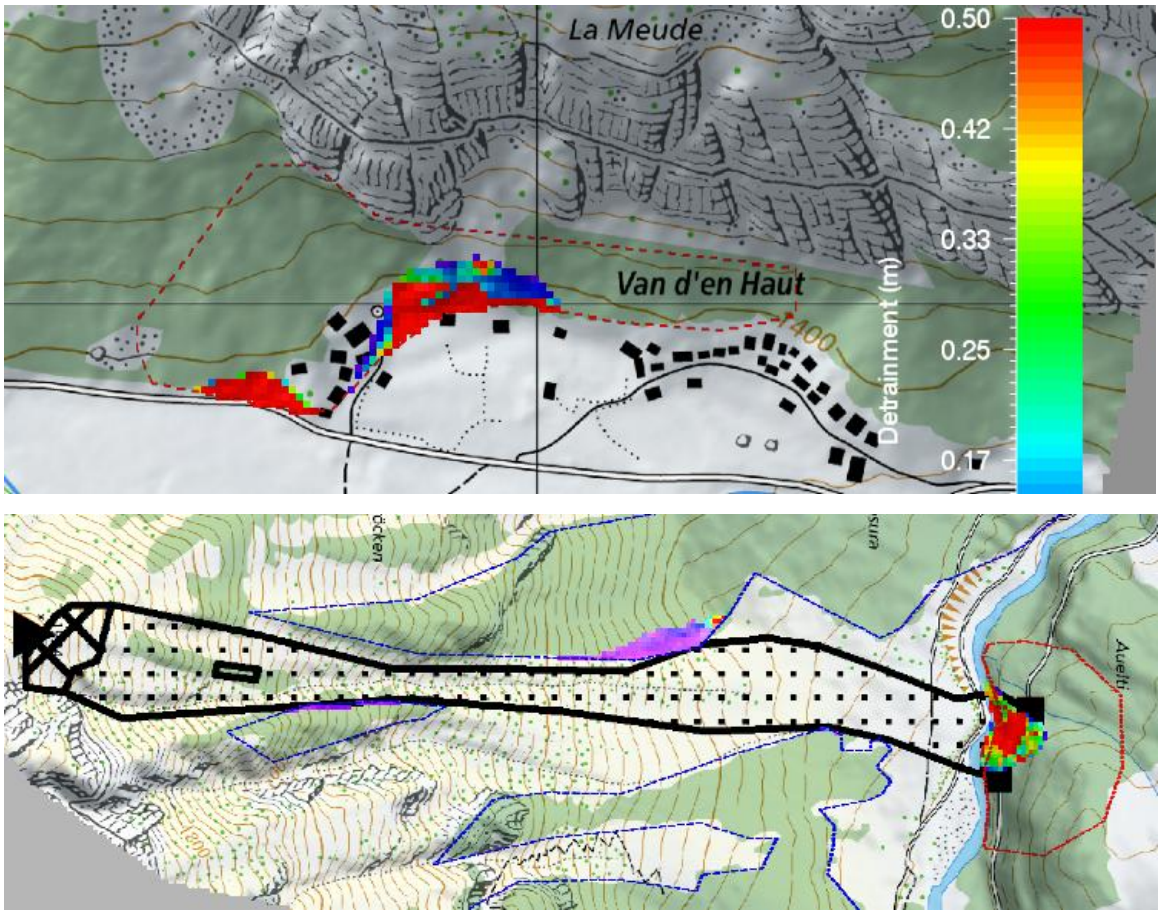


Figure 7.9: Visualization of detrained (stopped) snow within the forest area. This output is generated by selecting the D button (Detrainment Heights) from the forest tools in the Display tab. The forest polygon shapefile is shown as a dashed outline; to make it visible, the user must select and activate the corresponding shapefile in the File tab. Note that detrained snow heights never exceed 0.5 meters. Snow is not necessarily detrained after the tree stand is destroyed, see Reduce K_value After Tree Breaking.

7.3.5 Breaking Stress / Max Breaking Stress

Visualizations of stem bending stress depict the **distribution of bending moments at the outer edges of tree stems** (Figure 7.10), expressed in megapascals (MPa). These values reflect the mechanical response of trees to simulated loading conditions and indicate areas under tension or compression, but they **do not represent failure thresholds**. Bending stresses are only shown within the forest polygon shapefile. For improved spatial interpretation, contour lines can be enabled to highlight gradients in stress intensity. Users seeking a time-resolved view of stress development can explore individual simulation steps through the **Results → Forests** menu.

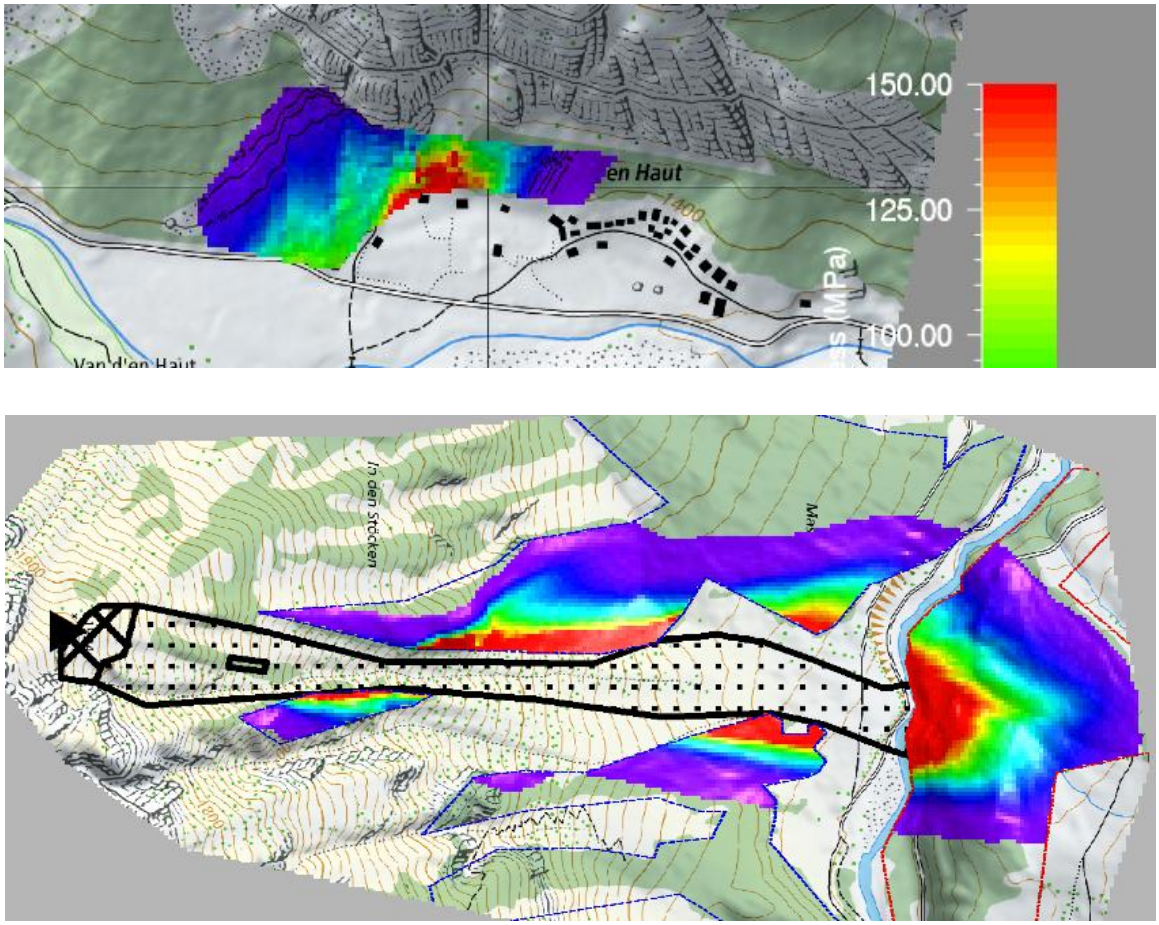


Figure 7.10: Visualization of maximum stem bending stresses within the forest polygon shapefile. This output displays the peak bending stress values experienced during the simulation and can be enhanced using the contour line option for improved spatial interpretation. For a time-resolved view of bending stresses, users can access individual time steps via the Results → Forests menu. Bending stresses only appear within the polygon shapefile.

7.3.6 Forest Destruction

Tree destruction visualizations represent a categorical output that identifies areas where stem-breaking thresholds have been exceeded due to avalanche forces (Figure 7.11 and Figure 7.12). Destruction is classified based on the source of pressure: **core flow (value = 1)**, **powder cloud (value = 2),** or both components **combined (value = 3)**. These classifications are determined by comparing simulated bending stresses against the breaking threshold (σ_Ψ), indicating whether failure was caused by the dense core

$$\sigma_\Phi^b > \sigma_\Psi,$$

the powder cloud ($\sigma_\Pi^b > \sigma_\Psi$), or a combination of both. All destruction is limited to the designated forest polygon shapefile. Users can activate this output via the green **T** button in the *Display* tab. Color-coded areas, such as red for powder cloud destruction, help distinguish the mode of impact.

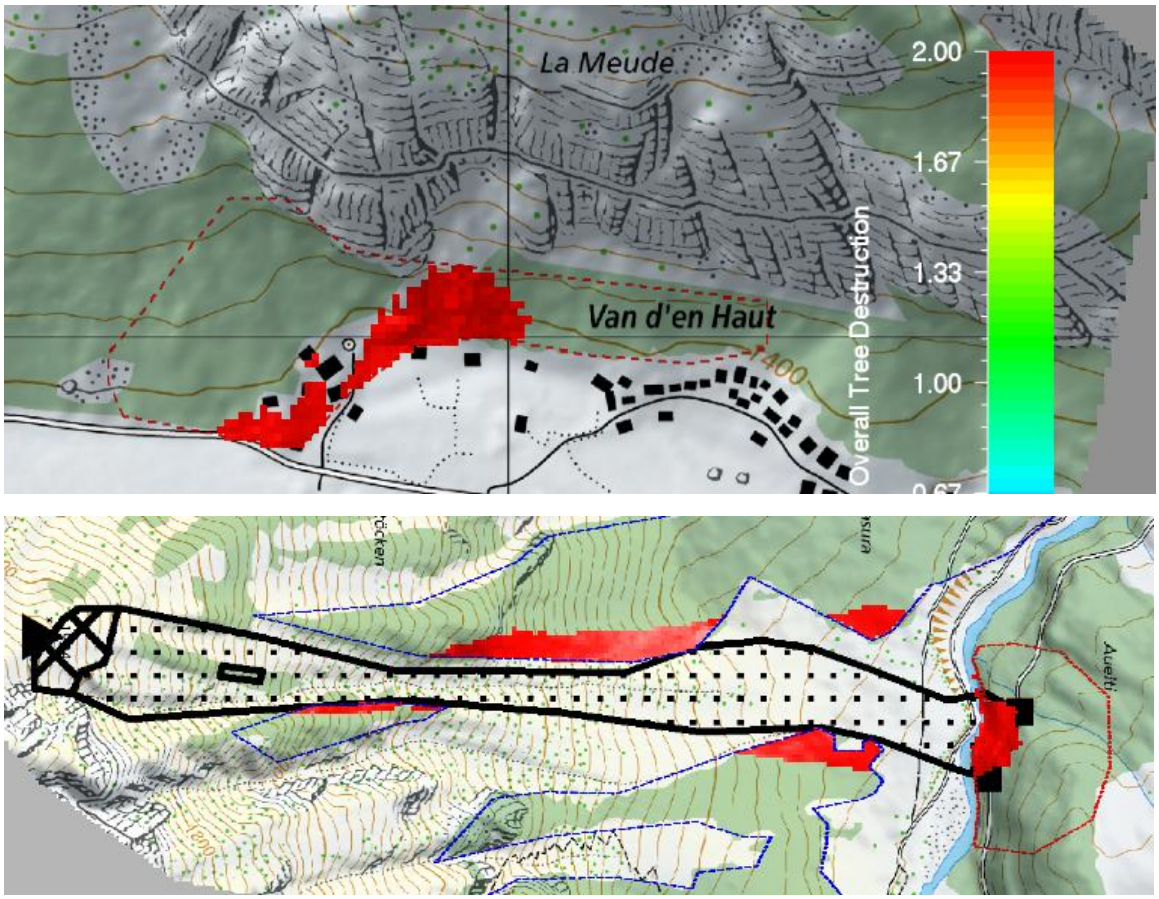


Figure 7.11: Visualization of tree destruction within the forested area. This output is activated by clicking the green T button in the Display tab. In this example, the affected area is highlighted in red (value = 2), indicating that trees were destroyed by the powder cloud component of the avalanche. All destruction occurs within the bounds of the designated forest polygon shapefile. In the lower figure (Masura, Klosters) two different forest shapefiles have been defined in red and blue.

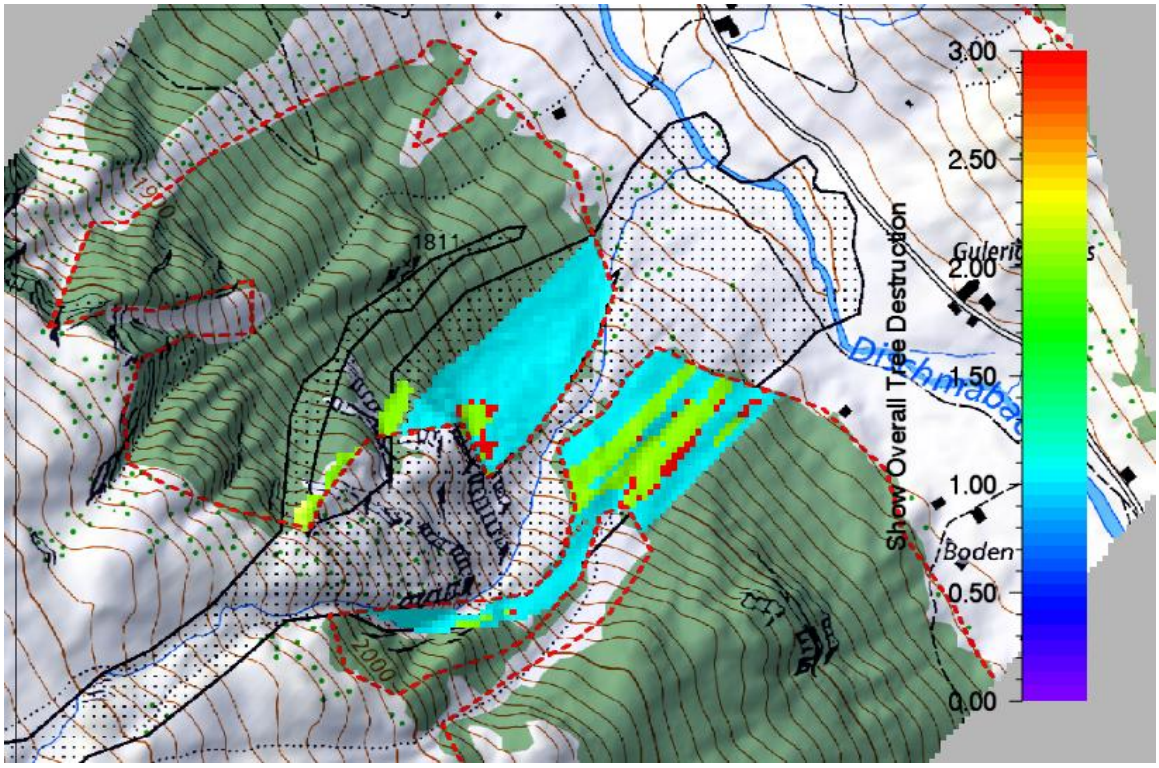


Figure 7.12: The forest is destroyed by the core (value = 1), the cloud (value = 2) and both (value = 3). Avalanche in the Dischma valley near Davos.

7.3.7 Volume and Mass Balance

The simulation provides detailed insights into how much avalanche snow is retained within forested terrain. Specifically, it is possible to visualize both the *location* and *quantity* of snow that is detrained in each forest polygon, denoted as $M_{\Phi \to \Psi}$.

In many cases, it is critical to assess the **total volume or total mass** of avalanche snow intercepted and held back by the forest. To support this, the **GUI** offers dedicated **Volume** and **Mass** tabs located on the right-hand side of the interface (Figure 7.13). These tabs are located directly to the right of the **Display tab**, containing the Forest buttons for tree destruction, detrainment heights and max bending stress.

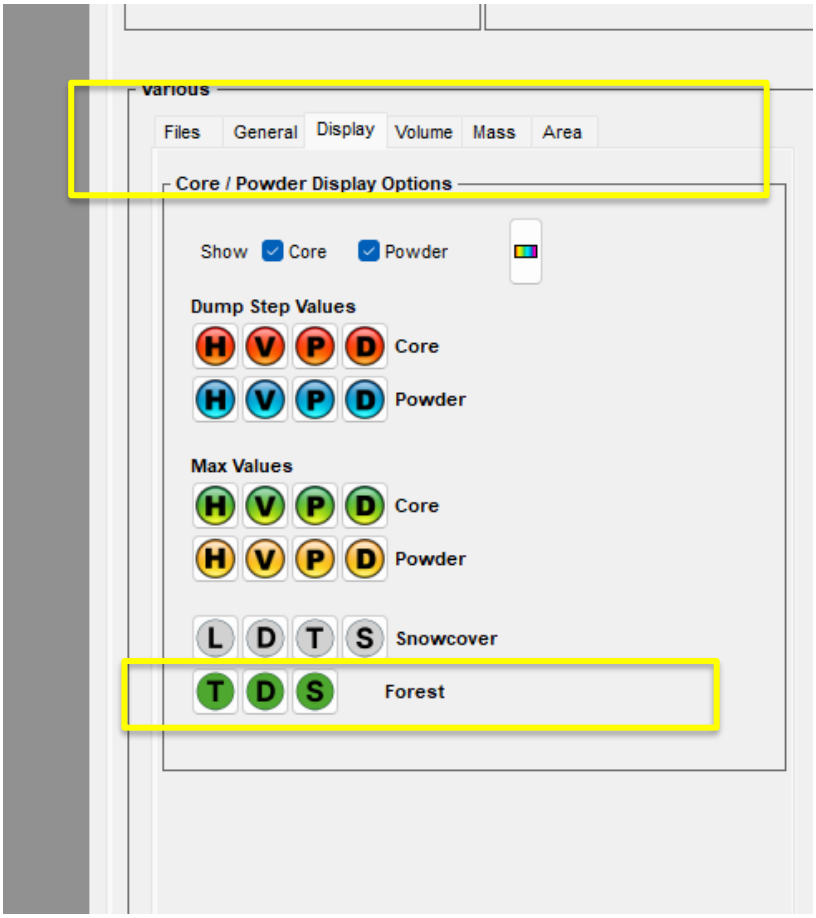


Figure 7.13: The Volume and Mass tabs are located to the left of the Display tab containing the green Forest buttons. The Volume and Mass tabs can be used to determine the total amount of snow detrained in the forest.

When either of these tabs is selected and the **Plot** option is enabled, a corresponding plot is generated (Figure 7.14 and Figure 7.15). These plots display the **total volume** or **mass** of snow retained by the forest under the label **"Forest"**. Notably, the plotted lines in both cases appear **negative**, reflecting the fact that snow is being **extracted** from the avalanche flow—representing a loss in momentum and mass due to forest interaction.

- **Volume data** is expressed in cubic meters (m^3) of snow, based on a predefined co-volume snow density ($\hat{\rho}_\Phi = 450 g/m^3$).
- **Mass data** is presented in **metric tons**.

For precise numerical values, refer to the field labeled **"Forest buildup mass"** (Figure 7.16). These values are not only shown in the graphical plot but are also **updated in real time** during the simulation run. Within the **Run Window**, the current forest detainment data appears under the heading **(FOREST BUILDUP)**.

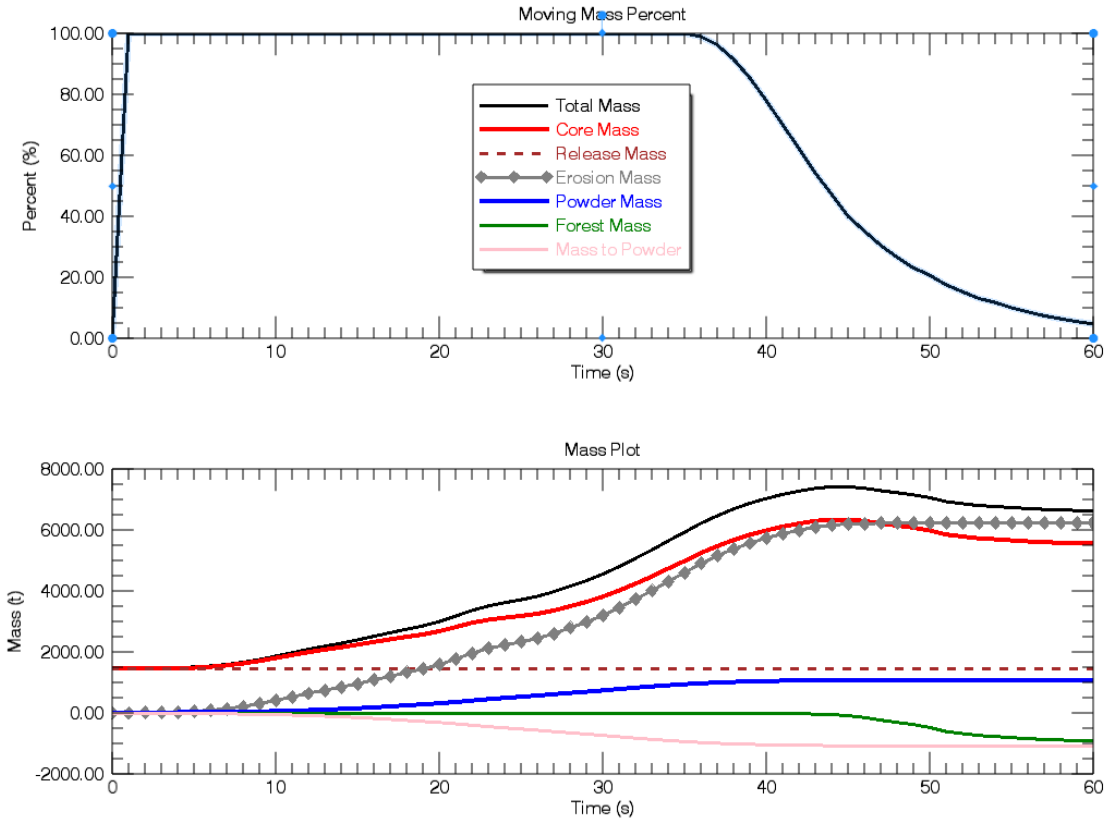


Figure 7.14: The Mass Plot is depicted with the plot option is activated in the Mass tab. The mass of detrained snow is shown as the red line in the lower panel. It is negative indicating the amount of core mass extracted from the avalanche. Note that the total mass decreases as mass is detrained in the forest (green line).

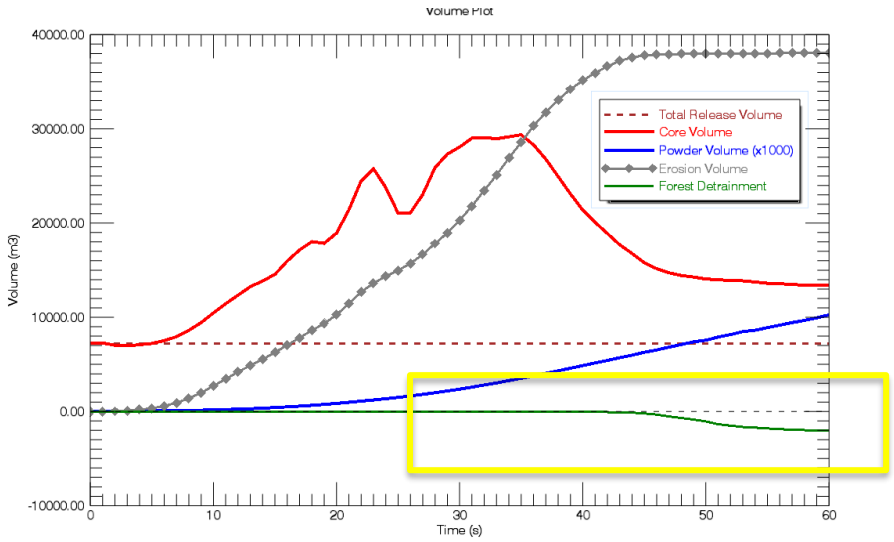


Figure 7.15: The Volume Plot is depicted with the plot option activated in the Volume tab. The volume of detrained snow is shown as the green line. It is negative indicating the amount of snow volume extracted from the avalanche.

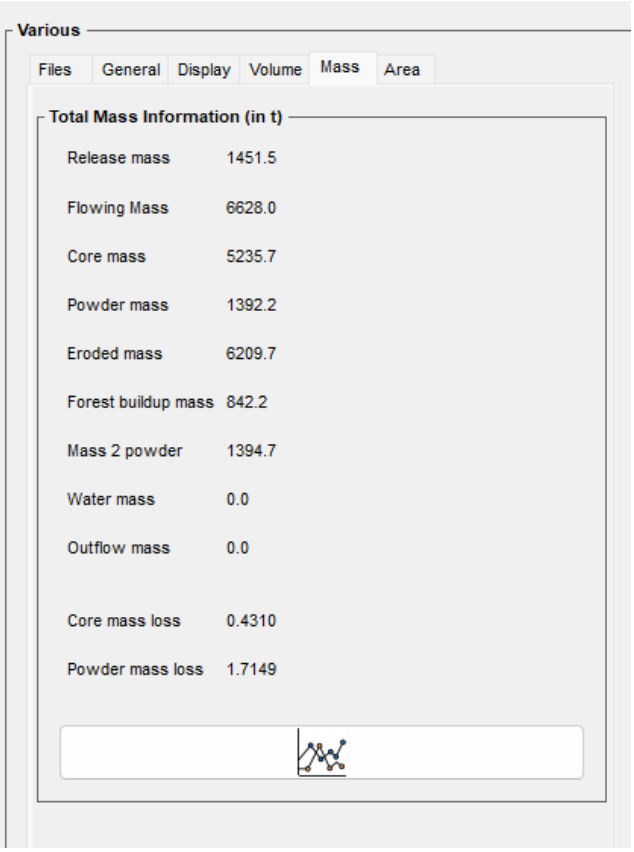


Figure 7.16: The Mass tab showing the numerical values of the Forest buildup mass. These values change when the time slider is moved. When the plot button is selected the total mass balance plot appears.

Additionally, the GUI displays the **percentage of total avalanche mass** that has been stopped by the forest, providing an immediate sense of the forest's effectiveness as a natural barrier (Figure 7.17).

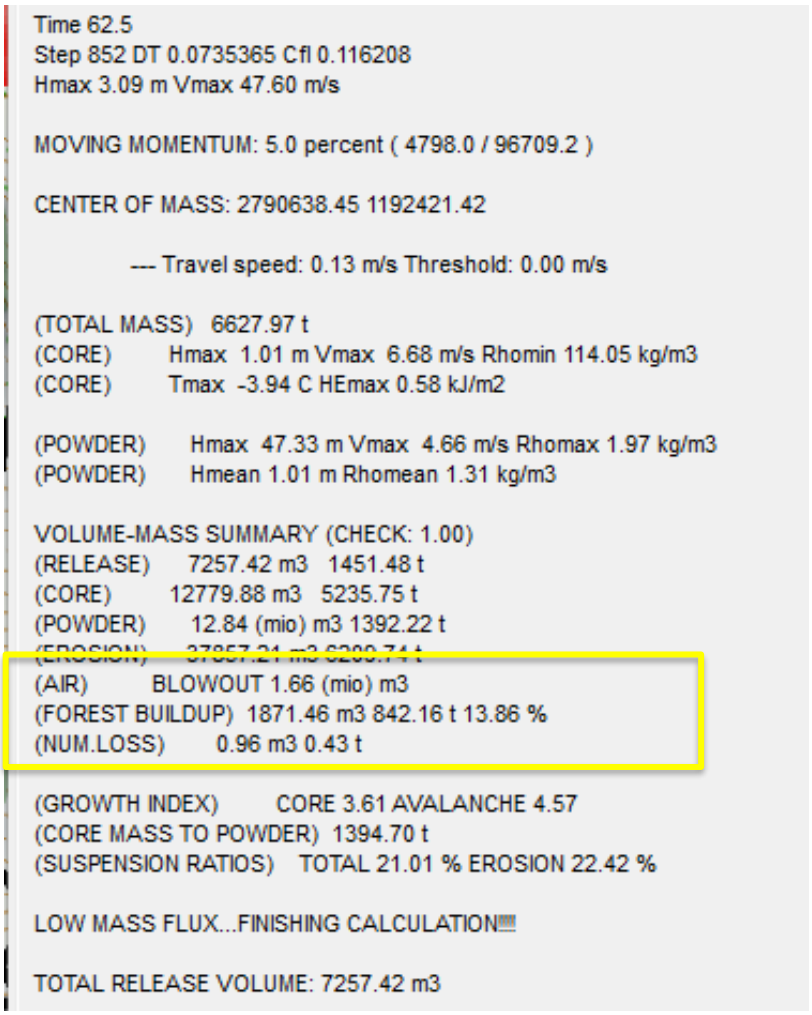


Figure 7.17: The Run Simulation window output indicating the amount of mass stopped by the forest.

Chapter 8 Wet Snow Avalanches

8.1 Introduction

Wet snow avalanches present one of the most difficult challenges for avalanche modelling. Unlike dry, fluidized avalanches—which are relatively well understood and widely measured—**wet avalanches lack robust field data, particularly on snowpack water content in release and entrainment zones**. In practice, only qualitative estimates of "wetness" are possible, and the processes that govern transitions from dry to wet flow regimes remain highly uncertain.

The presence of liquid water on snow grains fundamentally alters **avalanche mechanics**. Frictional shear heating, rainfall, or entrainment of isothermal snowpacks can all introduce meltwater into the flowing mass. At first, thin films of water raise capillary pressures, causing particles to stick together in agglomerates. This increases internal resistance and suppresses the dispersion of snow grains. At the same time, basal sliding friction is reduced by lubrication, **allowing avalanches to run long distances at relatively low velocities**.

In **RAMMS::Extended**, these processes are represented through a modification of the random kinetic energy balance. Both the **production of random energy** (controlled by the *Generate* parameter, α_Φ) and its **persistence** (controlled by the decay coefficient, β_Φ) are affected:

- **Production is suppressed:** Wet snow cannot easily fluidize. Typical α_Φ values are reduced from α_Φ =0.07–0.08 (dry) to α_Φ =0.03–0.05 (wet).
- **Decay is accelerated:** Once random motion is generated, it is rapidly dissipated through sticky, inelastic collisions. β_Φ roughly doubles from β_Φ~1 s^{-1} (dry) to β_Φ~2 s^{-1} (wet).

The net effect is that fluidization energy decays faster than it can be produced. This explains why wet avalanches rarely form powder clouds, deposit material abruptly at their tails, and exhibit dense, plug-like motion.

A further complication is that "wet snow avalanches" are not a single regime but encompass a spectrum of flow types, depending on how water enters the avalanche:

1. **Frictional-melt avalanches:** Initiate dry but at warm mean snow temperatures (e.g., –3 °C). Meltwater arises primarily from shear heating during flow.
2. **Mixed entrainment avalanches:** Start dry but entrain warm, isothermal snowpacks at lower elevations. Water content increases rapidly as the flow evolves.

3. **Fully wet avalanches:** Release directly from isothermal, water-saturated snowpacks. These flows are highly lubricated at initiation, perhaps bypassing the transitional capillary regime.

Modeling all three types requires careful adjustment of input parameters in the **Snowcover Tab**, **Powder/Wet Tab**, and **Friction Tab**. The user must define initial snow and air temperatures, specify water content and gradients, and set friction parameters that evolve dynamically with wetness.

In summary, wet avalanche modelling in **RAMMS::Extended** is not only a question of specifying friction values, but of capturing the coupled processes of melting, water film formation, energy suppression, and rapid decay. This chapter provides guidance for setting these parameters in practice, with the aim of offering a transparent workflow for users while highlighting the physical reasoning behind the model design.

8.2 Snowcover Temperature and Wetness

The **Snowcover Tab** defines both the thermal state of the snowpack and its liquid water content. These inputs determine how the avalanche core evolves as it entrains snow along its path. Because wet avalanche behavior is sensitive to both variables, careful specification of snowcover conditions is essential.

Defining Water Content

The input for **water content** m_Σ is given as a **percentage of snowcover height** at the reference altitude:

- Example: If the snowcover height is 1.20 m at 2000 m, and the water content is set to m_Σ =1%, then the snowcover contains: .01×1.20 m=0.012 m=12 mm of water

The key point is that the user specifies a **percentage**, not a millimeter value. Because snowcover height varies with **elevation** and **slope angle**, the actual millimeters of water contained in the snowpack also vary across the calculation domain. In steep, shallow snowcovers the water content per unit area will be smaller; in flatter, deeper snowcovers it will be larger.

The **water content gradient** Δm_Σ is specified in **percent per 100 m elevation difference**. This gradient controls how the liquid water fraction changes as the avalanche descends through the calculation domain.

- A **positive gradient** increases water content toward lower elevations, mimicking the effect of rain-on-snow events or isothermal snowpacks that are wetter in valleys than in high alpine starting zones.
- A **negative gradient** is rarely used, but could in principle represent situations where lower elevations are shaded or colder, resulting in reduced liquid content downslope.

Care should be taken when selecting the gradient value. Gradients larger than Δm_Σ = **1% per 100 m** will quickly generate extremely wet snowcovers in the runout zones. For example, starting with 1% water content at 2000 m and applying a gradient of +2%/100 m, the snowcover at 1000 m elevation would contain 21% liquid water—a value well beyond most field observations. Such inputs may produce unrealistic runout behavior dominated by lubrication effects.

As a practical guideline, gradients in the range **0.1–0.5%/100 m** are usually sufficient to represent wetter snowpacks at lower elevations without exaggerating the liquid content. Larger gradients should be reserved for extreme cases, such as heavy rain-on-snow events, and should be used cautiously, as they can overwhelm the rheological feedback mechanisms in the model.

Independence of Temperature and Wetness

Snow temperature (T_Σ) and water content (m_Σ) are defined **independently** in **RAMMS::Extended.** It is therefore possible to specify combinations that at first appear contradictory, for example:

- A **sub-zero snowpack** (e.g. –2 °C) with a nonzero water content. This represents ice granules colder than melting, but with thin water films already present on their surfaces.
- It is **not possible**, however, to assign water temperatures above 0 °C. By definition, the liquid phase is always fixed at the melting point.

When such a snowpack is entrained into the avalanche:

- The **ice component** enters the flow at the specified temperature (which may be below 0 °C).
- The **water component** is always assumed to be at 0 °C.

This distinction is important: entrainment can cool or warm the avalanche core, depending on whether the incoming snow is colder or warmer than the flow, but the entrained liquid fraction never exceeds 0 °C.

Density Specification

Another key input is snow density. The **three-day settlement option** is restricted to **new, dry snow** and should not be used for wet avalanche conditions.

- If a density above 200 kg/m³ is entered, RAMMS::Extended automatically resets the value to the “new snow” limit, preventing unrealistic compaction for fresh snow.
- For wet snow scenarios—where densities are often much higher—users must instead select the **Variable** or **Constant** options. These allow specification of denser, water-laden snowpacks typical of isothermal conditions.

Figure 8.1: Example of the input specification of a wet snowcover. In this case 1% water is defined at an elevation of 2000m. The height of the snowcover at 2000m is 1.20m, thus there will be 12.00mm of water in the snowcover. Because snow height gradients are defined, the amount of water content will vary with elevation. Typically, with wet snow avalanche, higher densities are applied, in the case ϱ_Σ=300kg/m3. We define the older snowcover using the Variable method. Also note that the splash parameter is set to 0.

Visualization in the GUI

The Snowcover Tab provides immediate feedback through **color bar visualizations** next to the water content field. These color bars show the spatial distribution of liquid water across the terrain, after applying the specified reference value and vertical gradient. They offer a quick check to ensure that inputs are realistic and consistent with the terrain and snowcover conditions.

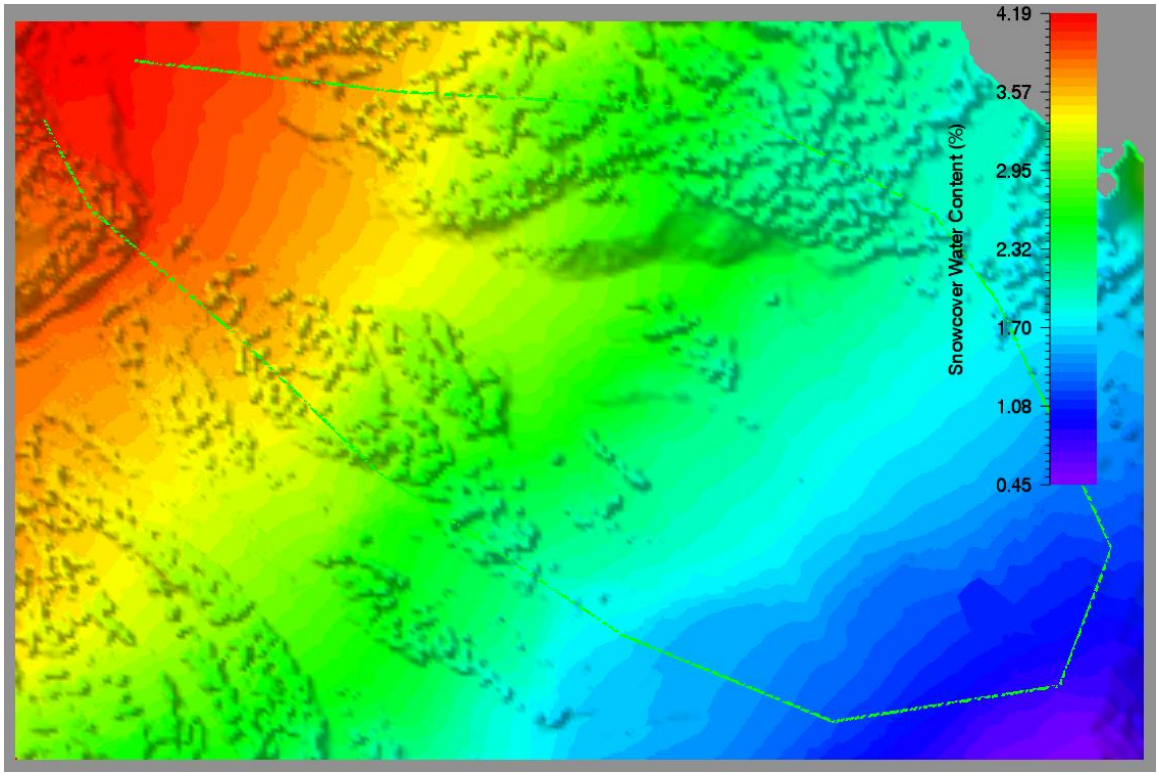


Figure 8.2: Snowcover water content in percent. In this example, we defined a 1% water content at 2000m (release zone) and a positive gradient of 0.5%/100m. This leads to approximately 4% water content in the runout zone. This depiction was made by clicking the colorbar next to the water content in the Snowcover input tab.

8.3 Air Temperature

The **Powder/Wet Tab** contains the input for ambient air temperature (T_Λ). At first glance this parameter may appear secondary, but it provides an important thermal boundary condition for the avalanche as it travels across large elevation gradients. The surrounding air can either **retard melting** (in cold conditions) or **accelerate melting** (in warm conditions).

- **Cold air (below freezing):** Suppresses additional meltwater generation at the surface of snow granules, prolonging the persistence of dry or transitional regimes.
- **Warm air (near or above freezing):** Promotes enhanced melting, particularly in the runout zones, encouraging the transition to wet, lubricated flow.

Unless direct measurements are available, it is common practice to set air temperature equal to the snowcover temperature ($T_\Lambda \approx T_\Sigma$), assuming a near-equilibrium between air and snowpack.

Convective Heat Exchange

RAMMS::Extended includes a convective heat-transfer formulation originally developed in chemical engineering during the heat exchange with spherical particles in turbulent flow (Ranz and Marshall, 1952). The air temperature T_Λ enters the model through the heat flux term:

$$q_{\Lambda\to\Phi} = H_\Phi A_{s,\pi} \, (T_\Phi - T_\Lambda)$$

where:

- T_Φ = avalanche core (or phase) temperature,
- T_Λ = ambient air temperature,
- $A_{s,\pi}$ = total surface area of snow granules exposed to air,
- H_Φ = convective heat transfer coefficient.

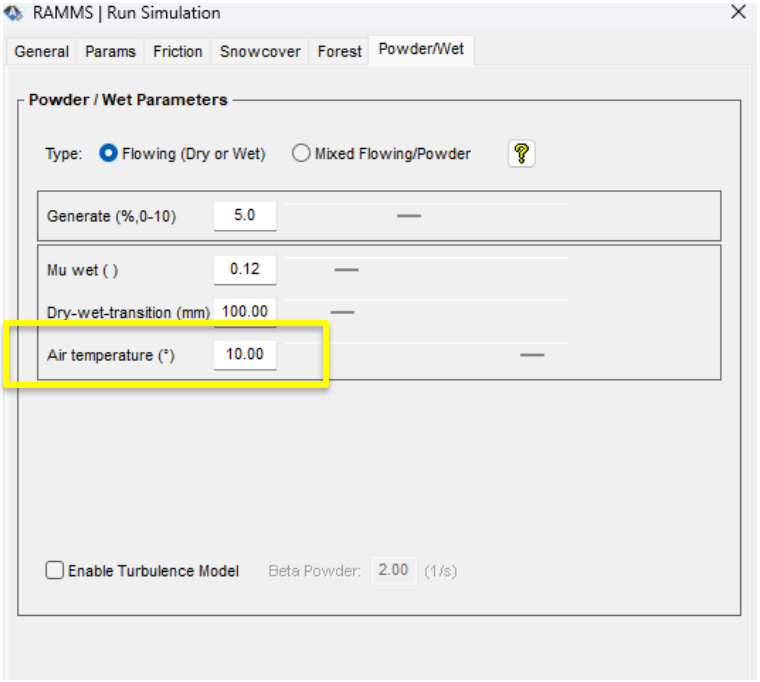


Figure 8.3: Input of the air temperature is found in the Powder/Wet tab. Temperatures can vary between -30C and +30C.

The heat transfer coefficient is obtained from standard dimensionless correlations:

$$H_\Phi = \frac{Nu \cdot k_a}{2r_\pi}$$

with

- k_a = thermal conductivity of air (≈ 0.0257 W·m^{-1}·K^{-1}),

- r_π = mean particle radius,
- Nu = Nusselt number, defined as:

$$Nu = 2 + 0.6Re^{\frac{1}{2}}Pr^{\frac{1}{3}}$$

Here:

- The Reynolds number depends on the speed of the avalanche:

$$Re = \frac{2\|\boldsymbol{u}_\Phi\| r_\pi}{\nu_a}$$

where:

  - $\|\boldsymbol{u}_\Phi\|$: flow velocity of the avalanche core (slope parallel),
  - ν_a: kinematic viscosity of air.

- The Prandtl number of air is:

$$Pr = \frac{c_a \nu_a}{k_a}$$

with c_a the specific heat capacity of air.

The total surface area exposed to air is computed assuming spherical snow granules:

$$n_p = \frac{3\,\hat{\rho}_\Phi \hat{h}_\Phi A}{4\rho_\pi \pi r_\pi^3}$$

with

- $\hat{\rho}_\Phi$ = co-volume density of the avalanche core,
- $\hat{h}_\Phi$ = co-volume height of phase p,
- A = computational cell area,
- ρ_π = material density of snow granules (300–500 kg/m³ typical).

Practical Implications

Because snow granules are small, their specific surface area is high, and even modest temperature differences with the atmosphere can have a measurable effect. Still, in most simulations the effect of air temperature is **secondary** compared to frictional heating and entrainment of warm snow.

- In **very cold air masses** (e.g., T_Λ < –10 °C), convective cooling can delay or even prevent wet transitions, leading to shorter runout distances than expected.
- In **warm air masses** (T_Λ > 0 °C), convective fluxes can enhance meltwater production at the particle surface, particularly in valley bottoms, promoting lubricated flow.

Thus, **while air temperature is not usually the dominant factor in wet avalanche dynamics**, it provides an important correction that can shift the timing and extent of dry–wet transitions, especially for avalanches spanning large elevation ranges.

8.4 Flow Friction in Wet Snow Avalanches

In **RAMMS::Extended**, flow resistance is modeled using the **Voellmy approach**, which combines a Coulomb sliding term and a velocity-squared (viscous/turbulent) term. For wet avalanches, both terms are modified dynamically as meltwater accumulates in the avalanche core.

The governing relationships are:

$$\mu' = \mu_w + (\mu_0 - \mu_w)e^{\left(-\frac{m_\Phi}{m_t}\right)}$$

$$\xi' = \xi_w + (\xi_0 - \xi_w)e^{\left(-\frac{m_\Phi}{m_t}\right)}$$

where:

- μ_0, ξ_0 = baseline dry friction coefficients,
- μ_w, ξ_w = limiting wet friction coefficients,
- m_t = transition parameter controlling the sensitivity to water content,
- m_Φ = meltwater content in the avalanche core.

These equations ensure that:

- At **zero water content**, friction values remain at their dry limits (μ_0, ξ_0).
- As water content increases, both μ_0 and ξ_0 decay exponentially toward their wet limits (μ_w, ξ_w).

Physically, m_t(m) represents the **characteristic thickness of the water layer on snow granules required to cause a significant reduction in Coulomb friction and a shift toward plug-like motion**. Small values of m_t (1–10 mm) describe situations where even thin water films strongly influence mobility, as in transitional capillary regimes. Larger values (0.1–1.0

m) represent saturated conditions, where much more water is needed before friction values approach their wet limits.

The transition from dry to wet behavior is governed initially by **capillary surface processes**. Thin water films form on granule surfaces, reducing basal sliding resistance while at the same time binding grains into aggregates. These agglomerates stiffen the flow, inhibit particle rearrangement, and increase velocity-dependent resistance. This is the **transitional or capillary regime**, in which avalanches move more slowly than their dry counterparts but can still travel long distances due to basal lubrication.

As water content rises further, a threshold is reached where the pore space of the snowpack becomes filled with liquid. At this point, **pore-water effects dominate**. The snow grains lose their cohesive strength, bonds disintegrate, and shear resistance collapses. The mass behaves more like a water-saturated granular slurry: fluidization energy can no longer be generated, plug-like structures disintegrate, and mobility becomes controlled primarily by lubrication at the base. This is the **fully saturated regime**, characterized by very low Coulomb friction and minimal internal resistance.

In **RAMMS::Extended**, this rheology is captured by dynamically reducing μ and ξ toward wet limit values as water content increases. For the Coulomb term, typical wet values are μ_w =0.00–0.15. For the turbulent term, **RAMMS::Extended** adopts the Swiss standard wet-snow value ξ_w=800 m/s^2.

The combined effect of lubrication and aggregation explains the **paradoxical mobility** of wet avalanches: in the capillary regime, **lubrication reduces sliding resistance,** but **aggregation raises internal stiffness**, while in the pore-water regime, basal resistance collapses entirely and the flow transitions to a water-dominated slurry. In extreme cases, particularly under fully saturated conditions, **RAMMS::Avalanche** (without fluidization) may provide a more realistic description of the flow.

Practical Guidance for Users

When configuring wet avalanche friction in the **Friction Tab** and **Powder/Wet Tab**:

- Start with baseline dry values (μ_0, ξ_0) appropriate for the avalanche return period (Chapter 3).
- Select wet limit values $\mu_w \in$ (0.00–0.15). Presently ξ_w= 800 m/s² is automatically set within **RAMMS::Extended**. Because RAMMS::Extended uses an exponential relationship for the sliding friction, the value μ_w =0.0 is never reached but is approached in the limit as the water content increases.
- Choose a transition parameter mt depending on whether you expect a thin water film regime (1–10 mm) or a fully saturated regime (0.1–1.0 m).

- If modelling highly lubricated block flows, set N_0 close to zero; for cohesive plug-like flows, use N_0 around 500 Pa.
- Always check that selected values are consistent with the specified snowcover wetness and air/snow temperatures.

8.5 Generate Parameter (α_Φ) and Random Energy Decay (β_Φ)

A central feature of RAMMS::Extended is its treatment of **random kinetic energy** (also called fluidization energy). This energy describes the random motions of snow granules that allow avalanches to fluidize, disperse, and generate powder clouds. In wet avalanches, both the **production** and **decay** of this energy are strongly affected by the presence of liquid water.

The **Generate parameter (α_Φ)** controls the rate at which random energy is produced by shearing and particle collisions. In dry avalanches, collisions are elastic enough to sustain a significant level of random motion, and typical values of α_Φ are in the range 0.07–0.08 (7–8%). In wet avalanches, however, capillary forces and liquid bridges between grains cause collisions to become inelastic and sticky. Particles form aggregates rather than dispersing, and very little random motion is created. For this reason, α_Φ must be reduced to values of 0.03–0.05 (3–5%) when modeling wet snow avalanches.

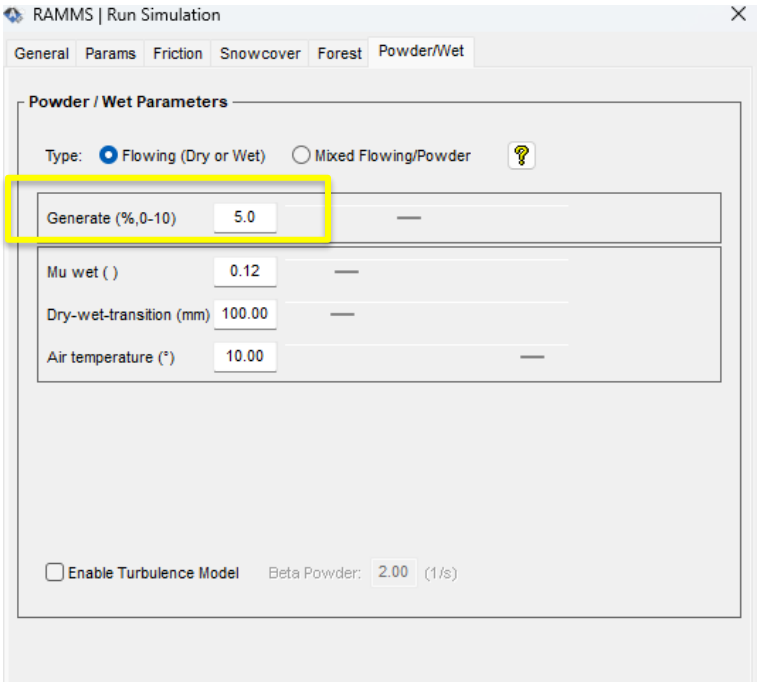


Figure 8.4: Generate α_Φ is specified in the powder/wet tab. The decay parameter β_Φ is automatically defined by temperature.

At the same time, the **Random Energy Decay coefficient (β_Φ)** determines how quickly this random motion is dissipated. In cold, dry flows, random energy persists for relatively long times, with $\beta_\Phi \approx 1\ s^{-1}$. In wet avalanches, however, every collision between grains drains energy as liquid bridges rupture and reform. This dramatically increases dissipation, and β_Φ rises to about $2\ s^{-1}$. As a result, even the small amount of random energy generated in wet avalanches decays rapidly, so fluidization never dominates. (In a way this is the very definition of a wet snow avalanche: no fluidization.)

The dependence of β_Φ on temperature is especially important. As the avalanche core warms toward isothermal conditions, capillary bonding becomes more effective, and the rate of energy decay accelerates sharply. In **RAMMS::Extended**, β_Φ is formulated as a function of avalanche temperature (T_Φ), doubling over the narrow range where snow transitions from cold and dry to warm and wet. The figure below illustrates how β_Φ increases with both temperature and water content.

The combination of reduced α_Φ and increased $\beta_\Phi$captures one of the defining characteristics of wet avalanches: they lose random kinetic energy faster than they can produce it. The consequence is a dense, sluggish flow with little or no powder cloud, rapid deposition at the avalanche tail, and runout behavior dominated by basal lubrication rather than by dispersion.

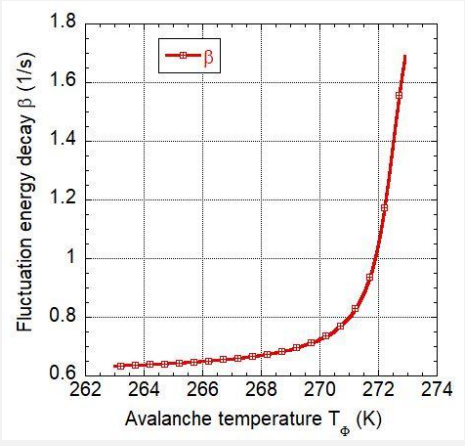
Parameter	Definition/GUI	Range
β_Φ	**Decay**  $\beta_\Phi = 1.40 + \frac{1.7}{\pi}\ atan[1.6(T_\Phi - 272.5)]$	$\beta_\Phi \approx 1/s$ (Dry, powder) $\beta_\Phi \approx 2/s$ (Wet snow) $0.5 \leq \beta_\Phi \leq 2.0$ (Range)

Figure 8.5: The decay parameter increases rapidly after the avalanche reaches T_Φ=-3C.

For practical modeling:

- Use α_{Φ} **= 0.07–0.08** for dry avalanches and α_{Φ} **= 0.03–0.05** for wet avalanches.
- The decay parameter is $\beta_{\Phi} \approx 1\ s^{-1}$ for dry avalanches and $\beta_{\Phi} \approx 2\ s^{-1}$ for wet avalanches, with temperature dependence defined as in the **RAMMS::Extended** formulation (see Figure 8.5).

8.6 Entrainment in Wet Snow Avalanches

Overview

The entrainment routines in **RAMMS::Extended** are based on the elastic/plastic collision framework described in Chapter 4. Entrainment not only **increases avalanche mass** but also **modifies internal energy fluxes** through the partitioning of heat and random kinetic energy (ε) and through changes in avalanche temperature. For powder avalanches, entrainment at the front produces substantial fluidization energy, enabling particle dispersion and cloud formation.

In wet snow avalanches, however, **the entrainment process operates fundamentally differently**. Wet snow covers are **dense, warm, and often isothermal** (ρ ≥ 200–300 kg/m³). Their interaction with the avalanche produces less fluctuation energy, immediate temperature equilibration, and stronger dissipative effects.

Key Differences to Dry/Powder Entrainment

1. **Dense, warm snow**
   Wet snow is heavier and more compact than fresh snow. Its erosion therefore requires more energy, while the accelerated mass is more dissipative. **This slows the avalanche relative to powder entrainment**.

2. **Immediate mixing and temperature balance**
   Entrainment of wet snow is modeled as **immediate mixing into the avalanche core**. Both snow mass and liquid water are instantly redistributed, causing a rapid adjustment of core temperature. This can accelerate the transition toward a wet flow regime.

3. **Suppressed fluctuation energy (ε)**
   The production of **random kinetic energy during entrainment is strongly damped in warm, wet snow**. Instead of dispersive fronts and powder clouds, the avalanche tends toward dense, plug-like flow. In practice, ε is automatically reduced as a function of snow temperature, meaning wet entrainment directly decreases dispersive capacity.

4. **Thermal feedback**
   Entrainment of isothermal, near-melting snow introduces both mass **and heat**. This increases avalanche core temperature, promoting meltwater generation and further reducing fluidization. The thermal feedback is therefore much stronger than in cold snow entrainment.

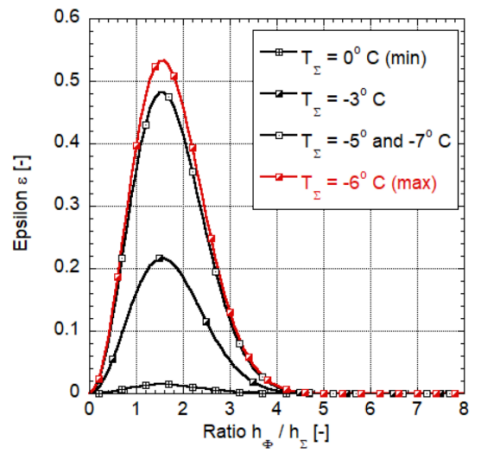
Parameter	Definition/GUI	Range
ε_Φ	**Partitioning Entrainment** $\varepsilon_\Phi = f(T_\Sigma)g(r) = \left[e^{\frac{-(T_\Sigma-267)^2}{10}}\right]\left[\frac{r^2 e^{\left(\frac{-r^2}{2a^2}\right)}}{a^3}\right]$ $r = h_\Phi / h_\Sigma$ $a = 1.1$ 	$0 \leq \varepsilon_\Phi \leq 0.5$ (Range) See Chapter 4

Figure 8.6: The entrainment partitioning factor ε_Φ as a function of temperature and impact geometry. Entrainment of wet snow is highly dissipative, hindering the formation of powder clouds.

Parameterization in RAMMS::Extended

To capture these behaviors, users should adjust the snowcover input fields in the **Snowcover Tab** as follows:

- **Release and erosion densities:**
  Use the *Variable* or *Constant Density* options (not the 3-day settlement method).
  Densities ≥ 300 kg/m³ are recommended for wet, compact snowcovers.

- **Splash (γ):**
  Set *Splash* = 0. Wet snow behaves plastically; no splashed front develops.

- **Yield stress (τ):**
  Increase from the default 300 Pa to 500–1000 Pa. This limits entrainment to locations where the avalanche has sufficient mass and velocity and suppresses lateral flow arms.

- **Partitioning parameter (ε):**
  The parameter ε is **automatically reduced by snow temperature and avalanche geometry (**Figure 8.6**)**. In wet snow conditions, $\varepsilon \rightarrow 0$, reflecting the near absence of fluidization energy.

- **Decay parameter (β_Φ):**
  Wet snow entrainment accelerates the decay of random kinetic energy. Typical values are $\beta_\Phi \approx 2\ s^{-1}$ (vs. $\approx 1\ s^{-1}$ for dry snow).

Practical Recommendations

- Use **high-density snowcovers** ($\rho \geq 300$ kg/m³) when simulating entrainment into old, wet snow.
- **Suppress splash** and **raise yield stress** to mimic the strongly plastic and dissipative character of wet snow entrainment.
- Be aware that entrainment in warm conditions not only increases avalanche mass but **also *heats* the core**, which can trigger or reinforce a regime transition from fluidized to wet flow.
- Remember that entrainment of wet snow inherently reduces ε and accelerates β_Φ, producing avalanches that are dense, slower, and lacking in powder clouds.

8.7 Temperature and Melting

Role in regime transition

Liquid water is the main driver of the **dry-to-wet transition**. Small amounts form thin films at grain contacts, which reduce Coulomb friction, raise viscous resistance, damp dispersion, and encourage plug-like motion. As water content grows, basal resistance drops and wetting accelerates, pushing the flow fully into the wet regime.

In **RAMMS::Extended**, the core's thermal state is tracked by the internal heat energy E_Φ (temperature is its scale). E_Φ is advected with the core and gains energy from shear work, entrainment, and the decay of random kinetic energy; it also exchanges heat with the powder cloud. Melting adds one more term: a latent-heat sink. This term links heat input to

the local meltwater production rate—and that rate is what ultimately governs the evolving rheology.

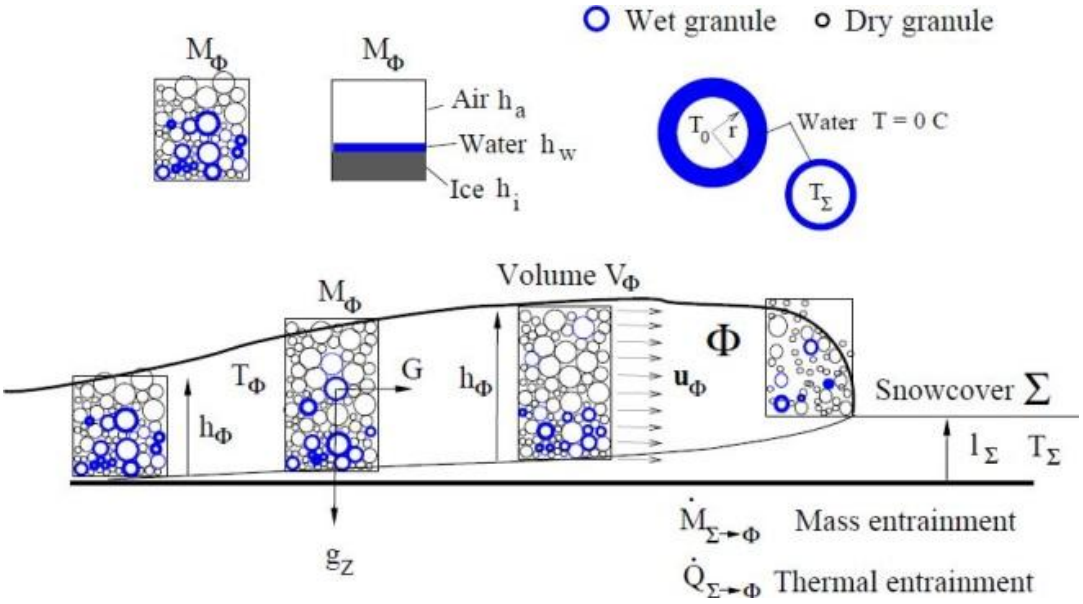


Figure 8.7: The "temperature" of an avalanche is hard to define. Because frictional/collisional heating raises granule surface temperatures before the mean, RAMMS::Extended starts melting slightly below the physical melting point by using a model constant T_m=−1℃.

The internal heat energy balance for the avalanche core is

$$\left(\hat{h}_\Phi E_\Phi\right)_t + div\left(\hat{h}_\Phi E_\Phi \vec{u}_\Phi\right) = \dot{Q}_{in} - \dot{Q}_{out}$$

with the thermal input

$$\dot{Q}_{in} = [1-\alpha_\Phi]\dot{W}_\Phi + \rho_\Sigma[1-\varepsilon_\Phi]\dot{L}_{\Sigma\to\Phi} + \beta_\Phi \hat{h}_\Phi R_\Phi + \varrho_\Sigma c_\Sigma T_\Sigma \dot{M}_{\Sigma\to\Phi}$$

and the loss of internal heat energy to the cloud

$$\dot{Q}_{out} = \dot{M}_{\Phi\to\Pi} E_\Phi .$$

Melting is introduced by adding a latent-heat sink to the right-hand side and a consistent moisture equation. Let m_Φ be the **meltwater thickness** (m) in the core. Then the transport of meltwater is

$$(m_\Phi)_t + div(m_\Phi \vec{u}_\Phi) = \dot{S}_m + \frac{\varrho_w}{\hat{\rho}_\Phi}\phi_{\Sigma w}\dot{M}_{\Sigma\to\Phi}$$

where $\dot{S}_m$ is the **local melt production rate**. The internal-energy equation becomes

$$\left(\hat{h}_\Phi E_\Phi\right)_t + div\left(\hat{h}_\Phi E_\Phi \vec{u}_\Phi\right) = \dot{Q}_{in} - \varrho_w L\dot{S}_m - \dot{Q}_{out}$$

The **latent-heat sink** is $-\varrho_w L\dot{S}_m$. Two options are provided to specify $\dot{S}_m$.

Ad-hoc method (empirical onset at T_m=−1℃)

Motivation. Because frictional/collisional heating raises **granule surface** temperatures before the mean, **RAMMS::Extended** starts melting slightly **below** the physical melting point by using a model constant $\boldsymbol{T_m}$**=−1**℃.

We apply a complementarity condition on the **mean** core temperature T_Φ:

$$T_\Phi \le T_m, \qquad \dot{S}_m \ge 0, \qquad [T_\Phi - T_m]\dot{S}_m = 0$$

with the "excess" enthalpy production $\dot{Q}_m$ in the core

$$\int_{\Delta t} \dot{Q}_m dt = \varrho_\Phi c_\Phi h_\Phi [T_\Phi - T_m]$$

being used to define $\dot{S}_m = \dot{Q}_m / \varrho_w L$. While $T_\Phi < T_m$, melting is off ($\dot{S}_m$=0). When $T_\Phi \ge T_m$, the temperature is **clamped to** $\boldsymbol{T_m}$ any additional, excess heat energy is diverted to melting.

Fourier-based surface heating

A new method addresses this limitation by explicitly modeling surface and core temperatures of snow granules using a Fourier-based parabolic profile.

Key ideas:

- Heat penetrates into granules according to the Fourier number $Fo(t) = \alpha t / {r_\pi}^2$ where r_π is grain radius and α thermal diffusivity.
- The internal temperature distribution is approximated by a parabolic profile with coefficients $A(Fo)$ and $B(Fo)$, constrained to conserve total energy.
- This ensures that:
    - Surface temperatures rise faster than the mean (reflecting intense shearing and friction at boundaries).
    - Core temperatures lag behind, mimicking thermal diffusion.
- Melting occurs when the surface temperature reaches T_m=0 °C, not the mean. Excess energy is then diverted into melting a surface shell of thickness, from which meltwater volume is computed.

This approach is more physical because it explicitly accounts for thermal heterogeneity and avoids relying on an artificial T_m=–1 °C threshold.

8.8 Case Studies

8.8.1 VdIS 25.02.1999

The avalanche of 25 February 1999 (estimated volume: V_0 = 316,000 m³) is an outstanding example of a **dry–wet transitional avalanche**. Information about this avalanche is documented in the 1999 VdIS report.

In the week preceding the event, between 16 and 23 February, approximately 190 cm of new snow accumulated in the VdIS release zone. During this period, air temperatures at 2,400 m (the approximate elevation of the starting zones) rose to about –4 °C between 18 and 22 February. This warming phase densified the snowpack considerably, with mean release zone densities of roughly 300 kg/m³ (range: 260–340 kg/m³, measured). It was followed by a colder period, with air temperatures dropping to –15 °C at 2,400 m. At the time of release, the snowpack can therefore be described as relatively warm in its interior (around –5 °C in the release zone), overlain by colder near-surface layers (around –10 °C).

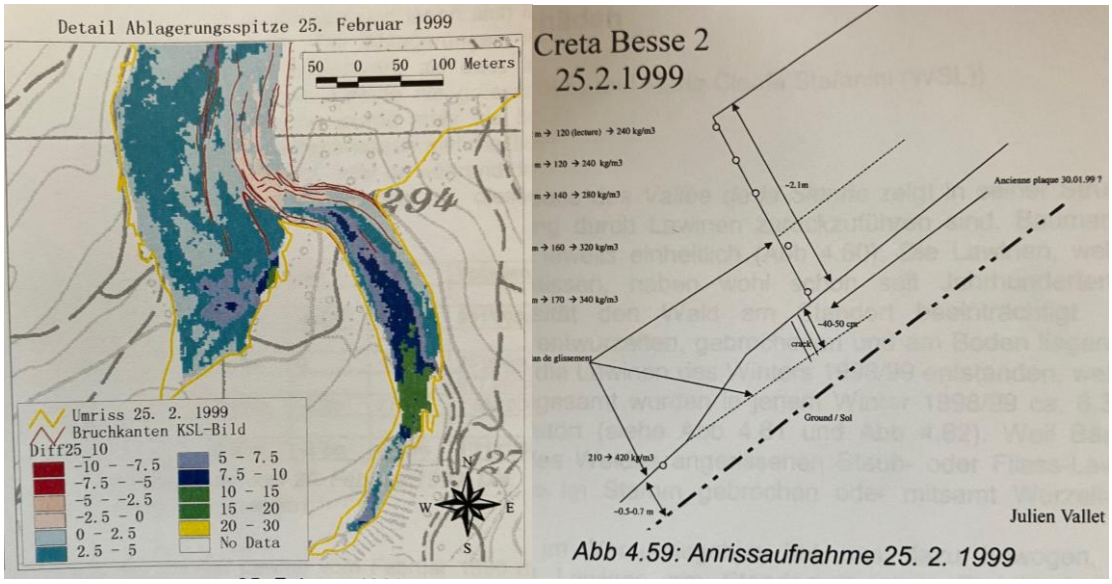


Figure 8.8: Measured depositions at the end of the 25.02.1999 VdIS avalanche in the region of Seillon (left). Measured snow densities near the fracture crown of the avalanche (right). Depth below surface → estimated density: 0.2 m → ≈ 260 kg m^{-3}; 0.5 m → ≈ 260–280 kg m^{-3}; 0.9 m → ≈ 280–300 kg m^{-3}; 1.4 m → ≈ 320 kg m^{-3}: 2.0 m → ≈ 340 kg m^{-3}; At/near the sliding plane ("Plan de glissement") → a note indicates ≈ 420 kg m^{-3}. Both documents are taken from the 1999 VdIS report.

The avalanche displayed a combination of **dry-snow behavior** and **wet-snow dynamics**. Its powder cloud rose dramatically during descent, destroying large sections of forest on the opposite slope of Berguges, while its wet-snow runout left massive depositions stretching the entire length of the Berguges valley down to 1,300 m. The avalanche front continued more than one kilometer beyond the VdIS bunker, following the course of the

Sionne river. A further portion of the flow advanced into the Seillon area, where deposition depths reached up to 5 m. These extensive deposits are particularly valuable for constraining **wet-snow transitional parameters**, including the effective friction coefficient (μ_w) and the transitional scaling length (m_t).

To simulate the VdlS avalanche of 25 February 1999, we relied extensively on information from the official avalanche report, including the mapped extent of the release zone. The release area exhibited highly variable crown heights, ranging from 0.5 m to 3.0 m (Table 4.5). For the simulation, we adopted the mean crown height of 1.5 m (report), yielding a release volume of approximately 315,800 m³ (in close agreement with the reported value of 316,000 m³).

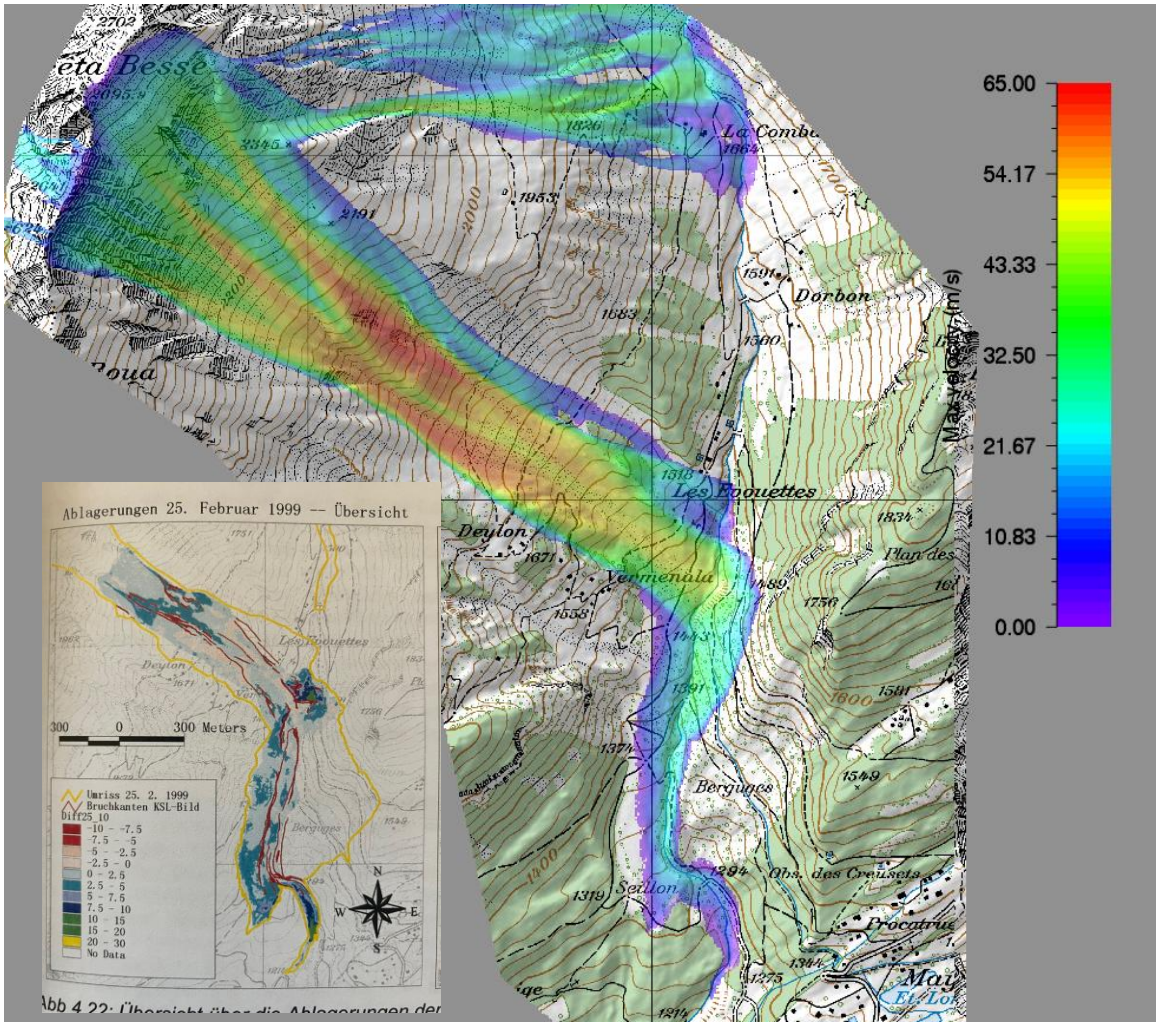


Figure 8.9: Simulated maximum flow velocity and runout of the 25.02.1999 VdlS avalanche. The avalanche reaches a peak velocity of 60-65m/s. Reported mean values are 63m/s. Note the long simulated and observed runout to the region of Seilon.

The snowpack in the release zone (at ~2600 m) was specified as relatively warm, with an assumed temperature of ($T_0 \approx T_\Sigma$ =–5 °C. This reflects the influence of the preceding

warming phase (with air temperatures up to T_Λ = −4 °C), which likely increased internal snow temperatures, while the surface layers remained colder. The snow was treated as dry (no liquid water present), and we applied a generation parameter of α_Φ = 7%, consistent with dry-snow avalanche behaviour. Curvature was set to ON.

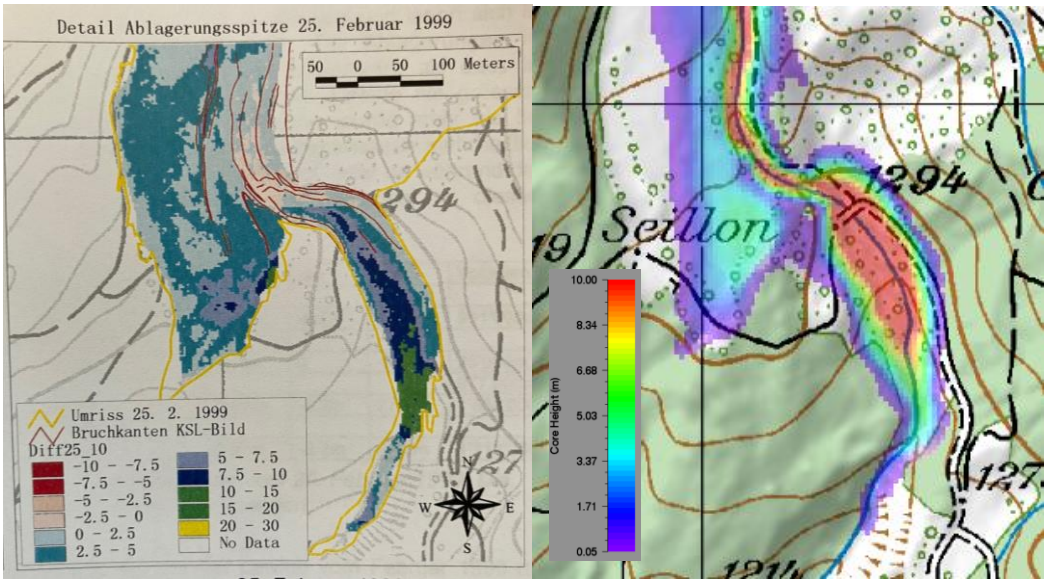


Figure 8.10: Measured and calculated runout VdIS avalanche 25.02.1999. Calculated deposition heights in the left arm were over 5m (observed) and in the Sionne torrent over 10m (observed). The long runout could only be reached with transitional wet snow parameters of μ_w = 0.15 and m_t =2mm.

A deep snow cover of h_Σ = 150 cm was prescribed, with a bulk density of ϱ_Σ = 300 kg/m³. This density, slightly lower than the reported 190 cm snowfall accumulation, accounts for settlement effects. Standard vertical gradients were imposed for both snow depth (Δh_Σ = 30 cm per 100 m) and snow temperature (ΔT_Σ = 0.5 °C per 100 m). The **variable** snow density option was selected, representing a heavily settled, warm snowcover. The snow entrainment conditions led to a calculated avalanche with an end volume of 1 mio m³ (measured values are estimated to be 980'000m³).

The avalanche was simulated on a 5m×5m computational grid, parameterized as a **100Y large avalanche (100L)** scenario. The calculated avalanche reached peak velocities of 65m/s, again in agreement with both photogrammetric analysis of front locations (report).

To simulate the wet snow avalanche runout, we selected μ_w = 0.15 (wet sliding friction) and a small transition value m_t =2mm. The small transition corresponds to a large avalanche (large flow heights) running over a compacted snowcover. In this case the snowcover was cold, suggesting capillary effects raised the coefficient of sliding friction to μ_w = 0.15. In the next example (Rigopianno) of a large avalanche running over a warm, perhaps slightly moist snowcover, the coefficient sliding friction decreases to μ_w = 0.00-0.05.

In this case study, **air temperature** proved to be a key factor in controlling the dry–wet transition. When colder values of T_Λ =−8°C were prescribed, runout distances were underestimated, as melting and liquid water production were largely suppressed. Realistic runout behavior required warmer air temperatures of about T_Λ =−5 to −6°C, which allowed the formation of a wetting layer and a corresponding reduction in basal friction through increased water content.

Case Study: Dry-Wet Transitional Snow Avalanche			
Case study name	VdIS 200 25.02.1999	Grid resolution (m)	5mx5m
Avalanche size / Return period	100L	Forest present (yes/no)	Yes
Starting volume (m³)	315’800	End volume (m³)	1 mio
Snowcover Information			
Reference altitude (m a.s.l.)	2600	Snow height gradient Δh_Σ (m/100 m)	0.03
Snow temperature T_Σ (°C)	-5	Snow temp. gradient ΔT_Σ (°C/100 m)	0.5
Snow water content m_Σ (%)	0	Water content gradient Δm_Σ (%/100 m)	0
Erosion depth h_Σ or $d_0{}^*$ (m)	1.50	Density method (3-day/variable/constant)	**Variable**
Erosion density ϱ_Σ (kg/m³)	300	Air temperature T_Λ (°C)	-6
Release density ϱ_0 (kg/m³)	300	Release depth d_0 (m)	1.50
Powder Parameters			
Generate α_Φ (%)	7	Cloud drag (–)	4
Cliff factor (–)	2	Air entrainment ψ (m/s scale)	2
Turbulence decayβ_Π (s^{-1})	2.00		
Wet Snow Friction			
μ_w(wet sliding friction) (–)	0.15	m_t (transition thickness, mm)	2
Comments			
A deep snow cover of h_Σ=150cm with a bulk density of ϱ_Σ=300 kg/m³ was prescribed, slightly lower than the reported 190 cm snowfall to account for settlement. Standard vertical gradients were applied for snow depth (Δh_Σ =30cm/100 m) and snow temperature (\ΔT_Σ= 0.5C/100 m). Under these entrainment conditions, the avalanche reached a simulated end volume of 1.0 million m³, closely matching the measured estimate of 980,000 m³. The event was modeled on a 5 × 5 m computational grid as a **100Y large avalanche (100L)** scenario. To reproduce the wet-snow runout, we selected a sliding friction coefficient of μ_w=0.15 together with a small transition length m_t=2 mm, representing a large avalanche with high flow depths traveling over a compacted, cold **snow cover where capillary effects likely increased basal friction.**			

8.8.2 Rigopiano 18.01.2017

On 18 January 2017, after days of heavy snowfall and a seismic swarm, a massive avalanche swept down the eastern Gran Sasso, engulfing the Hotel Rigopiano in Farindola (Abruzzo, Italy). Twenty-nine guests and staff were killed; eleven people were rescued alive after a difficult, days-long search hindered by blocked roads and communication failures. The disaster highlighted vulnerabilities in hazard zoning and emergency coordination during compound events where weather and earthquakes interact.

Case Study: Dry-Wet Transitional Snow Avalanche			
Case study name	Rigopiano 18.01.2017	Grid resolution (m)	2mx2m / 5mx5m
Avalanche size / Return period	300L	Forest present (yes/no)	Yes
Starting volume (m³)	77'150	End volume (m³)	135'000
Snowcover Information			
Reference altitude (m a.s.l.)	1500	Snow height gradient Δh_Σ (m/100 m)	0.05
Snow temperature T_Σ (°C)	-2	Snow temp. gradient ΔT_Σ (°C/100 m)	0.1
Snow water content m_Σ (%)	0	Water content gradient Δm_Σ (%/100 m)	0
Erosion depth h_Σ or $d_0{}^*$ (m)	1.50	Density method (3-day/variable/constant)	**Variable**
Erosion density ϱ_Σ (kg/m³)	250	Air temperature T_Λ (°C)	-2
Release density ϱ_0 (kg/m³)	250	Release depth d_0 (m)	2.0
Powder Parameters			
Generate α_Φ (%)	7	Cloud drag (–)	3
Cliff factor (–)	2	Air entrainment ψ (m/s scale)	2
Turbulence decayβ_Π (s⁻¹)	2.00		
Wet Snow Friction			
μ_w(wet sliding friction) (–)	0.00-0.05	m_t (transition thickness, mm)	2
Comments			
Rigopiano starts as a dry avalanche (no water) at an elevation of 1500m, in warm temperatures $T_\Sigma = T_\Lambda$ =-2°C. No water in release zone, generate $\alpha_\Phi = 7\%$. Release zone depth extreme (2m, ϱ_0 =250 kg/m³, consistent with warm snow), therefore 300L parameters for flow friction. Erosion parameters given by **snow profile measurements in the runout zone** h_Σ =1.5m/ϱ_Σ = 250 kg/m³, dry. After approximately 30s avalanche begins to generate meltwater. Meltwater water lubrication is responsible for long, sliding runout in deposition zone m_t =2mm. **Wet sliding friction set to μ_w=0.00 - 0.05 to model sliding friction on a compacted snowcover.**			

The January 2017 Rigopiano avalanche is a clear example of a warm, initially dry flow that transitions to a wet regime. Snowpack measurements indicated a deep ($h_\Sigma \gtrsim$1.5m), dense

(ϱ_Σ = 200-250kg/m³), and warm snowcover ($T_0 \approx T_\Sigma$ =-2C). A thick release slab (≈ 76'700 m³) initiated high on the slope and funneled into a narrowing gully; for the simulation we prescribed h_Σ =1.50m with ϱ_Σ = 250kg/m³, consistent with observations.

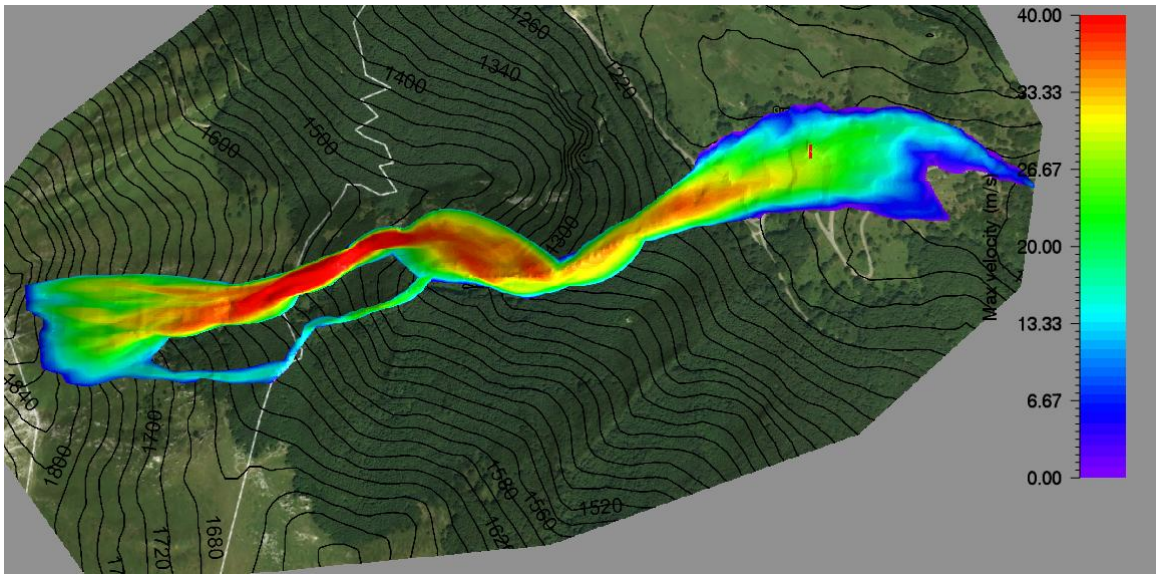


Figure 8.11: Calculated max velocity of the Rigopiano avalanche. The avalanche reached a maximum velocity of 45m/s. The avalanche struck the hotel with a speed of 25 m/s and exerted pressures of 290kPa (Wall), 175kPa (Voellmy) and/or 220kPa (Work-Energy). The long extend finger in the runout zone is due to lubricated motion at the end of the event.

As the confined corridor twisted, it opened onto a forested slope that was completely swept and entrained, adding mass and further warming the core. The avalanche reached ≈ 45 m/s before the gully and maintained this speed through the bends—consistent with weak curvature effects (curvature term disabled in the run). Grain melting began almost immediately, prior to the winding torrent, and continued downslope; the model predicts a ~2 mm surface water film on grains in the deposition zone, explaining the long, lubricated runout and finger-like elongation documented in the field.

At the hotel site, peak impact pressures were estimated between 175 and 290 kPa, depending on method: 290 kPa from the **standard wall formula** (C_d = 2), 175 kPa using the **Voellmy approach**, and 220 kPa from a **work–energy balance**. Simulations with colder-snow settings could not reproduce the observed runout and distinctive, fingered deposition; only warm-snow entrainment matched the event, emphasizing temperature control on flow behavior. To reflect the absence of meltwater in the release area we set Generate to α = 7%. A further distinctive requirement was an exceptionally low wet-friction value (μ_w = 0.00 -0.05) together with a small melt-transition thickness (m_t = 2 mm), indicating that even very thin water films can trigger the dry→wet regime shift.

Summary: Large warm ($T_0 \approx T_\Sigma$ =-2°C) avalanches running a moist, compacted snowcover can have low values of wet friction $\mu_w \approx$ 0.00-0.05 and small transition

values $m_t \approx$ 2mm. The low μ_w appears to be determined by the warm snow in the runout zone.

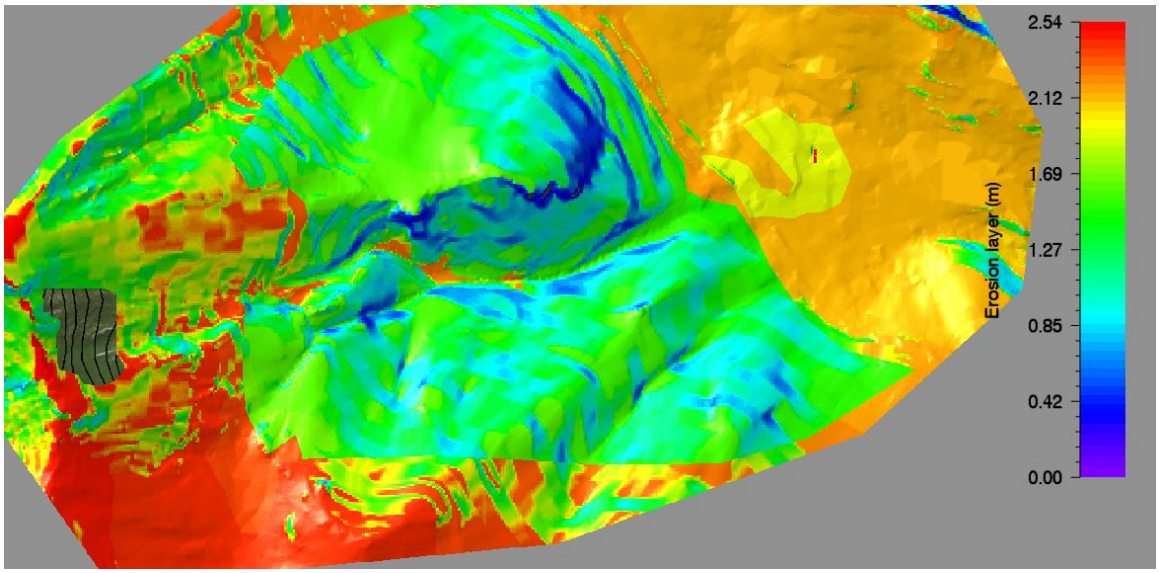


Figure 8.12: A deep snowcover was specified with the variable option. Forest interception was used in the forested areas surrounding the torrent.

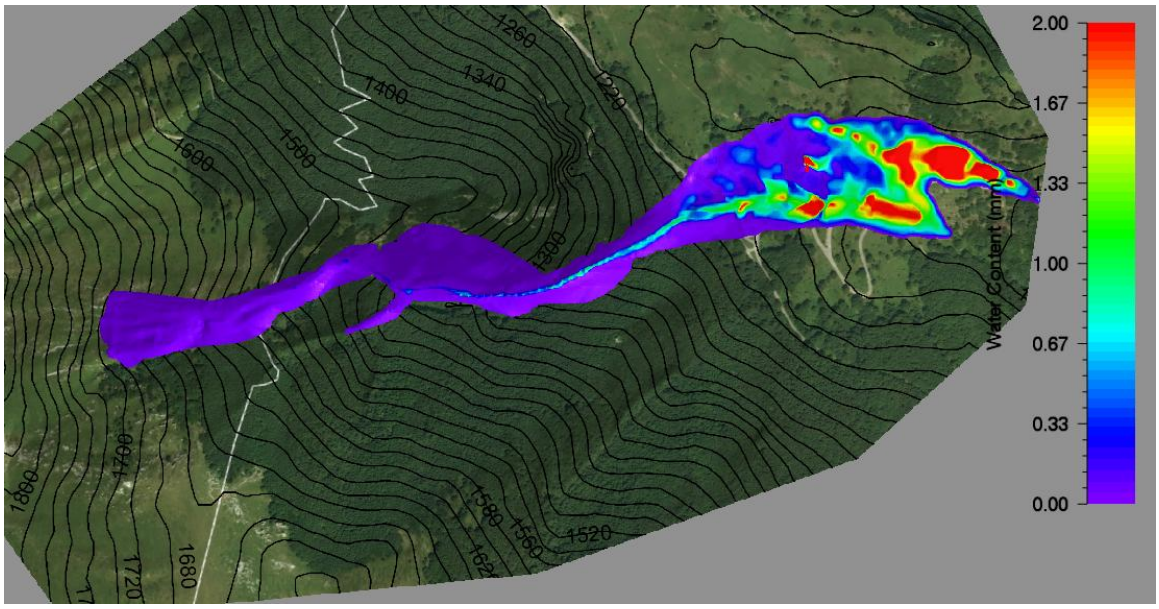


Figure 8.13: Calculated meltwater in the avalanche deposits. The surfaces of the granules were covered with a fine thin surface of meltwater.

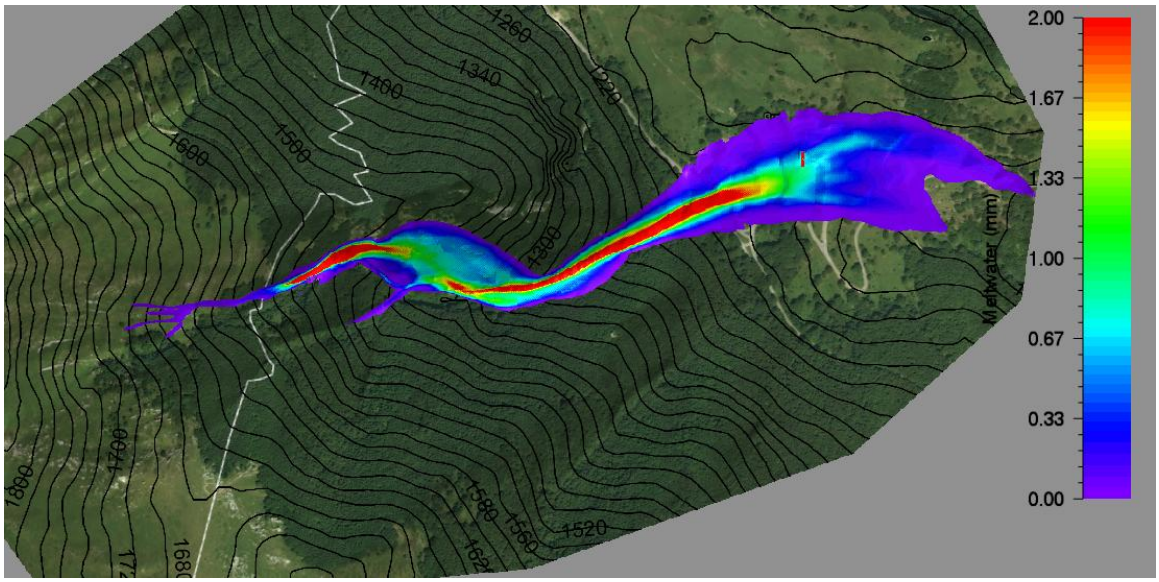


Figure 8.14: Calculated meltwater production. Due to the warm temperatures of the snow, meltwater was produced almost immediately after release.

8.8.3 Grüenbödeli 19.12.2023

Grüenbödeli is a relatively small avalanche track located between Davos and Klosters. Owing to its modest elevation range—starting at approximately 2,000 m and running out at around 1,400 m—the site is particularly susceptible to small, wet-snow avalanches. These events often scour the snowpack down to the ground surface, producing classic **Grundlawinen** (ground avalanches).

The site is readily accessible from Davos, making it especially well-suited for detailed field investigations, including drone-based surveys of both release and deposition zones.

During December 2023, several small avalanches occurred at Grüenbödeli. The largest, with an estimated volume of roughly 2,500 m³, descended all the way to the valley floor. A drone survey successfully mapped the approximate release area as well as the runout and deposition zones (see Figure).

For modeling purposes, this avalanche was treated as a warm ($\boldsymbol{T_0} \approx \boldsymbol{T_\Sigma} \approx 0$ °C) ground avalanche, releasing from a moist snowpack with an estimated liquid water content of about 1%. Because the avalanche began with moist snow we set generate to α_Φ = 5%. Release and entrainment densities were taken to be 300kg/m³. We used a **variable** density distribution method (no three-day snow); **curvature ON**, due to the compact flow core. The depth of snowcover we took as 40cm-50cm with a standard elevation gradient of 3cm/100m. The temperature gradient was weak: 0.2C/100m. We modelled the avalanche using a low wet friction $\mu_w \approx 0.12 - 0.15$ and a transition parameter roughly on the scale of a single granule diameter $m_t \approx$ 50mm – 100mm. We considered the avalanche to be a 10T (10 year, tiny). This input reproduced the observed runout, flow width and deposition pattern. The avalanche was modelled without the powder cloud.

Summary: Small wet avalanches running on the ground require higher friction values $\mu_w \approx 0.12 – 0.15$ and higher transition values $m_t \approx$ 50mm – 100mm. The transition values are on the scale with the granule size. Higher wet snow densities are required to simulate the event.

Case Study: Small, wet ground avalanche			
Case study name	Grüenbödeli	Grid resolution (m)	1mx1m / 2mx2m
Avalanche size / Return period	10T	Forest present (yes/no)	No
Starting volume (m³)	2'525	End volume (m³)	4'600
Snowcover Information			
Reference altitude (m a.s.l.)	2000	Snow height gradient Δh_Σ (m/100 m)	0.03
Snow temperature T_Σ (°C)	0	Snow temp. gradient ΔT_Σ (°C/100 m)	0.2
Snow water content m_Σ (%)	1	Water content gradient Δm_Σ (%/100 m)	0.2
Erosion depth h_Σ or $d_0{}^*$ (m)	0.40	Density method (3-day/variable/constant)	**Variable**
Erosion density ϱ_Σ (kg/m³)	300	Air temperature T_Λ (°C)	0
Release density ϱ_0 (kg/m³)	300	Release depth d_0 (m)	1.0
Powder Parameters			
Generate α_Φ (%)	5	Cloud drag (–)	No powder
Cliff factor (–)	No powder	Air entrainment ψ (m/s scale)	No powder
Turbulence decayβ_Π (s^{-1})	No powder		
Wet Snow Friction			
μ_w(wet sliding friction) (–)	0.15	m_t (transition thickness, mm)	50
Comments			
Wet avalanches running on the ground on shallow, entrainable snowcovers require higher friction values than large avalanches, running on warm, compacted snowcovers.			

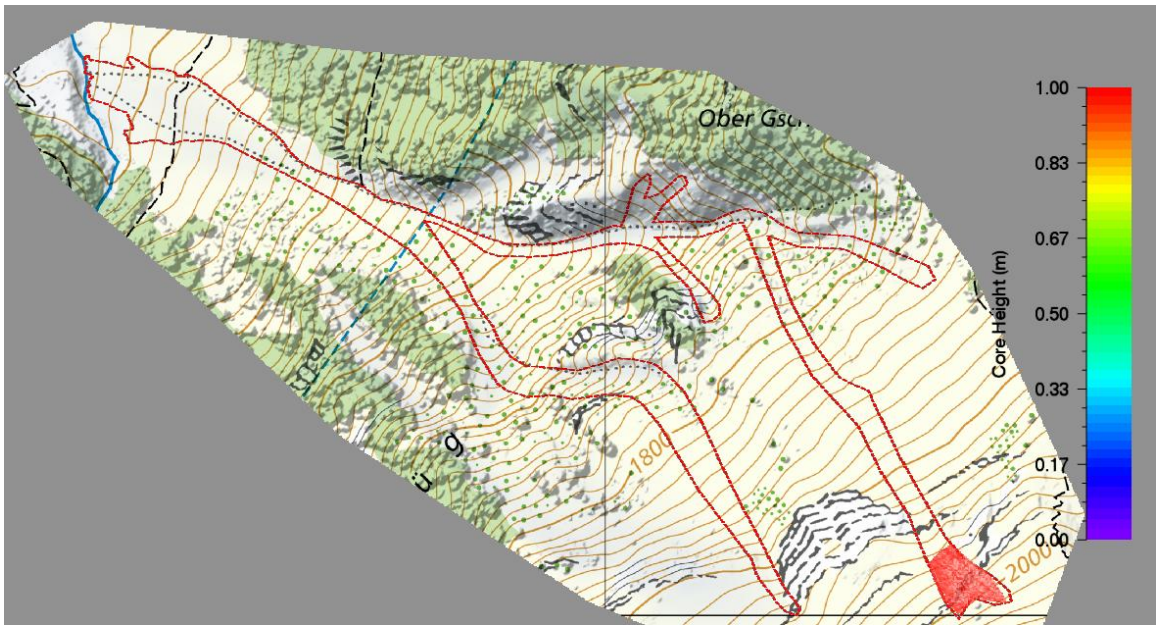


Figure 8.15: Grüenbödeli avalanche track: release location and documented avalanche path.

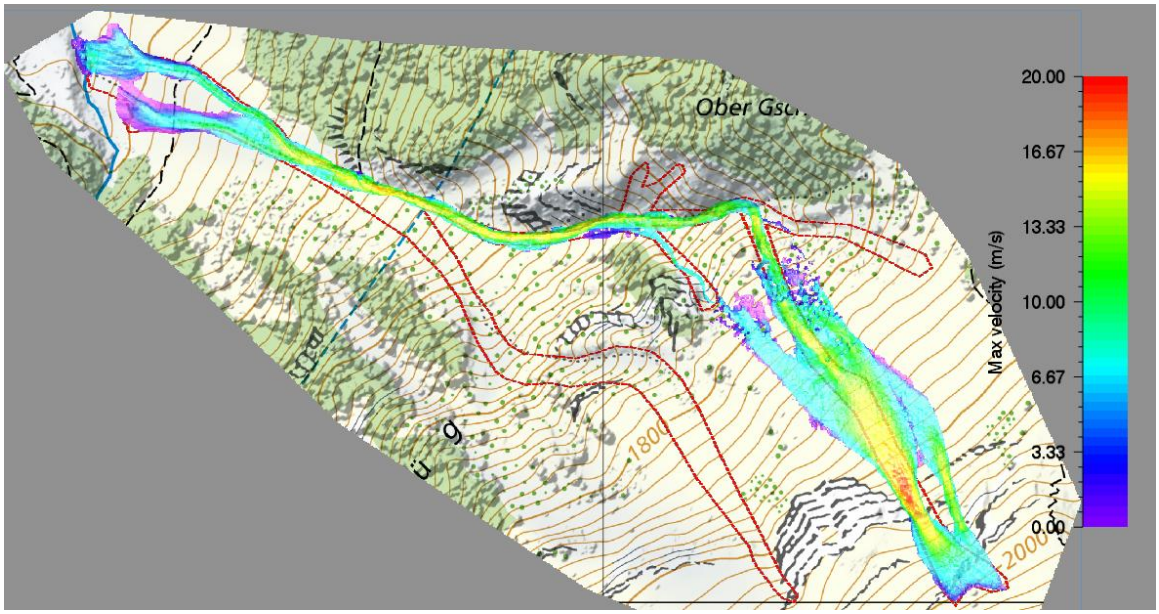


Figure 8.16: Calculated maximum flow velocities.

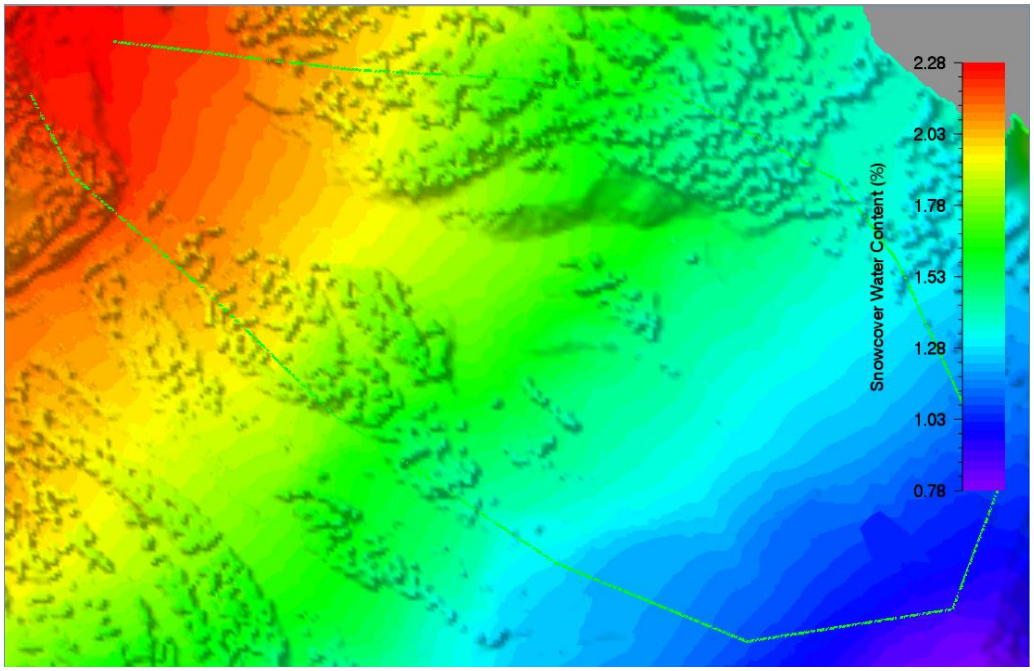


Figure 8.17: Snowcover water content in percent. In the release 1%, with a gradient of 0.2%/100m. The DEM was captured with a drone; forest cover was not removed from the elevation model.

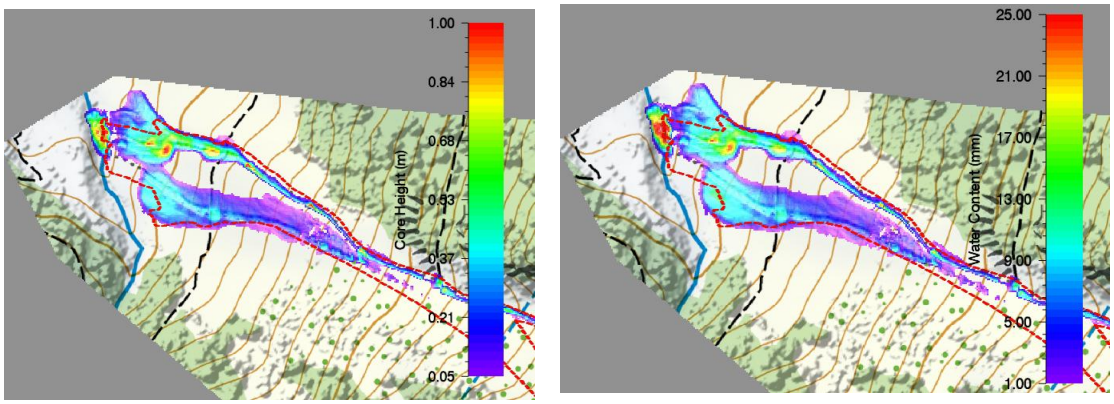


Figure 8.18: Calculated deposition heights (left) and water content (right). A total of 25mm of water is found in 1m deposits.

References

These publications, and many more, had a strong influence on **RAMMS::Extended**:

Reynolds, O. (1885). *On the Dilatancy of Media Composed of Rigid Particles in Contact. With Experimental Illustrations.* Philosophical Magazine, Series 5, 20(127), 469–481. https://doi.org/10.1080/14786448508627791.)

W. E. Ranz & W. R. Marshall, “Evaporation from Drops” (Parts I & II), Chemical Engineering Progress, 48 (1952)

Voellmy, A. (1955). Uber die Zerstorungskraft von Lawinen. *Bauzeitung*, *73*, 159-165.

Kojima, K. (1967). Densification of seasonal snow cover. In H. Oura (Ed.), *Physics of Snow and Ice* (Vol. 1, Part 2, pp. 929–952). Institute of Low Temperature Science, Hokkaido University.

Salm, B. (1979). *Fliessübergänge und Auslaufstrecken von Lawinen*. (Eidgenössisches Institut für Schnee- und Lawinenforschung: interner Bericht, Report No.: 566). Eidgenössisches Institut für Schnee- und Lawinenforschung.

Perla, R., Cheng, T. T., & McClung, D. M. (1980). A two-parameter model of snow-avalanche motion. *Journal of Glaciology*, 26(94), 197–207.

Gubler, H. U. (1981). *Messungen an Fliesslawinen*. (Eidgenössisches Institut für Schnee- und Lawinenforschung: interner Bericht, Report No.: 600). Eidgenössisches Institut für Schnee- und Lawinenforschung.

Haff, P. K. (1983). Grain flow as a fluid-mechanical phenomenon. Journal of Fluid Mechanics, *134*, *401-430*.

McClung, D. M., & Schaerer, P. A. (1985). Characteristics of flowing snow and avalanche impact pressures. *Annals of Glaciology*, 6, 9–14.

Gubler, H., Hiller, M., Klausegger, G., & Suter, U. (1986). *Messungen an Fliesslawinen. Zwischenbericht 1986. Mitteilungen des Eidg. Institutes für Schnee- und Lawinenforschung: Vol. 41.* Weissfluhjoch/Davos: Eidgenössisches Institut für Schnee- und Lawinenforschung.

Norem, H., Kristensen, K., & Tronstad, K. (1986). *The Ryggfonn project: Avalanche data from the winter 1984/85*. NGI Report 58120-08. Norwegian Geotechnical Institute.

Salm, B., & Gubler, H. (Eds.). (1987). *Internationales Symposium über die Bildung, Bewegung und Wirkungen von Lawinen. Symposium international sur la formation, le*

movement et les effets des avalanches. Davos, September 1986. Praxisbezogener Teil. Partie pratique. *Mitteilungen des Eidg. Institutes für Schnee- und Lawinenforschung: Vol. 43*. *Internationales Symposium über die Bildung, Bewegung und Wirkungen von Lawinen*. Davos: Eidgenössisches Institut für Schnee- und Lawinenforschung

Nishimura, K., Narita, H., Maeno, N., & Kawada, K. (1989). The internal structure of powder-snow avalanches. *Annals of Glaciology*, 13, 207–210. https://doi.org/10.3189/S0260305500007904

Norem, H., Irgens, F., & Schieldrop, B. (1989). Simulation of snow-avalanche flow in run-out zones. *Annals of Glaciology*, 13, 218–225. https://doi.org/10.3189/S026030550000793X

Gubler, H. (1989). Comparison of three models of avalanche dynamics. *Annals of Glaciology*, *13*, 82-89. https://doi.org/10.3189/s0260305500007680

Salm, B., Burkard, A., & Gubler, H. U. (1990). *Berechnung von Fliesslawinen. Eine Anleitung fuer Praktiker mit Beispielen*. *Mitteilungen des Eidg. Institutes für Schnee- und Lawinenforschung: Vol. 47*. Weissfluhjoch/Davos: Eidgenössisches Institut für Schnee- und Lawinenforschung.

Salm, B. (1993). Flow, flow transition and runout distances of flowing avalanches. *Annals of Glaciology*, 18, 221–226. https://doi.org/10.3189/S0260305500011551

Bozhinskiy, A. N., & Losev, K. S. (1998). *The fundamentals of avalanche science*. *Mitteilungen des Eidg. Institutes für Schnee- und Lawinenforschung: Vol. 55*. Davos: Eidg. Institut für Schnee- und Lawinenforschung.

Margreth, S., & Ammann, W. J. (2004). Hazard scenarios for avalanche actions on bridges. *Annals of Glaciology*, *38*, 89-96. https://doi.org/10.3189/172756404781814951

Bartelt, P., Buser, O., & Platzer, K. (2006). Fluctuation-dissipation relations for granular snow avalanches. *Journal of Glaciology*, *52*(179), 631-643. https://doi.org/10.3189/172756506781828476

Sovilla, B., Margreth, S., & Bartelt, P. (2007). On snow entrainment in avalanche dynamics calculations. *Cold Regions Science and Technology*, *47*(1-2), 69-79. https://doi.org/10.1016/j.coldregions.2006.08.012

Platzer, K., Bartelt, P., & Kern, M. (2007). Measurements of dense snow avalanche basal shear to normal stress ratios (S/N). *Geophysical Research Letters*, *34*(7), L07501 (5 pp.). https://doi.org/10.1029/2006GL028670

Sovilla, B., Schaer, M., & Rammer, L. (2008). Measurements and analysis of full-scale avalanche impact pressure at the Vallée de la Sionne test site. *Cold Regions Science and Technology*, 51(2–3), 122–137. https://doi.org/10.1016/j.coldregions.2007.05.006

Sovilla, B., Schaer, M., Kern, M., & Bartelt, P. (2008). Impact pressures and flow regimes in dense snow avalanches observed at the Vallée de la Sionne test site. *Journal of Geophysical Research: Earth Surface*, *113*(F1), 1-12. https://doi.org/10.1029/2006JF000688

Thibert, E., Baroudi, D., Limam, A., & Berthet-Rambaud, P. (2008). Avalanche impact pressure on an instrumented structure. *Cold Regions Science and Technology*, 54(3), 206–215. https://doi.org/10.1016/j.coldregions.2008.01.005

Buser, O., & Bartelt, P. (2009). Production and decay of random kinetic energy in granular snow avalanches. *Journal of Glaciology*, *55*(189), 3-12. https://doi.org/10.3189/002214309788608859

Buser, O., & Bartelt, P. (2011). Dispersive pressure and density variations in snow avalanches. *Journal of Glaciology*, *57*(205), 857-860. https://doi.org/10.3189/002214311798043870

Bartelt, P., Bühler, Y., Buser, O., Christen, M., & Meier, L. (2012). Modeling mass-dependent flow regime transitions to predict the stopping and depositional behavior of snow avalanches. *Journal of Geophysical Research: Earth Surface*, *117*, F01015 (28 pp.). https://doi.org/10.1029/2010JF001957

Sovilla, B., Schaer, M., & Rammer, L. (2013). The full-scale avalanche dynamics test site Vallée de la Sionne. In *Proceedings of the International Snow Science Workshop* (pp. 1350–1356). https://arc.lib.montana.edu/snow-science/objects/ISSW13_paper_O4-33.pdftc.copernicus.org+5

Feistl, T., Bebi, P., Dreier, L., Hanewinkel, M., & Bartelt, P. (2014). Quantification of basal friction for technical and silvicultural glide-snow avalanche mitigation measures. *Natural Hazards and Earth System Sciences*, *14*(11), 2921-2931. https://doi.org/10.5194/nhess-14-2921-2014

Buser, O., & Bartelt, P. (2015). An energy-based method to calculate streamwise density variations in snow avalanches. *Journal of Glaciology*, *61*(227), 563-575. https://doi.org/10.3189/2015JoG14J054

Feistl, T., Bebi, P., Teich, M., Bühler, Y., Christen, M., Thuro, K., & Bartelt, P. (2014). Observations and modeling of the braking effect of forests on small and medium avalanches. *Journal of Glaciology*, *60*(219), 124-138. https://doi.org/10.3189/2014JoG13J055

Bartelt, P., Vera Valero, C., Feistl, T., Christen, M., Bühler, Y., & Buser, O. (2015). Modelling cohesion in snow avalanche flow. *Journal of Glaciology*, *61*(229), 837-850. https://doi.org/10.3189/2015JoG14J126

Feistl, T., Bebi, P., Christen, M., Margreth, S., Diefenbach, L., & Bartelt, P. (2015). Forest damage and snow avalanche flow regime. *Natural Hazards and Earth System Sciences*, *15*(6), 1275-1288. https://doi.org/10.5194/nhess-15-1275-2015

Bartelt, P., Buser, O., Vera Valero, C., & Bühler, Y. (2016). Configurational energy and the formation of mixed flowing/powder snow and ice avalanches. *Annals of Glaciology*, *57*(71), 179-188. https://doi.org/10.3189/2016AoG71A464

Gauer, P., & Kristensen, K. (2016). Four decades of observations from NGI's full-scale avalanche test site Ryggfonn—Summary of experimental results. *Cold Regions Science and Technology*, 125, 162–176. https://doi.org/10.1016/j.coldregions.2016.02.009

Dreier, L., Bühler, Y., Ginzler, C., & Bartelt, P. (2016). Comparison of simulated powder snow avalanches with photogrammetric measurements. *Annals of Glaciology*, *57*(71), 371-381. https://doi.org/10.3189/2016AoG71A532

Bartelt, P., Bebi, P., Feistl, T., Buser, O., & Caviezel, A. (2018). Dynamic magnification factors for tree blow-down by powder snow avalanche air blasts. *Natural Hazards and Earth System Sciences*, *18*(3), 759-764. https://doi.org/10.5194/nhess-18-759-2018

Bartelt, P., Christen, M., Bühler, Y., Caviezel, A., & Buser, O. (2018). Snow entrainment: avalanche interaction with an erodible substrate. In *International snow science workshop proceedings 2018* (pp. 716-720).

Margreth, S. (2019). Lawinengefahrenkarten in der Schweiz. Avalanche hazard maps in Switzerland. *Wildbach- und Lawinenverbau*, *83*(184), 80-92.

Bartelt, P., Buser, O., Christen, M., & Caviezel, A. (2019). Dynamic magnification factors for snow avalanche impact (with pile-up) on walls and pylons. In M. Papadrakakis & M. Fragiadakis (Eds.), *Vol. 3. COMPDYN 2019. 7th international conference on computational methods in structural dynamics and earthquake engineering. Proceedings* (pp. 4376-4385). https://doi.org/10.7712/120119.7234.20047

Caviezel, A., Margreth, S., Ivanova, K., Sovilla, B., & Bartelt, P. (2021). Powder snow impact of tall vibrating structures. In M. Papadrakakis & M. Fragiadakis (Eds.), *Compdyn 2021 proceedings* (p. 19112 (13 pp.). https://doi.org/10.7712/120121.8868.19112

Bründl, M., Hafner, E., Bebi, P., Bühler, Y., Margreth, S., Marty, C., … Schweizer, J. (2021). Evaluation of the extraordinary avalanche situation in January 2018 in Switzerland. In N. Beyer Portner (Ed.), *14th Congress INTERPRAEVENT 2021. May 31st to June 2nd 2021.*

Virtual Congress, Norway. Conference Proceedings (pp. 17-23). International Research Society INTERPRAEVENT.

Gorynina, O., & Bartelt, P. (2023). Powder snow avalanche impact on hanging cables. *International Journal of Impact Engineering*, *173*, 104422 (11 pp.). https://doi.org/10.1016/j.ijimpeng.2022.104422

Wikstrom Jones, K., Wolken, G. J., Janes, M., Wilbur, C., Glaus, J., Bartelt, P., & Christen, M. (2023). Assessing powder cloud impact on electrical transmission lines at snowslide creek avalanche path in Southeast Alaska. In *ISSW proceedings*. *International snow science workshop proceedings 2023* (pp. 890-897).

Glaus, J., Wikstrom Jones, K., Bühler, Y., Christen, M., Ruttner-Jansen, P., Gaume, J., & Bartelt, P. (2023). *RAMMS::EXTENDED – Sensitivity Analysis of Numerical Fluidized Powder Avalanche Simulation in Three-Dimensional Terrain*. Proceedings of the International Snow Science Workshop (ISSW) 2023, Bend, Oregon.

Zhuang, Y., Xing, A., Bartelt, P., Bilal, M., & Ding, Z. (2023). Dynamic response and breakage of trees subject to a landslide-induced air blast. *Natural Hazards and Earth System Sciences*, *23*(4), 1257-1266. https://doi.org/10.5194/nhess-23-1257-2023.

Zhuang, Y., Piazza, N., Xing, A., Christen, M., Bebi, P., Bottero, A., … Bartelt, P. (2023). Tree blow-down by snow avalanche air-blasts: dynamic magnification effects and turbulence. *Geophysical Research Letters*, *50*(21), e2023GL105334 (12 pp.). https://doi.org/10.1029/2023GL105334.

Stoffel, L., Bartelt, P., & Margreth, S. (2024). Scenario-based avalanche simulations including snow entrainment, snow temperature and density. In K. Gisnås, P. Gauer, H. Dahle, M. Eckerstorfer, A. Mannberg, & K. Müller (Eds.), *Proceedings of the international Snow Science Workshop 2024* (pp. 374-380). Norwegian Geotechnical Institute.

Oberschmied, C., Stoffel, L., Margreth, S., & Bartelt, P. (2024). Large powder snow avalanches in South Tyrol in January 2019: comparison with scenario-based avalanche. In K. Gisnås, P. Gauer, H. Dahle, M. Eckerstorfer, A. Mannberg, & K. Müller (Eds.), *Proceedings of the international Snow Science Workshop 2024* (pp. 346-352). Norwegian Geotechnical Institute.