MASTER THESIS

SIMULATION OF SNOW SLIDE-OFF FROM SOLAR PANELS AND MODELLING OF SUBSEQUENT SNOW ACCUMULATION USING THE MATERIAL POINT METHOD

SUBMITTED BY: YAEL FRISCHHOLZ

AFFILIATION: SECTION OF ENVIRONMENTAL SCIENCE AND ENGINEERING (SIE),
ECOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE (EPFL),
LAUSANNE, SWITZERLAND

SUPERVISORS: DR. JOHAN GAUME,
EPFL ENAC IIE SLAB,
HEAD OF UNIT

PROF. MICHAEL LEHNING,
EPFL ENAC SSIE-GE,
FULL PROFESSOR

DR. XINGYUE LI,
TONGJI UNIVERSITY,
DISTINGUISHED RESEARCH FELLOW

April 2022

Acknowledgements

I would like to thank Varun Sharma for the time taken regularly and the precious advice. I also thank Xingyue Li for all the info shared on the MPM. I thank Michael Lehning for the careful revision and the logistics management in Davos. Finally, I thank the SLF workshop and cold lab teams for the support during the experiments.

Summary

Snow accumulation on and around solar power plants is one of their principal mechanical interactions in an alpine environment and has yet not been studied. In this thesis, a Material Point Method (MPM) adapted for snow avalanches is used in a multi-events fashion to produce simulated data of the snow accumulation below the panels under diverse snow type and geometrical configurations. A sensitivity analysis on snow parameters shows the snow intrinsic material parameters have a first order impact on the shape characteristics of the snow accumulation, while the parameters relative to the plastic deformation have a second order impact. The MPM snow model is then calibrated on small scale laboratory experiments which assesses the capacity of reproducing real snow accumulation below panels and provides a model with fixed snow parameters within 10% to 20% accuracy. A statistical regression model is then built based on meter scale simulations with varying geometrical parameters. The predictions have sub-centimeter average accuracy and precision.
The potential of the employed MPM model for simulation and modelling of snow slide-off is thus assessed. The snow parameters sensitivity analysis and experiments should be repeated for larger scales. The statistical model implemented is well adapted to behavior of the snow accumulation under varying panel geometry.

Contents

Introduction

In 2015, the international community agreed on the Sustainable Development Goals (SDGs) at the United Nation General Assembly to provide a sustainable future to the next generations. Objective number 7 focuses on the importance of renewable energy which potential towards the mitigation of climate change had already been reported by the Intergovernmental Panel on Climate Change (IPCC) in its special report in 2011[1].
In that context and for midlatitude mountainous regions, [2] showed the potential of solar photovoltaic (PV) technology if optimally placed in alpine environments. [3] modeled 100% renewable energy feasibility in Switzerland, a typical example of midlatitude and mountainous country, and PV appears as a key ingredient in the hydro-wind-PV mix. Since then, interest is growing for such alpine PV technology in the Swiss Alps. In the image of the already existing Axpo/IWB AlpinSolar project[4] or the future EES/Alpiq Gondosolar[5] installation, public and private entities have started to invest in this new type of infrastructure. While the complex interaction between radiation and the alpine environment is under investigation [6], [2], the interaction with the PV panels remains almost untouched, especially with respect to the snow component. The following questions which may have a large impact on the design of alpine PV technology, remain unaddressed: *Do panels locally affect snow accumulation? How high should panels be placed? Does snow affect panels efficiency? How does snow build up on panels? When does it slide off? When does it melt?*
This study tackles the impact solar panels have on the surrounding snow accumulation. In particular, it concentrates on the modelling of the accumulation below solar panels which evolves along the season. It also aims at characterizing the accumulation morphology and behavior. The structure of this study doesn't follow the classical scientific "methodology-results-discussion" scheme for its significant exploratory dimension in both the tools and the material. This study constitutes a first attempt to the modelling of snow accumulation below solar panels based on Material Point Method (MPM)[7] simulations which, in turn, relies on a snow constitutive model[8] that has yet never been used to that end.
This report is thus structured as follows: First, theory on the MPM is provided. Second, experiments that bring data to compare and calibrate the simulations, are presented. Third, a sensitivity analysis is conducted on the snow model parameters to understand how they impact the snow accumulation. Fourth, the model is calibrated on laboratory experiments. This provides with simulation-reality comparison data and yields a model with fixed snow parameters. Finally, simulations with varying geometry of panels are computed to feed a statistical model. Its performance and generalization are assessed.

1 | Methodology

KEY CHAPTER FEATURES

- **Theoretical background on continuum mechanics**
- **Description of the snow constitutive model**
- **Introduction to the Material Point Method (MPM)**

In this section, theoretical background on continuum mechanics is first presented (Section 1.1). Then, the Material Point Method (MPM) is introduced, as well as the constitutive model applied for snow (Section 1.2).

1.1 Theoretical background

The Material Point Method simulates motion and deformation of materials considered as continuum domains under constraint. It splits them into continuous particles to keep trace of mass, momentum and deformation. As explained in further details in Section 1.2, it needs a constitutive model for the considered material. Inspired from [9] and [10] some basics of continuum physics are hereafter presented as its formalization is used to apply and describe snow motion and its elastoplastic constitutive model. Derivation and basic elements are included. To guide the reader, the most important results are highlighted in boxes.

1.1.1 Motion

A material object is considered as a body (B) that consists of particles (P_i, i=1,2,3,...,N). Its motion yields different possible states called configurations and written Ω_t, where t is a parameter and can be seen as time. Hence, in this formalization, motion (incl. deformation) of the body B is a series of configurations. The series can be mapped as

$$\phi(\cdot, t) : \Omega_0 \rightarrow \Omega_t \text{ for } \Omega_0, \Omega_t \subset \mathbb{R}^d \tag{1.1.1}$$

where d is the dimension considered (usually 2 or 3). The mapping ϕ is assumed bijective. It is sometimes called the deformation map. Particles in the reference configuration Ω_0 are denoted as **X**. In the configuration at time t, Ω_t, particles are denoted as **x**.
Speed and acceleration can be derived from the mapping

$$\mathbf{V}(\mathbf{X}, t) = \frac{\partial \phi}{\partial t}(\mathbf{X}, t) \tag{1.1.2}$$

$$\mathbf{A}(\mathbf{X}, t) = \frac{\partial^2 \phi}{\partial t^2}(\mathbf{X}, t) = \frac{\partial \mathbf{V}}{\partial t}(\mathbf{X}, t) \tag{1.1.3}$$

1.1.2 Material derivative

Using the bijectivity of ϕ, speed and acceleration can be described equivalently both in terms of **x** and **X**. While the former can be referred to as the Eulerian description, the latter is usually referred to as the Lagragian decription. Equations 1.1.2 and 1.1.3 are Lagrangian and their counterparts are

$$\mathbf{v}(\mathbf{x}, t) = V(\phi^{-1}(\mathbf{X}, t), t) \tag{1.1.4}$$

$$\mathbf{a}(\mathbf{x}, t) = A(\phi^{-1}(\mathbf{X}, t), t) \tag{1.1.5}$$

The above expressions can then be rewritten in their sometimes called pull-back[1] notation

$$\mathbf{V}(\mathbf{x}, t) = v(\phi(\mathbf{X}, t), t) \tag{1.1.6}$$

$$\mathbf{A}(\mathbf{x}, t) = a(\phi(\mathbf{X}, t), t) \tag{1.1.7}$$

From equations 1.1.2 and 1.1.4

$$\frac{\partial \phi}{\partial t}(\phi^{-1}(\mathbf{X}, t), t) = \mathbf{v}(\mathbf{x}, t) \tag{1.1.8}$$

Now, deriving speed from acceleration using the chain rule on equation 1.1.6

$$\mathbf{A}(\mathbf{X}, t) = \frac{\partial}{\partial t}\mathbf{V}(\mathbf{X}, t) = \frac{\partial \mathbf{v}}{\partial t}(\phi(\mathbf{X}, t), t) + \frac{\partial \mathbf{v}}{\partial \mathbf{x}}(\phi(\mathbf{X}, t), t)\frac{\partial \phi}{\partial t}(\mathbf{X}, t) \tag{1.1.9}$$

The relationship between the Eulerian **a** and **v** can be written using equations 1.1.8 and 1.1.9

$$\mathbf{a}(\mathbf{x}, t) = A(\phi^{-1}(\mathbf{X}, t), t) = \frac{\partial \mathbf{v}}{\partial t}(\mathbf{x}, t) + \frac{\partial \mathbf{v}}{\partial \mathbf{x}}(\mathbf{x}, t)\frac{\partial \phi}{\partial t}(\phi^{-1}(\mathbf{X}, t), t) \tag{1.1.10}$$

So equation 1.1.9 simplifies to

$$\boxed{\mathbf{a}(\mathbf{x}, t) = \frac{\partial \mathbf{v}}{\partial t}(\mathbf{x}, t) + \frac{\partial \mathbf{v}}{\partial \mathbf{x}}(\mathbf{x}, t)\mathbf{v}(\mathbf{x}, t)} \tag{1.1.11}$$

This last expression gives the relation between Eulerian **a** and **v** which differs from the classical (Lagrangian) partial time differential. This will be useful in Section 1.2 and has the following notation

$$\boxed{\mathbf{a} = \frac{D}{Dt}\mathbf{v}} \tag{1.1.12}$$

1.1.3 Deformation

The gradient of the deformation map **F** describes the deformation of a body at time t

$$\boxed{\mathbf{F}(\mathbf{X}, t) = \frac{\partial \phi}{\partial \mathbf{X}}(\mathbf{X}, t)} \tag{1.1.13}$$

As illustrated in Figure 1.1.1, **F**(**X**,t) describes the material deformation at time t. It has the form of a d×d matrix

$$F(\cdot, t) : \Omega_0 \rightarrow \mathbb{R}^{d \times d} \tag{1.1.14}$$

An useful derived value is $\mathbf{J} = det(\mathbf{F})$, the deformation gradient Jacobian, which locally repre-

[1] Referring to Figure 1.1.1 the "pull-back" wording gives the idea of expressing motion as function of the original configuration (left). In other words, the pull-back version of a Eulerian decription is its Lagrangian counterpart. Following the same semantic, the pushed-forward notation expresses motion as function of the actual configuration (right). The pushed-forward version of a Lagrangian description is its Eulerian counterpart.

sents the ratio of the deformed body volume to the original one.

$\mathbf{J} = 1$ Rigid deformation (e.g. rotation)

$\mathbf{J} > 1$ Volume increased

$\mathbf{J} < 1$ Volume decreased

$\mathbf{J} \leq 0$ Special or degenerated cases

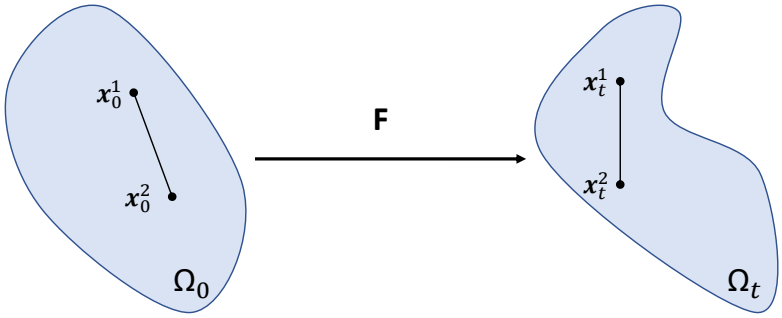

Figure 1.1.1: Illustration of the deformation gradient F between two configurations.

1.1.4 Stress

Stress is defined as the ratio of a force to the area it is applied to. If the force is parallel to the surface normal vector, it is a normal stress (σ). It results in either pure compression or tension according to the direction of the force. If the force is perpendicular to the normal vector, it is a shear stress (τ).

$$\sigma_{xx} = \frac{F_x}{A_x} \qquad \tau_{xy} = \frac{F_y}{A_x} \tag{1.1.15}$$

A common and practical way of expressing all stresses is the stress tensor $\boldsymbol{\sigma}$ which in 3D takes the following matrix form

$$\boldsymbol{\sigma} = \begin{vmatrix} \sigma_{11} & \tau_{12} & \tau_{13} \\ \tau_{21} & \sigma_{22} & \tau_{23} \\ \tau_{31} & \tau_{32} & \sigma_{33} \end{vmatrix} \tag{1.1.16}$$

It is worth noting that:

- $\boldsymbol{\sigma}$ is symmetric (i.e. $\tau_{21} = \tau_{12}$).
- $\boldsymbol{\sigma}$ values change with respect to the referential.
- The eigenvalues of $\boldsymbol{\sigma}$ are called the principal stresses, such that if the referential is placed on the basis formed by the eigenvectors, $\boldsymbol{\sigma}$ is diagonal and only pure normal stresses remain.

- There exist many referential invariants of $\boldsymbol{\sigma}$. Important ones in this context are

$$\boxed{\begin{aligned} p &= -\frac{1}{3} tr(\boldsymbol{\sigma}) \\ \mathbf{s} &= \boldsymbol{\sigma} + p\mathbf{I} \\ q &= (\frac{3}{2}\mathbf{s} : \mathbf{s})^{1/2} \end{aligned}} \tag{1.1.17}$$

$$\mathbf{s} = \boldsymbol{\sigma} + p\mathbf{I} \tag{1.1.18}$$

$$q = (\frac{3}{2}\mathbf{s} : \mathbf{s})^{1/2} \tag{1.1.19}$$

  where, p can be seen as the isotropic pressure, **s** the deviatoric stress tensor and q the Mises equivalent stress.

More on stress tensors

The principal denominator of stress tensors is the configuration they use to describe stress. The Cauchy stress $\boldsymbol{\sigma}$ describes the stress in the deformed configuration. It is the physically most relevant and most used stress tensor.

$$\boldsymbol{\sigma} = \begin{vmatrix} \sigma_{11} & \tau_{12} & \tau_{13} \\ \tau_{21} & \sigma_{22} & \tau_{23} \\ \tau_{31} & \tau_{32} & \sigma_{33} \end{vmatrix} \tag{1.1.20}$$

The Kirchhoff stress $\boldsymbol{\tau}$ also called the weighted Cauchy stress is expressed in the deformed configuration as well.

$$\boxed{\boldsymbol{\tau} = \mathbf{J}\boldsymbol{\sigma}} \tag{1.1.21}$$

The First Piola-Kirchhoff ($\boldsymbol{\sigma}^{PK1}$) stress is expressed with forces in the deformed configuration applied on areas described in the reference configuration. It is an asymmetric tensor. For infinitesimal deformations, $\boldsymbol{\sigma}^{PK1}$ is equal to $\boldsymbol{\sigma}$.

$$\boxed{\boldsymbol{\sigma}^{PK1} = \mathbf{J}\boldsymbol{\sigma}\mathbf{F}^{-T}} \tag{1.1.22}$$

The Second Piola-Kirchhoff stress ($\boldsymbol{\sigma}^{PK2}$) is expressed with forces in the reference configuration applied on areas described in the reference configuration as well. It is a symmetrical tensor.

$$\boxed{\boldsymbol{\sigma}^{PK2} = \mathbf{J}\mathbf{F}^{-1}\boldsymbol{\sigma}\mathbf{F}^{-T}} \tag{1.1.23}$$

1.1.5 Strain

Strain is the deformation of a solid due to an applied stress. Again, strain decomposes in normal and shear components which are directly grouped in the small strain tensor

$$\boldsymbol{\epsilon} = \begin{vmatrix} \epsilon_{11} & \gamma_{12} & \gamma_{13} \\ \gamma_{21} & \epsilon_{22} & \gamma_{23} \\ \gamma_{31} & \gamma_{32} & \epsilon_{33} \end{vmatrix} \tag{1.1.24}$$

where $\epsilon_{ii} = \frac{\partial \mathbf{u}_i}{\partial X_i}$ with **u(X)** a displacement field and i=1,2,3 in 3D and $\gamma_{ij} = \frac{\partial \mathbf{u}_i}{\partial X_j} + \frac{\partial \mathbf{u}_j}{\partial X_i}$ with i≠j. Similar to stress note that:

- $\boldsymbol{\epsilon}$ is symmetric (i.e. $\gamma_{21} = \gamma_{12}$).
- $\boldsymbol{\epsilon}$ values change with respect to the referential.
- The eigenvalues of $\boldsymbol{\epsilon}$ are called the principal strains, such that if the referential is placed on the basis formed by the eigenvectors, $\boldsymbol{\epsilon}$ is diagonal and only pure normal strains remain.

The *small* strain tensor lacks exactitude for pure rotations. An example of the inaptitude of ϵ towards rotation is given in Appendix A.1. It is enough to remember that ϵ is a good approximation when rotation angles are small and exact when no rotation occurs. The more common deformation-gradient-based expression for the small strain tensor reads

$$\boldsymbol{\epsilon} = \frac{1}{2}(\mathbf{F} + \mathbf{F}^T) - \mathbf{I} \tag{1.1.25}$$

More on strain tensors

Many strain descriptions exist. Those adapted to rotation decompose the gradient **F** into a rotation **R** and a non-rotational deformation **U** in the polar decomposition of **F**

$$\mathbf{F} = \mathbf{RU} \tag{1.1.26}$$

Then knowing that **R** doesn't lead to any strain, only **U** can be treated for strain. Using equations 1.1.25 and 1.1.26 yields a new strain tensor

$$\boldsymbol{\epsilon}' = \mathbf{U} - \mathbf{I} \tag{1.1.27}$$

Obtaining U is however not an easy task. The brute force method is to use the singular value decomposition of **F**. It is a heavy operation. Alternatively different expression of a strain tensor were proposed using the right or left Cauchy-Green deformation tensor

$$\mathbf{F}\mathbf{F}^T = (\mathbf{RU})(\mathbf{RU})^T = \mathbf{U}^T\mathbf{I}\mathbf{U} = \mathbf{U}^T\mathbf{U} \tag{1.1.28}$$

which also removes the rotation using $\mathbf{R}\mathbf{R}^T$=**I**.
The Hencky strain uses the left one

$$\boxed{\boldsymbol{\epsilon} = \frac{1}{2} ln(\mathbf{F}\mathbf{F}^T)} \tag{1.1.29}$$

and the Green-Lagrangian strain uses the right one

$$\mathbf{E} = \frac{1}{2}(\mathbf{F}^T\mathbf{F} - \mathbf{I}) \tag{1.1.30}$$

Both strains tensors are rotation insensitive. The Green-Lagrangian strain is a good approximation of the small strain tensor for small strains. The Hencky strain is more appropriated for larger and elastic deformation [8].

1.1.6 Hyperelasticity

A hyperelastic material is an ideal purely elastic material for which an energy density function (Ψ) is a constitutive model. For this type of materials the first Piola-Kirchhoff stress directly relates Ψ to stress [9]

$$\boxed{\boldsymbol{\sigma}^{PK1} = \frac{\partial \Psi}{\partial \mathbf{F}}} \tag{1.1.31}$$

Using equ. 1.1.22 the density function is related to the Cauchy stress

$$\boxed{\boldsymbol{\sigma} = \frac{1}{\mathbf{J}} \frac{\partial \Psi}{\partial \mathbf{F}} \mathbf{F}^T} \tag{1.1.32}$$

The Saint Venant-Kirchhoff constitutive model is one simple example of the models adopted for Ψ

$$\boxed{\boldsymbol{\sigma}^{PK2} = 2\mu \mathbf{E} + \lambda \, tr(\mathbf{E}) \mathbf{I}} \tag{1.1.33}$$

where, σ^{PK2} is the second Piola-Kirchhoff stress and with μ and λ the Lamé parameters

$$\mu = \frac{\mu}{2(1+\nu)} \qquad \lambda = \frac{E\nu}{(1+\nu)(1-2\nu)} \tag{1.1.34}$$

where, E is the Young's modulus and ν is the Poisson's ratio. Both originate from Hook's law. Other models exist (Neo-Hookean, Fixed Corotated, etc.). Refer to [9] for more details on the energy density function options.

1.2 The Material Point Method (MPM)

The Material Point Method is a hybrid Eulerian/Lagrangian numerical algorithm. It descretizes continuum material into Lagrangian particles to track their position, mass, velocity and deformation. In order to update their states, it solves the governing mass (equ. 1.2.1) and momentum (equ. 1.2.2) conservation equations on a regular Eulerian grid in the background

$$\frac{D}{Dt} \rho(\mathbf{x}, t) + \rho(\mathbf{x}, t) \nabla^{\mathbf{x}} \cdot \mathbf{v}(\mathbf{x}, t) = 0 \tag{1.2.1}$$

$$\rho(\mathbf{x}, t) \frac{D\mathbf{v}}{Dt} = \nabla^{\mathbf{x}} \cdot \sigma + \rho(\mathbf{x}, t) \mathbf{g} \tag{1.2.2}$$

MPM adopts a user-adjustable material constitutive model used for deformations. It links the Cauchy stress to strain through the first Piola-Kirchhoff stress

$$\boldsymbol{\sigma} = \frac{1}{\mathbf{J}} \frac{\partial \Psi}{\partial \mathbf{F}} \mathbf{F}^T \tag{1.2.3}$$

where Ψ is a potential energy density, **F** is the deformation gradient and **J** and $\boldsymbol{\sigma}$ are the deformation gradient determinant and the Cauchy stress tensor respectively.

1.2.1 Snow constitutive model

The constitutive model was developed by [8]. Different from hyperelastic materials, snow shows a mixed mode elastic and plastic behavior. Its deformation gradient thus has the following multiplicative decomposition

$$\mathbf{F} = \mathbf{F}^E \mathbf{F}^P \tag{1.2.4}$$

where, F^E and F^P are the elastic and plastic part of the deformation respectively.

The model is composed of a constitutive relation for the elastic energy density and a shear-compression yield surface, a hardening law and a flow rule for the plastic energy dissipation. The St. Venant-Kirchhoff energy density is adopted for the elastic constitutive relation

$$\Psi(\mathbf{F}^E) = \mu tr(\boldsymbol{\epsilon}^2) + \frac{1}{2}\lambda tr(\boldsymbol{\epsilon})^2 \tag{1.2.5}$$

where μ and λ are the Lamé parameters and ϵ is the Hencky strain.
For the plastic relation, a Modified Cam-Clay yield criterion (Figure 1.2.1) is taken and is expressed in the invariants p-q space as

$$y(p,q) = (1+2\beta)q^2 + M^2(p+\beta p_0)(p-p_0) \tag{1.2.6}$$

where, the pressure p = -tr($\boldsymbol{\tau}$)/d with $\boldsymbol{\tau}$ the Kirchhoff stress tensor and d the dimension. Positive and negative p values respectively indicate compression and tension. The Mises stress q indicates shear when positive. The internal friction M is the slope of the critical state line. The isotropic tensile strength is βp_0, with β the cohesion factor and p_0 the pre-consolidation pressure.
The yield surface is used when comparing with p-q trials (p_t, q_t) obtained assuming pure elastic deformation of the material. If y(p_t,q_t)>0, as in Figure 1.2.1, some plastic deformation must be involved. The following hardening law is applied

$$p_0 = K\sinh(\xi max(-\epsilon_v^P, 0)) \tag{1.2.7}$$

with $K = \frac{E}{3-6\nu}$, where E is the Young's modulus, ν is the Poisson's ratio, ξ is the hardening factor and ϵ_v^P is the volumetric plastic strain. When the trials are on the left of the yield surface, the material is in tension and the plastic volume increases (-ϵ_v^P<0) which, according to the hardening law, causes p_0 to decrease. It's a softening of the material. On the contrary, when the trials are on the right of the yield surface, the material is compressed (-ϵ_v^P>0), p_0 increases, it causes a hardening. The direction of ϵ^P in Figure 1.2.1 is governed by the flow rule adopted. Refer to [8] for details on the flow rule.

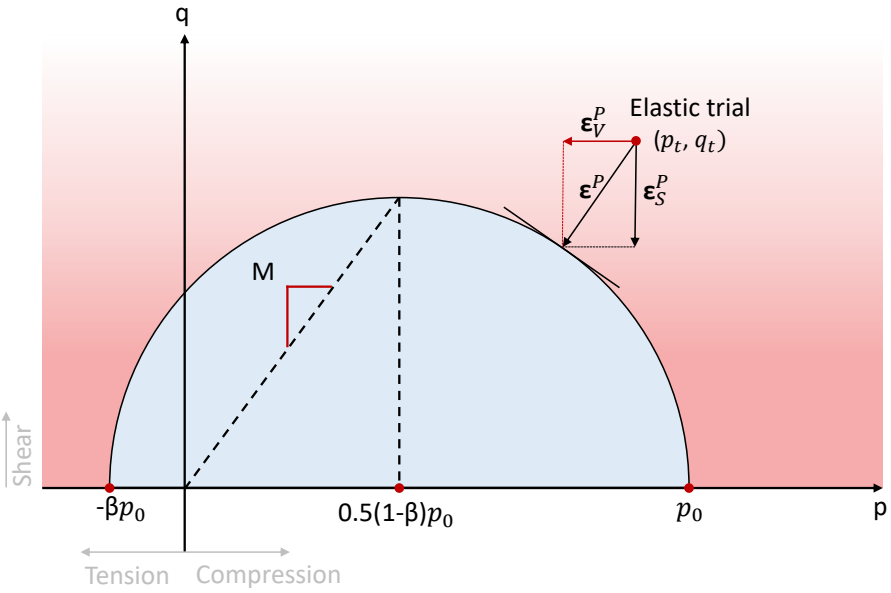

Figure 1.2.1: Adapted from [11], the yield surface in the invariants p-q space with its key parameters. An example of elastic trial is also shown to illustrate the hardening law variable ϵ_V^P. ϵ^P is dictated by the flow rule.

1.2.2 MPM algorithm update procedure

Each step of the update algorithm is hereafter described inspired from [7], [9] and [11]. A symplectic Euler time integration is used on the grid for position and velocity update

$$\mathbf{x}_\mathbf{i}^{n+1} = \mathbf{x}_\mathbf{i} + \Delta t \mathbf{v}_\mathbf{i}^{n+1} \tag{1.2.8}$$

$$\mathbf{v}_\mathbf{i}^{n+1} = \mathbf{v}_\mathbf{i}^n + \Delta t \mathbf{f}_\mathbf{i}(\mathbf{x}_\mathbf{i}^n)/m_\mathbf{i} = \mathbf{v}_\mathbf{i}^n + \Delta t \mathbf{a}_\mathbf{i}^n \tag{1.2.9}$$

where, **x** is the position, **a** the acceleration, **v** the velocity, **i**=(i,j,k) the grid cell index, n the time step, **f** the force and Δt the length of the time step.
Particles are given initialized properties: a volume V_p^0, a position x_p, a velocity v_p and their material characteristics.

1. Mass and velocity of Lagrangian particles are transferred to the grid nodes. Mass is transferred using interpolating grid functions resulting in weighted masses

$$m_\mathbf{i}^n = \sum_p m_p w_{\mathbf{i},p}^n \tag{1.2.10}$$

   Velocity is transferred using a PIC-FLIP method[2] [7]. Refer to Section 1.2.3 for further details on the particle-grid transfer.

2. Forces at each node are updated considering the elastic stresses of particles through the material energy density function Ψ. The nodal elastic force is the negative gradient of

[2] In this case, a FLIP-PIC ratio of 0.99 was used.

the potential energy at each node [9]

$$\mathbf{f_i}(\mathbf{x}) = -\frac{\partial e}{\mathbf{x_i}}(\mathbf{x}) \tag{1.2.11}$$

where,

$$e = \sum_p V_p^0 \Psi_p(\boldsymbol{F_p}) \tag{1.2.12}$$

The nodal force is hence finally computed as [9]

$$\boldsymbol{f_i^n} = -\sum_p V_p^0 (\frac{\partial \Psi}{\partial \boldsymbol{F}}(\boldsymbol{F_p^n}))(\boldsymbol{F_p^n})^T \nabla w_{\boldsymbol{i}p}^n \tag{1.2.13}$$

3. Velocities are updated using equ.1.2.9.

4. A purely elastic behavior of the material is assumed and constitutes the elastic trial. The deformation gradient is updated as

$$\hat{\boldsymbol{F}}_{Ep}^{n+1} = (\boldsymbol{I} + \Delta t \nabla \boldsymbol{v}_p^{n+1}) \boldsymbol{F}_{Ep}^n \tag{1.2.14}$$

5. The elastic trial ($\hat{\boldsymbol{F}}_{Ep}^{n+1}$) is tested against the yield condition (equ.1.2.6). If y>0, some plastic deformation will occur

$$\hat{\boldsymbol{F}}_{Ep}^{n+1} = \boldsymbol{F}_{Ep}^{n+1} \boldsymbol{F}_{Pp}^* \tag{1.2.15}$$

and the plastic deformation is updated as

$$\boldsymbol{F}_{Pp}^{n+1} = \boldsymbol{F}_{Pp}^* \hat{\boldsymbol{F}}_{Pp}^{n+1} \tag{1.2.16}$$

According to the nature of the trial, the material will harden or soften (Section 1.2.1).

6. Velocities are finally transferred back to particles from the grid using the PIC-FLIP method (equ.1.2.26).

7. Particles positions are updated using equ. 1.2.8.

1.2.3 Particle-Grid transfer

The transfer between the particles and the grid and *vice versa* are key steps in the MPM algorithm. The different existing methods are hereafter described.

Interpolating functions

The transfer between continuous and discrete description requires an interpolating grid-based function which forms the weights.

In 3D, this function has the form of a dyadic product of three 1D basis functions [9]

$$N_{\mathbf{i}}^h(\mathbf{x}) = N(\frac{1}{h}(x-x_i))N(\frac{1}{h}(y-y_i))N(\frac{1}{h}(z-z_i)) \tag{1.2.17}$$

where, $\mathbf{x}$ = (x,y,z) is the particle position, h is the grid size, x_i=(x_i,y_i,z_i) is the position of node $\mathbf{i}$.

The weights between particles and nodes are consequently formed as

$$w_{\mathbf{i}p}^n = N_{\mathbf{i}}^h(\mathbf{x}_p^n) \tag{1.2.18}$$

where, $\mathbf{x}_p^n$ is the position of particle p at time step n.

The grid-based function N can take various forms, each with its advantages and disadvantages. They must however all make sure that the sum of the nodes mass is the same as the sum of particles mass.

$$\sum_i m_i = \sum_p m_p \tag{1.2.19}$$

Therefore, the function N must have the partition of unity property [9]:

$$\sum_i N_i^h(\mathbf{x}) = 1, \forall \mathbf{x} \tag{1.2.20}$$

The function adopted here is a quadratic B-spline defined as

$$w_{\mathbf{i}p}^n = \begin{cases} \frac{3}{4} - x^2 & 0 \le |x| < \frac{1}{2} \\ \frac{1}{2}(\frac{3}{2} - |x|)^2 & \frac{1}{2} \le |x| < \frac{3}{2} \\ 0 & \text{otherwise} \end{cases} \tag{1.2.21}$$

and its gradient, used to compute forces, reads

$$\nabla w_{\mathbf{i}p}^n = \begin{cases} -2x & 0 \le |x| < \frac{1}{2} \\ x + \frac{3}{2} & -\frac{1}{2} \ge x > -\frac{3}{2} \\ x - \frac{3}{2} & \frac{1}{2} \le x < \frac{3}{2} \\ 0 & \text{otherwise} \end{cases} \tag{1.2.22}$$

Other functions as cubic B-splines can be used and lead to less numerical errors [9]. However, quadratic B-splines are lighter and easier to compute which leads to a higher time performance of the algorithm. Linear interpolation function have discontinuous derivatives which would cause discontinuous forces. An illustration of the three types of interpolating functions and their derivatives is given in Figure 1.2.2.

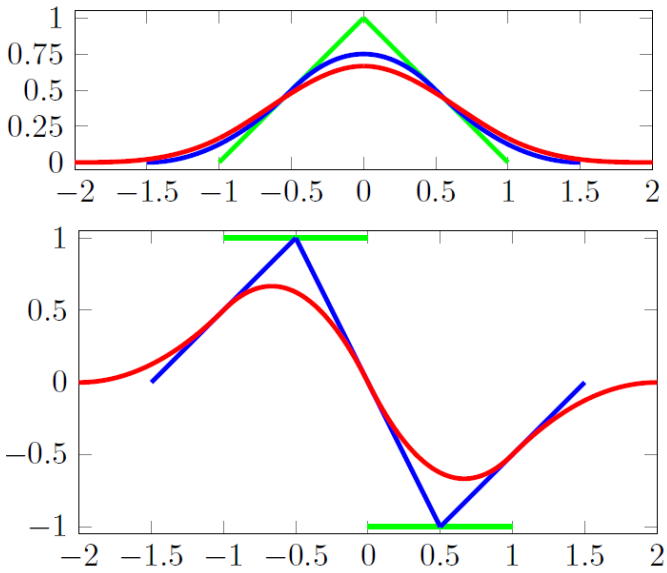

Figure 1.2.2: Plots of the different grid-based functions taken from [12]. Piecewise linear (green), quadratic B spline (blue), cubic B spline (red) functions (top) and their derivatives (bottom).

Velocity transfer

Several velocity transfer techniques exist, each with advantages and disadvantages. The two principal are the Particle-In-Cell method (PIC) [13] and the Fluid Implicit Particle method (FLIP) [14].
The particle to grid transfer using the PIC method [7] reads

$$\mathbf{v}_\mathbf{i}^n = \frac{\sum_p m_p w_{\mathbf{i}p}^n \mathbf{v}_p^n}{m_\mathbf{i}^n} \tag{1.2.23}$$

and the grid to particle transfer [15]

$$\mathbf{v}_{PICp}^{n+1} = \sum_i w_{\mathbf{i}p}^n \hat{\mathbf{v}}_\mathbf{i}^{n+1} \tag{1.2.24}$$

Using the FLIP method, only the grid to particle transfer differs

$$\mathbf{v}_{FLIPp}^{n+1} = \mathbf{v}_p^n + \sum_\mathbf{i} w_{\mathbf{i}p}^n (\hat{\mathbf{v}}_\mathbf{i}^{n+1} - \mathbf{v}_\mathbf{i}^n) \tag{1.2.25}$$

The PIC grid to particle transfer causes large dampening [15]. The FLIP grid to particle transfer only adds up velocity change and alleviate the dampening problem. However it shows more instability [15]. PIC and FLIP can thus be combined to decrease FLIP instability and reduce

PIC dampening

$$\mathbf{v}_p^{n+1} = \alpha \mathbf{v}_{FLIPp}^{n+1} + (1-\alpha)\mathbf{v}_{PICp}^{n+1} \tag{1.2.26}$$

In this context, α is set to 0.99 as [15] and [8] showed its good performances.

Collision objects

Within the simulation domain, the material particles potentially collide with objects (level sets) defined by the user. These objects restrain the simulation domain and form its boundaries. The MPM allows 3 different boundary conditions to apply in case of a collision between the material particles and any level set. Each implies a different treatment of the collision resulting velocity as follows:

1. STICKY: the tangential and normal components of the particle speed vector are set to 0. The material *sticks* to the collision object. This results in no friction force between the particles and the level sets.

2. SLIP: the collision resultant speed is projected on the collision object tangent. The normal speed is set to 0. Then, friction is considered.

3. SEPARATE: the collision resultant speed isn't modified and particles are free to leave the collision object surface. Then, friction is considered.

KEY CHAPTER TAKEAWAYS

- **The MPM is strongly modular. It adopts user defined material constitutive models and domains.**
- **In the following, the MPM will be thoroughly used to simulate and study snow slide-off from a tilted and elevated panel.**
- **To compare simulations results, since no data exist in the literature, experiments were carried and are presented in the following Chapter. Simulations and measurements are then joined in Chapter 4.**

2 | Laboratory experiment

KEY CHAPTER FEATURES

- **Laboratory experiment setup and procedure.**
- **First observations and characterization of the snow accumulation below a tilted and elevated panel after multiple slide-off events.**

The experiments provide with first rough observations of snow accumulation below a tilted and elevated panel. They also allow a first characterization of the accumulation. The model small scale was chosen so that manipulations are simple and allow to easily play with geometrical parameters. These experiments would then also provide data for simulation's parameters calibration.

2.1 Setup

The model presented in Figure 2.1.1 was placed in a cold room at -20°C. The panel and ground are Plexiglas(registered trademark) boards with a static friction coefficient measured *a posteriori* between 0.6 and 0.8. Natural snow from the Fluela pass (GR), also kept at -20°C, was used along with a 3mm mesh size sieve. Considering the cold temperature and the sieving, it is worth noticing that the used snow had a very low cohesion. Its density was $210 kg/m^3$.

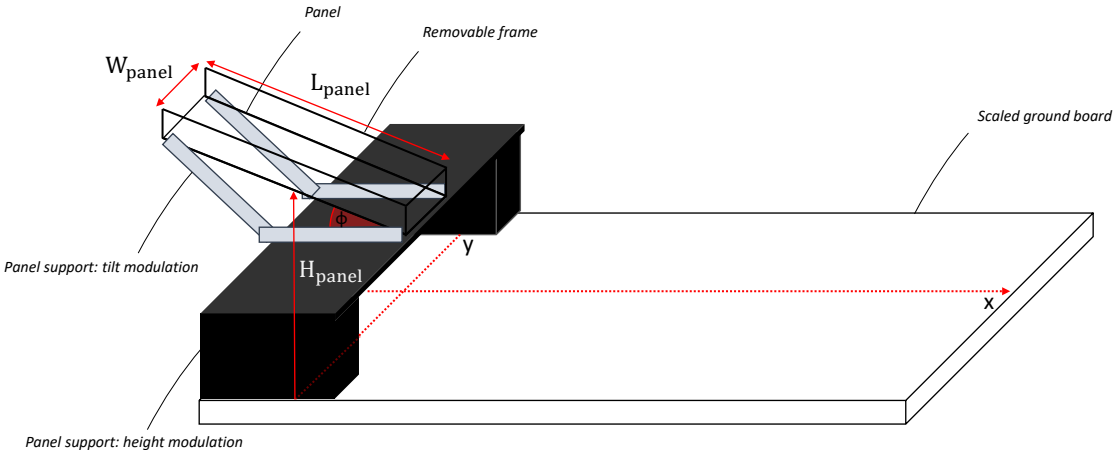

Figure 2.1.1: Scheme of the experiments setup in laboratory.

2.2 Characterization

As introduced at the beginning of this section, the small scale experiment offers first controlled observations. As observed in Figures 2.4.1 and 2.4.2, the accumulation takes an asymmetrical shape pictured in Figure 2.2.1 and inspired from [16]. The *Dunes* described by [16] have the same type of snow accumulation morphology. This wording is however not reused here because it originates from wind dependent phenomena and confers the idea of a lasting formation. In the case of slide-off events and free fall on ground, the idea of a sudden event, piling up towards a larger and higher accumulation justifies a new wording. For the accumulation of snow on ground resulting from slide-off on the panel, the name *stack* is introduced.

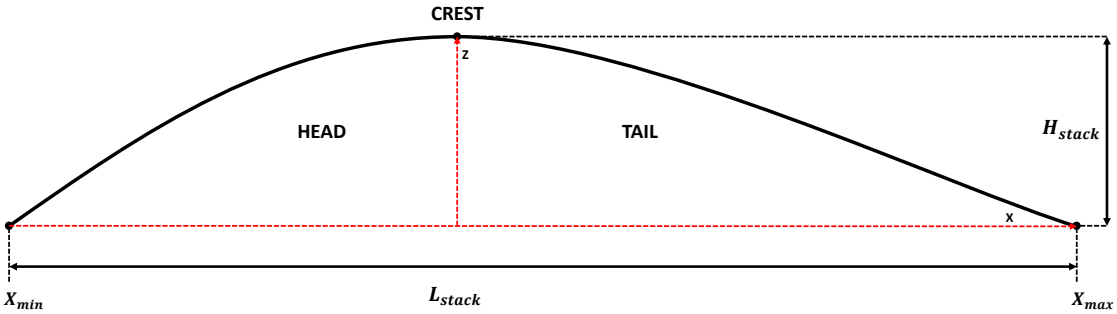

Figure 2.2.1: Characterisation of a typical *stack*. Its length is the distance between the two ends of the head and tail. Its height is the elevation at the crest. The x axis is in the direction of slide-off.

2.3 Procedure

Experiments for 30°, 40° and 50° of panel tilt were carried out. For each configuration, 5 slide-off events of 2cm thick snow layers were consecutively triggered, forming a growing stack on the horizontal ground. Each step was recorded to measure the stack after the experiments thanks to the 2.5 cm scales placed below the ground panel and beside it, parallel to the vertical axis as shown in Appendix A.2. The rather coarse scale resolution forced the estimation of in-between lying features. The stack was measured as a continuous body. In other words, the stack is considered until it has some discontinuity (e.g. a rolling compacted ball of snow that detached from the stack). Experiments for 30° panel tilt were not all successful. At several steps the slide-off stopped on the panel due to the snow-panel friction. In such cases, the slab was manually accompanied until completion. The results of the 30° tilt rounds are presented but are not considered in the calibration (Section 4).

Table 2.3.1

Geometrical parameters			
Panel height	H_{panel}	0.33	m
Panel Length	L_{panel}	0.245	m
Panel Width	W_{panel}	0.142	m
Panel tilt	Φ	30-40-50	°
Ground tilt	Ψ	0	°
Snow parameters			
Slab Thickness	T_{snow}	0.02	m
Slab Length	L_{snow}	0.245	m
Slab Width	W_{snow}	0.142	m
Number of slide-off	$N_{slide-off}$	3	-

The stacks' measured features are its height, H_{stack}, and its length, L_{stack}, along the X axis as presented in Figure 2.2.1. This last feature is derived from the maximum and minimum values on the X axis.

Table 2.3.2

Feature	Measurement	Other name
H_{stack}	max(Z)	z_Zmax
L_{stack}	max(X)-min(X)	Xlength

2.4 Measurements

Results from the experiments are hereafter presented. Different trends can be pointed out. Note that stack's boundaries are difficult to define when measured "by hand". This has a certain impact on the measurement of the stack length.

2.4.1 Accumulation length

Figure 2.4.1 presents the head (Xmin) position evolution. There is a clear order between groups. The smallest tilt being the closest to the origin and the largest tilt, the furthest. The head shows a constant displacement towards the origin[1]. The tail (Figure 2.4.2) shows a similar group behavior but has almost no displacement after the first slide-off event. This results in a stack length (Figure 2.4.3) strongly correlated to the stack's head. The growth and group order are similar.

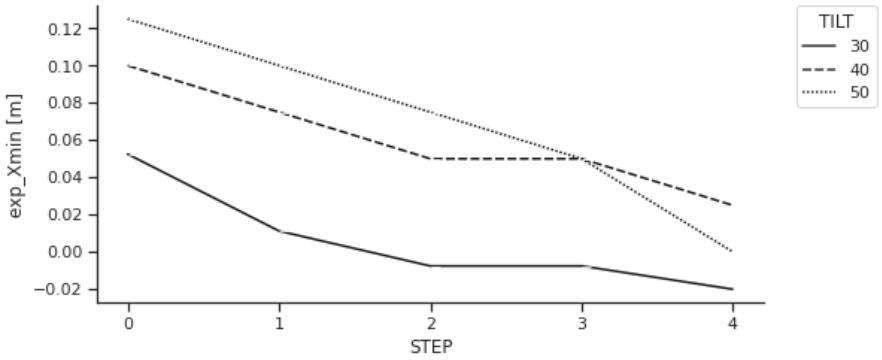

Figure 2.4.1: The accumulation minimum boundary on the x axis grows at each event.

[1]The origin is the center of the panel, the closest to the origin means the furthest from the impact point.

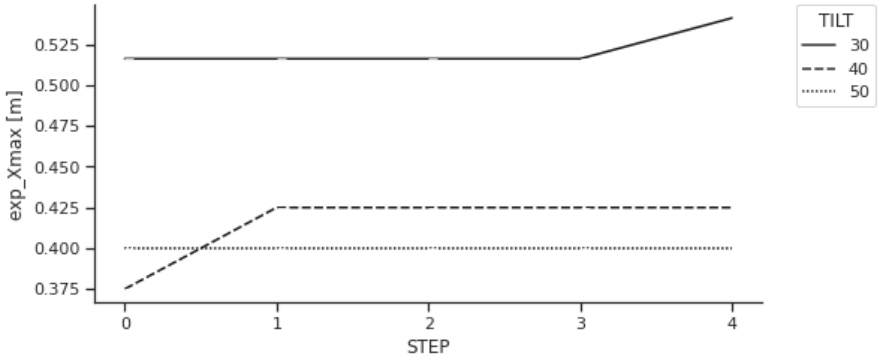

Figure 2.4.2: The accumulation maximum boundary on the x axis has a strong first growth and then stabilizes until the end. The kinetic energy of the falling snow layer at each step is theoretically the same. The first however only has the falling snow plasticity to dissipate it, whereas subsequent steps can use the existing stack plasticity as well. This results in a first slide-off lending rather elastic which causes the formation of the tail. It then eventually gets thicker but rarely longer.

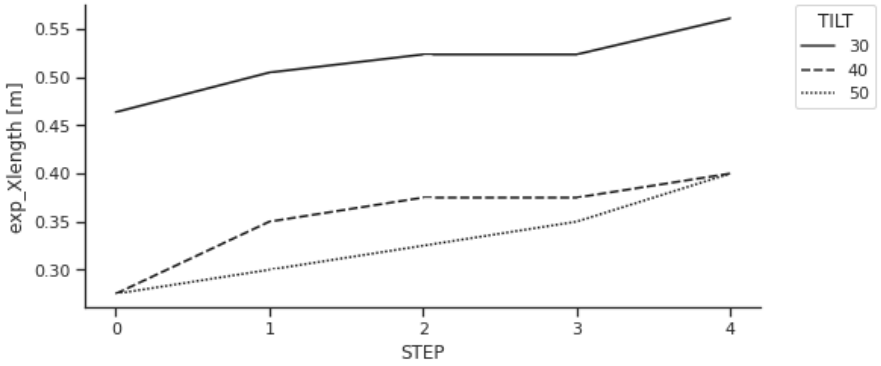

Figure 2.4.3: The length of the accumulation is the result of the tail and head difference. Since the tail is rather stable it is more correlated to the head change.

2.4.2 Accumulation height

The measurements show a logarithmic trend in the stack's height growth. The groups order isn't clear. In comparison with the length (Figure 2.4.3), it is expected that height is inversely proportional. Overall this is not verified. Groups TILT 30° and 40° almost show the same behavior for length and height. Only group TILT 50° shows a growth of length when the height remains constant in STEPS 1 to 3.

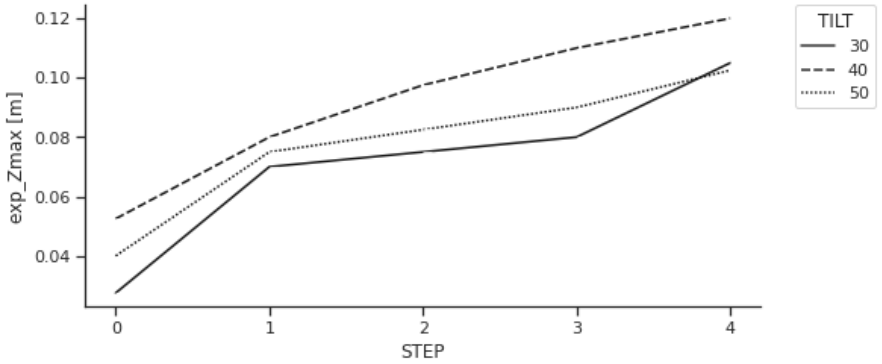

Figure 2.4.4: The height of the accumulation is expected to be inversely correlated to its length to some extents. The stack gets higher as long as it does not collapse and gets longer when it does. This effect is however buffered by the width of the stack and snow plasticity. The measurement show a constant growth that has a logarithmic trend. The bigger the stack, the slower its height growth.

KEY CHAPTER TAKEAWAYS

- **The observed accumulation is named *stack*.**
- **Based on these experiments, specific trends for the stack length or height are hard to identify. The stack length growth is mainly due to the progression of the head.**
- **These experiments have produced reference measurements. To use them as calibration reference (Chapter 4), the MPM is first studied through a sensitivity analysis presented in the next Chapter.**

3 | Sensitivity Analysis

KEY CHAPTER FEATURES

- **The simulation setup and its special procedure**
- **The full parametrization of the MPM and its reference values**
- **Two parameter variation approaches for behavior and sensitivity assessment**

The aim of this section is to obtain a first overview of the stack morphology descriptors' response to the different parameters of the MPM model.
First it presents the simulation setup (Section 3.1) and reference values (Section 3.2). Then, each parameter of the constitutive model is analysed. In a first step, parameters are analysed one by one, fixing the values of all other parameters using a single parameter sensitivity index (Section 3.3). This yields a quantitative comparison between all parameters. In a second step, parameters are analysed in the opposite way. Each parameter is fixed on some value, one by one, letting all other parameters vary (Section 3.4). For this last analysis, the yield criterion parameters (M, P_0, β) and the hardening law parameters (ξ,E) are analysed separately. Finally, parameter importance is discussed with respect to the calibration that follows in Section 4.
Refer to Annex A.3 for a complete description of the developed workflow and interplay between the MPM algorithm and the external editing tool (Houdini). The extraction of the simulated stack morphology characteristics requires some post-processing detailed in Annex A.4.

3.1 Simulation setup

The simulation geometry setup is presented in Figure 3.1.1. Simulations were run in 3D. All important information regarding the sensitivity analysis simulations parameters are grouped in Table 3.2.1.

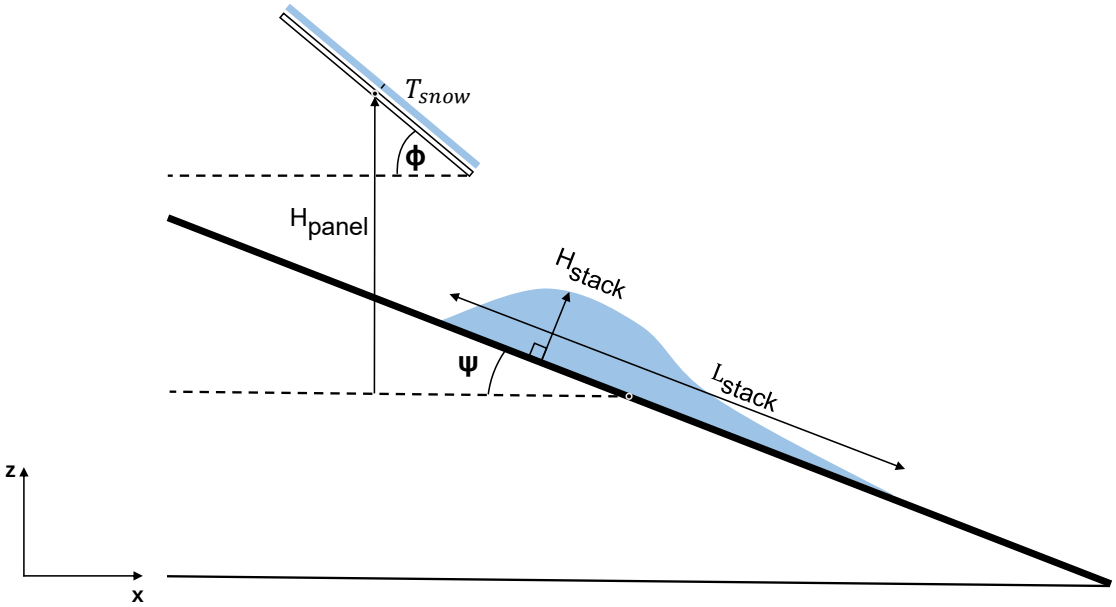

Figure 3.1.1: Side view of simulations setup. The key geometrical parameters are the panel height (H_{panel}), the panel tilt (ϕ), the thickness of the snow layer on the panel (T_{snow}) and the ground tilt (Ψ).

3.1.1 Multi-events loops

The multi-events workflow developed is illustrated in Figure 3.1.2. In the special case of 3 slide-off events, the step after the first slide-off is called STEP 0, the second STEP 1 and the last STEP 2. The steps names will be used in the sensitivity analysis.

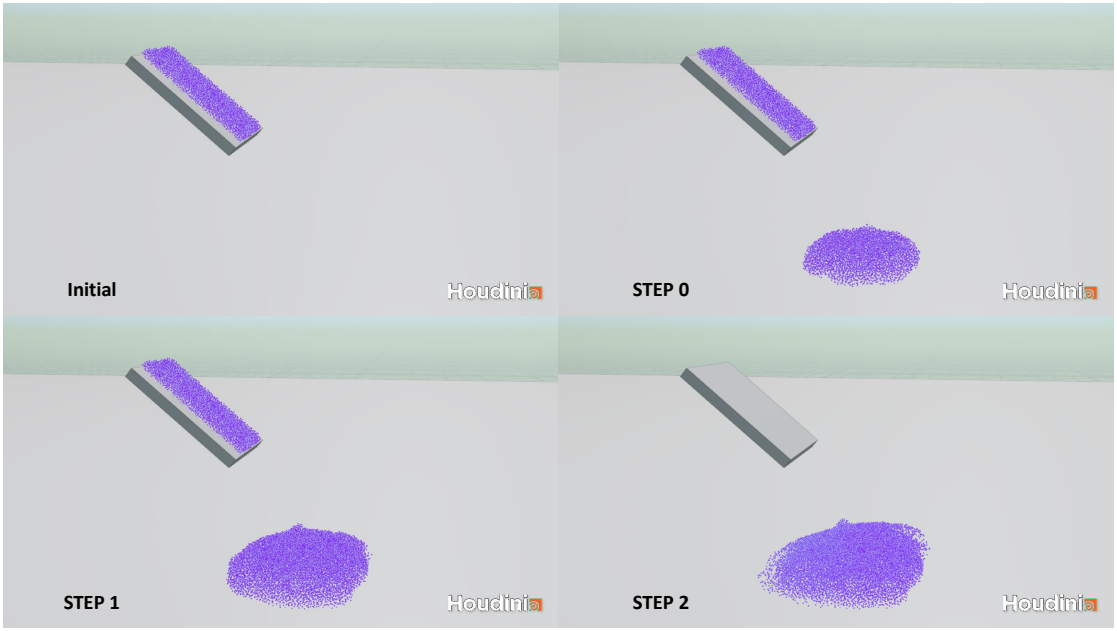

Figure 3.1.2: Illustration of the multi-events nature of the simulations. Here 3 slide-off events are simulated.

3.1.2 Grid size

Because the voxel size of the material VDB is set equal to the grid size (Section 1), changing the grid size (dx) also modifies the number of particles the material is discretized into. Note that [11] showed the grid size doesn't significantly affect the results of simulations.

3.1.3 Boundary conditions

Different boundary conditions are used for the two collision objects of the simulation domain. The panel is given a SLIP boundary condition to allow the whole slab sliding and to account for the force of friction between the snow and the panel. The ground is given a STICKY boundary condition. This means the actual friction between the ground and the snow falling off the panel is considered to be infinite. This assumption may not be exact but its total impact is small as the snow internal friction is still considered and is probably smaller than friction on a rough surface. Thus, internal friction takes over as soon as a first layer of snow on ground is present.

3.2 Reference values

Chosen initial values for each snow parameter are taken from literature for a dry, poorly cohesive, small scale slab. These characteristics are those of the snow used in the experiments (Section 2). This choice makes the first results of the sensitivity analysis comparable and easier to judge.

The Young's modulus E, the Poisson's ratio ν, the friction coefficient M and the hardening factor ξ are taken as weak layer values from [8]. The tensile strength $\sigma = \beta p_0$ is taken from [17] which grouped multiple studies results of tensile strength as a function of snow density (Figure 3.2.1). Accounting for the cohesion and scale properties of the simulated slab, the smallest values within the range corresponding to the density of 210 kg/m^3 are considered. Finally, the cohesion coefficient β is chosen small and comparable to values for the weak layer in [8] or to values of cold dense and warm shear avalanche regime in [11]. This fixes the values of p_0 using σ.

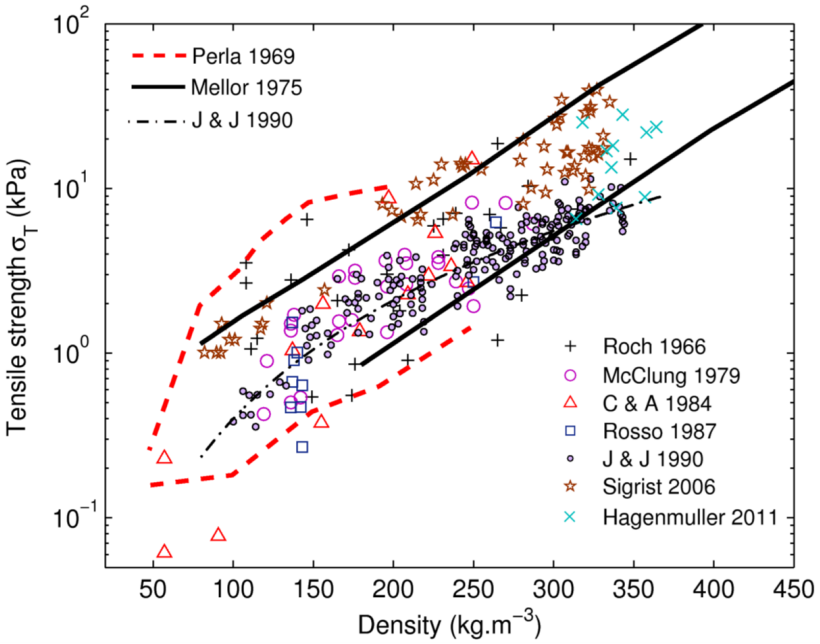

Figure 3.2.1: Taken from [17], plot of multiple snow tensile strength measurements as function of snow density.

Table 3.2.1: All parameters and their values adopted for the single parameter fixation sensitivity analysis (Section 3.4) and partially in the single parameter variation sensitivity analysis (Section 3.3). Reference values are in bold. The scale of the panel is chosen to fit the experiment (Section 2) in order to facilitate the calibration (Section 4). The panel is set 1.5 times wider than the snow slab to prevent particles from falling on the sides of the panel. The grid size was chosen half the flow thickness T_{snow}.

Geometry			
Panel Height	H_{panel}	0.5	m
Panel Length	L_{panel}	0.245	m
Panel Width	W_{panel}	0.213	m
Panel Tilt	Φ	40	°
Ground Tilt	Ψ	0	°
Snow			
Slab Thickness	T_{snow}	0.02	m
Slab Length	L_{snow}	0.245	m
Slab Width	W_{snow}	0.142	m
Density	ρ	210	kg/m^3
Young's modulus	E	**1**-2-3	MPa
Poisson's ratio	ν	**0.3**	-
Internal friction	M	0.3-**0.5**-0.7	-
Preconsolidation pressure	P_0	2-**5**-8	kPa
Hardening factor	ξ	**0.1**-0.2-1	-
Cohesion factor	β	**0.1**-0.2-0.3	-
Numerical setup			
Grid size	dx	0.01	m
Panel boundary condition	BC_{panel}	SLIP	-
Ground boundary condition	BC_{ground}	STICKY	-
Number of frames	N_{frames}	50	-
Number of loops	N_{loops}	3	-

3.3 Single parameter variation

The single parameter sensitivity index [18] is computed as

$$S = \frac{(R_a - R_n)/R_n}{(P_a - P_n)/P_n} \tag{3.3.1}$$

where R_a, R_n are the altered and nominal model outputs and P_a, P_n are the altered and nominal parameter values respectively. In this case R takes the value of the stack's height (H_{stack}) or length (L_{stack}) after 3 slide-off events. The parameter values are presented in Table 3.3.1.
Note that the S index, as used here, provides with a relatively local quantity only. Changes of behavior may appear non-linearly over the range of each parameter, but are not considered by the S index. However, values are chosen around the expected calibrated values (Section 4). It consists here in a first quantitative illustration of how the model behaves around these values. The altered values are chosen the double of the nominal value (Table 3.2.1).
Scores of computed S index for the stack height and length are presented in Figure 3.3.1.

Table 3.3.1: Nominal and altered adopted values for the computation of the S index. These are the values taken by each parameter when analysed. When fixed, a parameter takes its reference value (Table 3.2.1).

Yield criterion		P_n	P_a	
Internal friction	M	0.3	0.6	-
Preconsolidation pressure	p_0	3	6	kPa
Cohesion factor	β	0.15	0.3	-
Hardening law		P_n	P_a	
Hardening factor	ξ	0.1	0.2	-
Young's modulus	E	1	2	MPa

Stack height

On the ranges analysed, the cohesion factor β shows the strongest impact on height. It is proportional and increasing with the addition of mass (STEP). The preconsolidation pressure p_0 shows a constant, rather small, inversely proportional impact. The internal friction M has an impact at STEP 2 but otherwise is very small. For the hardening parameter ξ and the Young's modulus E, almost no impact is detected.

Stack length

Whatever the parameter, the resulting S-index has a lower absolute value. This means the impact of the snow parameters, in the analysed ranges, is smaller. The internal friction M

and the cohesion factor β are the only parameters showing a distinctive impact. It is overall inversely proportional with an exception for β at STEP 0. Their impact increases with the addition of mass. The Young's modulus growth is distinguishable but still very small in comparison with the two previous ones.

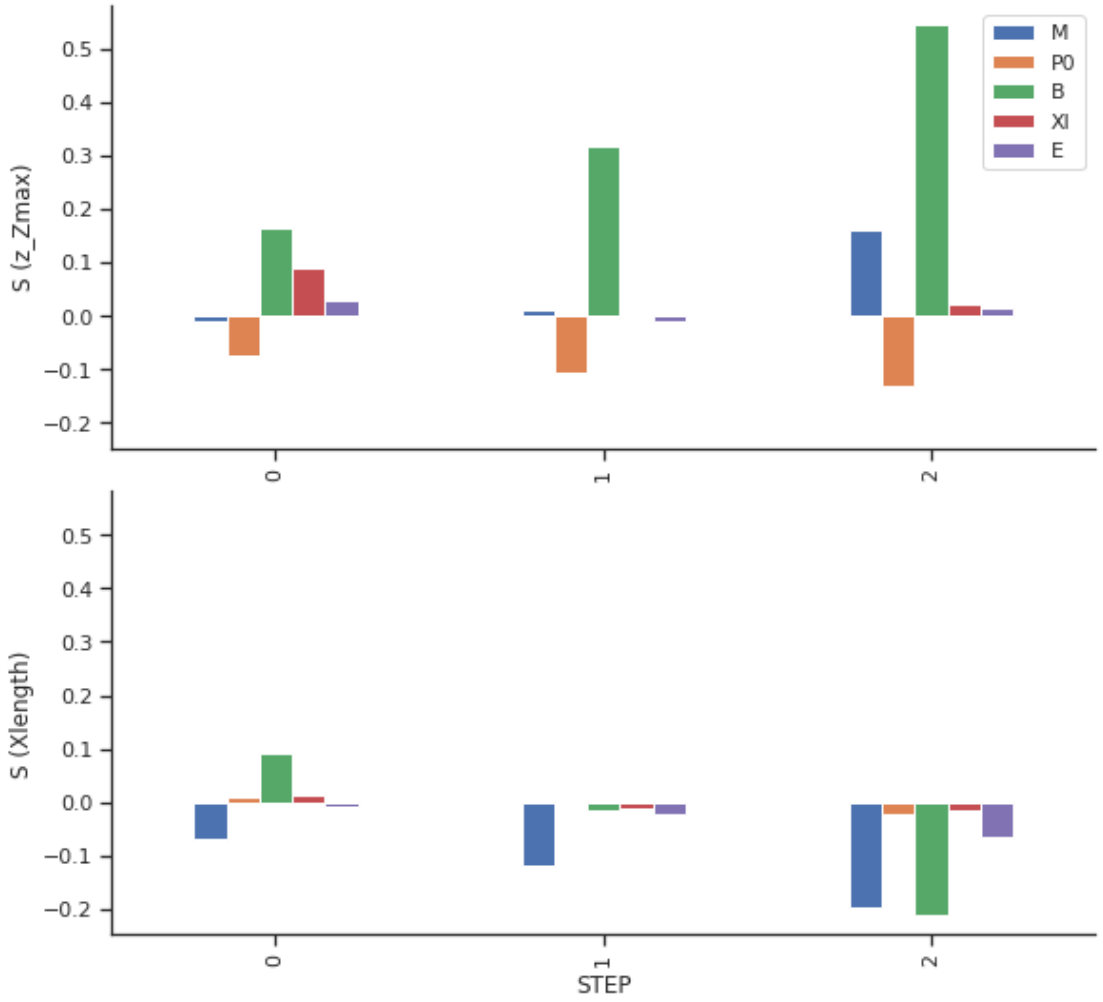

Figure 3.3.1: S-index scores. β and p_0 show the most consistent impact on the stack height. For its length, M has the most consistent impact.

3.4 Single parameter fixation

In this section, all line plots show simulations grouped by the number of slide-off STEP. The lines represent the mean value of each group and the envelope the 95% confidence interval (1σ). The distribution comes from all combinations simulated for a given fixed parameter value. For instance, taking M=0.5, group STEP=2, the line and the envelope aggregate all results of the following 9 simulation setups:

$$M \times p_0 \times \beta = [0.5] \times [4000, 6000, 8000] \times [0.1, 0.2, 0.3]$$

The constitutive model and hardening law parameters are however analysed separately, meaning the latter ones are fixed when the former are alternatively varied and *vice-versa*. This means, for instance, taking ξ=1, group STEP=2, the line and the envelope aggregate all results of the following 3 simulation setups:

$$\xi \times E = [0.5] \times [1e6, 2e6, 3e6]$$

The variables *z_Zmax* (stack height) and *Xlength* (stack length along axis X) are normalized using rescaling

$$x' = \frac{x - min(x)}{max(x) - min(x)} \tag{3.4.1}$$

where, x is the original variable and x' is the rescaled version. Variables are thus modified to span from 0 to 1 within their full range.

3.4.1 Yield criterion parameters

In the following simulation results, hardening law parameters ξ and E are kept constant, equal to their reference value (Table 3.2.1). The most relevant observations are repeated in the figures caption.

Stack height

The height sensitivity to yield criterion parameters is presented in Figure 3.4.1. Parameters M and p_0 show similar effects. The higher their value the higher the stack. Differences in slopes between the two parameters are minimal. However the inter group difference is notable. The group STEP 1 shows a somewhat steeper slope than group STEP 0 and group STEP 2 shows a clearly steeper slope than group STEP 1 in its range lower half. It is less clear in the upper half. The width of the envelope grows proportionally to the addition of mass, for both parameters similarly again. This witnesses that the height gets more sensitive to the yield criterion parameters towards high number of slide-offs.

The cohesion factor β has overall the same trend. The higher its value, the higher the stack. Its range lower half is however almost insensitive to other parameters regardless to the addition of mass. The three groups have similar slopes and similar envelope size. This means that for its lower analysed values, β, is driving the stack height. The second half yet shows a similar trend as for M and p_0. The envelope and the slope strongly grow in group STEP 2.

An important trend common to all three parameters is thus the growth of the envelope from low values of each parameters towards higher when enough snow mass is involved (STEP 2) and when a certain threshold is reached. In fact a small M means little internal friction, a low p_0 means little elasticity and a low β means, in combination with p_0, a small tensile strength. It can thus be observed that the less cohesive the snow, the less sensitivity towards the snow parameters. This goes together with a drastic decrease of the height value. Conversely, the more cohesive the snow, the higher the stack but also the higher the sensitivity to each parameter variation. In terms of usual snow characteristics, this means that, for fresh, cold and dry snow conditions, the model is less sensitive to small parameters variation in general. On the contrary, in warmer and more compact snow conditions, the stack's features are more sensitive to the snow parameters.

It is again worth recalling the importance of mass addition witnessed by having almost no overlap between the envelopes whatever their size.

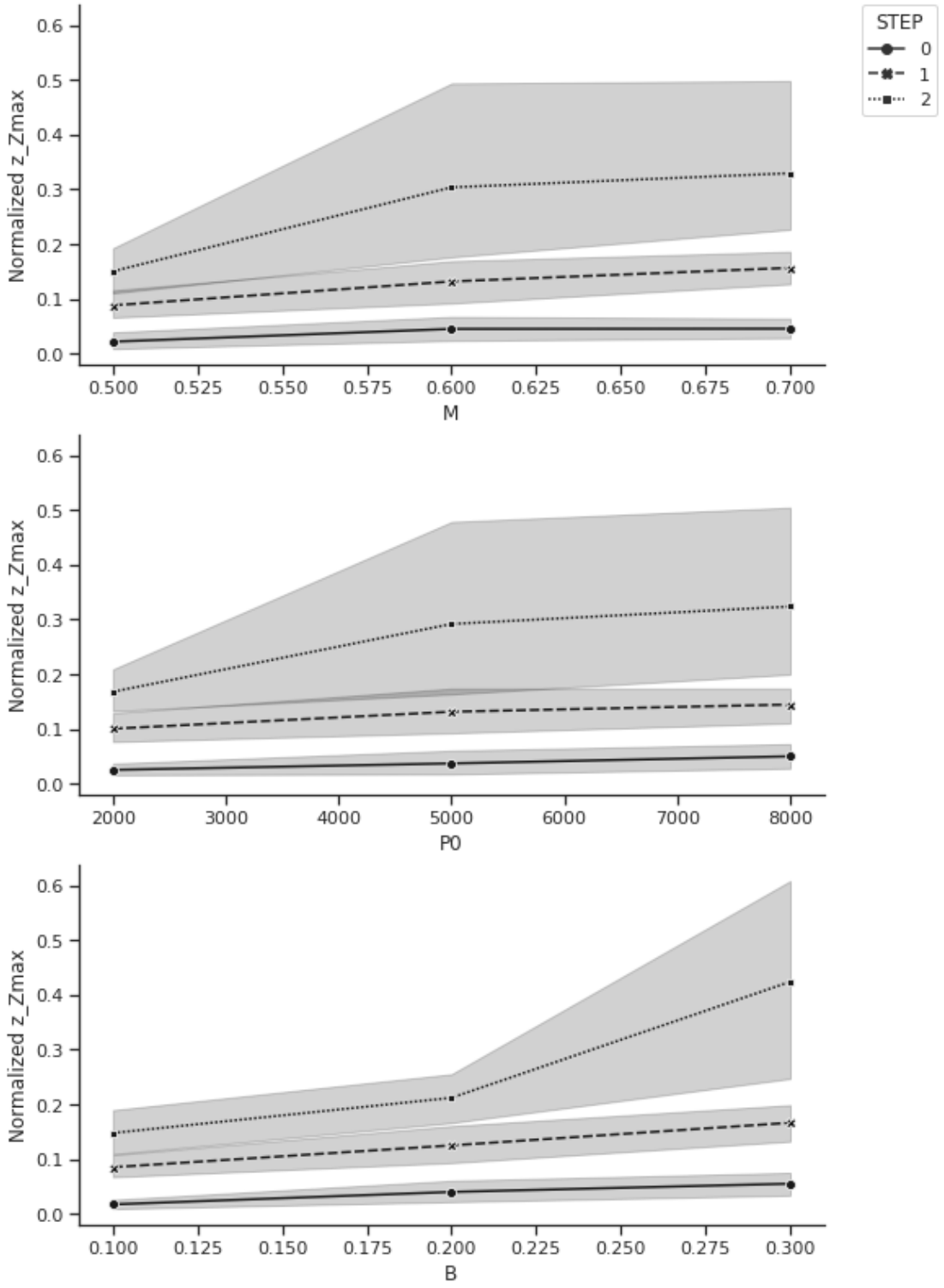

Figure 3.4.1: Stack height sensitivity to yield criterion parameters M (internal friction), p_0 (preconsolidation pressure) and β (cohesion factor). The stack height is proportional to each parameter's value. The less cohesive the snow, the lower the height, the lower the sensitivity to yield criterion parameters and the lower the sensitivity to addition of mass.

Stack length

The length sensitivity to yield criterion parameters is presented in Figure 3.4.2. To some extent[1], it is expected that the stack length shows the opposite sensitivity to M, p_0 and β. A less cohesive snow showed a smaller height and thus should show a larger length. This is verified at least for M and β. The preconsolidaiton pressure p_0 shows a mixed behavior with a flattening of the curve in group STEP 2 but still has positive slopes for the two other groups. While a p_0 can be associated to snow cohesion through the tensile strength ($\sigma = \beta p_0$), it is first of all a measure of the initial snow elasticity. The overall proportionality between the stack length and p_0 suggest that the more elastic the initial state of snow, the less energy is dissipated in plastic deformation and the more kinetic energy each particle keeps. Interestingly, the range lower half gets flatter with increase of mass (STEP 2). A potential understanding of this is that when enough snow is present enough energy can be dissipated in plastic deformation stopping the increase in length.

The trend in size of the envelopes are all similar across the parameters. They grow with addition of mass but to a lesser degree than for the height. Their variation as function of the parameter alone and even mixed to snow height remains very noisy. There is seemingly no driving parameter.

The addition of mass shows a peculiar behavior in comparison to the height. The gap between group STEP 0 and group STEP 1 is larger. In addition, the gap between group STEP 1 and group STEP 2 is smaller, leading to larger overlap. The applied STICKY boundary condition for the ground seem to play a role here. The first slide-off (STEP 0) sticks to the ground and doesn't have enough material to slide on its own snow compared to the two other groups. It is interesting to note that, for some very cohesive parameters combinations, the length doesn't grow much after the second slide-off (STEP 1).

[1] Snow is a plastic material. It therefore doesn't necessarily have a constant volume under mechanical constraint. However, its deformation does include an elastic component. The expectation on the stack height-length relation are hence limited to some extent. Moreover, settling and hardening of snow over time are both unaddressed processes which may as well impact the stack height-length relation.

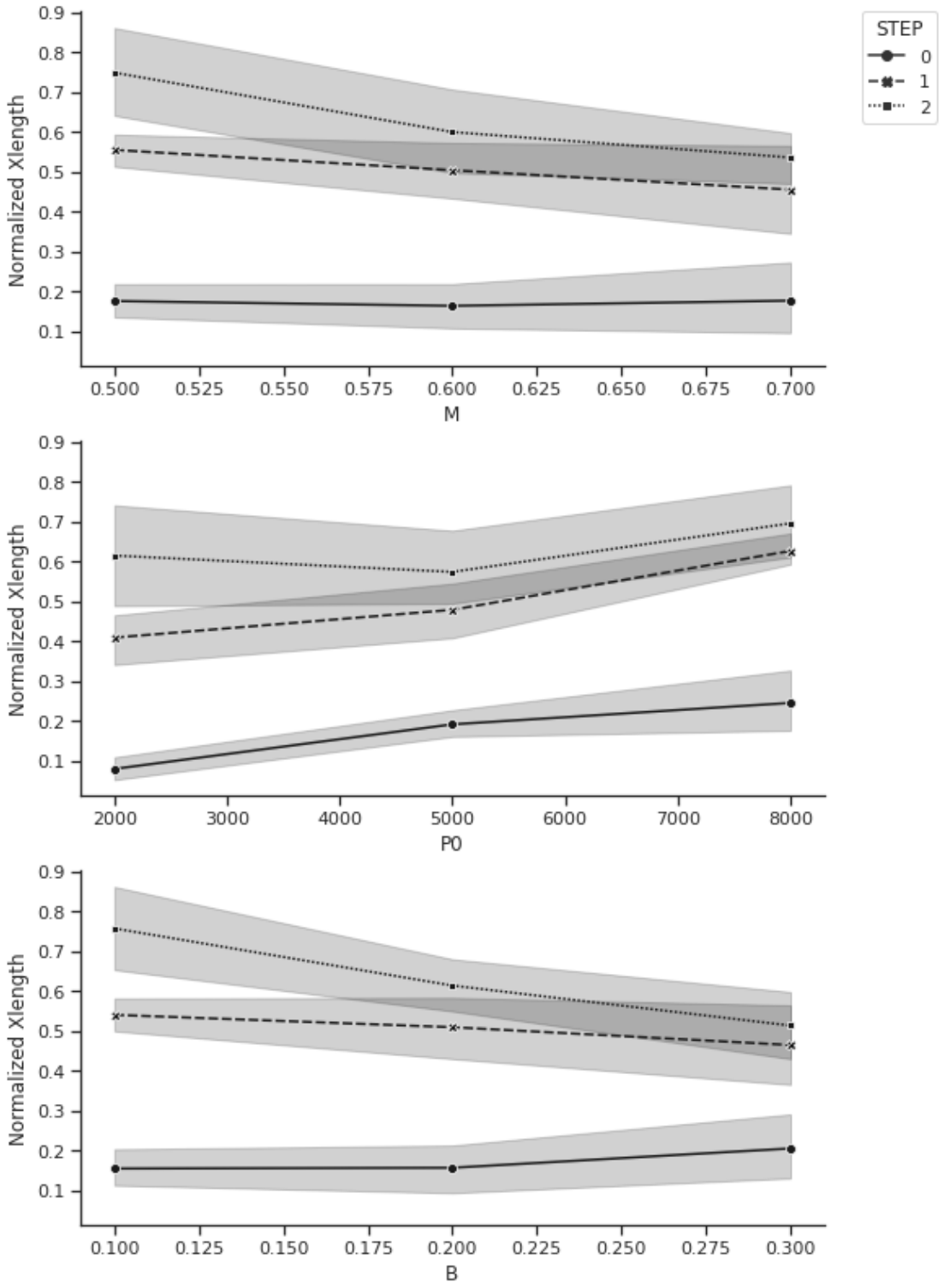

Figure 3.4.2: Stack length sensitivity to yield criterion parameters M (internal friction), p_0 (preconsolidation pressure) and β (cohesion factor). The stack length is inversely proportional to M and β values. For these two parameters a low snow cohesion combination results in a larger length. The preconsolidation pressure p_0 on the contrary shows a proportional increase. This behavior seems however affected and changing for the lowest part of the analysed range when enough snow mass is involved. Plastic deformation energy dissipation is an explicating hypothesis. The addition of mass has a great impact especially between groups STEP 0 and STEP 1, then less. The choice of a STICKY boundary condition probably plays an important role. The large overlap between groups STEP 1 and STEP 2 shows the addition of mass is at some point not as impactful as for the height.

3.4.2 Hardening law parameters

In the following simulation results, yield criterion parameters M, p_0 and β are kept constant, equal to their reference value (Table 3.2.1).

Stack height

The height sensitivity to hardening law parameters is presented in Figure 3.4.3. Both the hardening factor ξ and the Young's modulus E show linear behavior. Group STEP 2 for xi is the only exception. For this group there is an inversely proportional relation with the stack height.

The size of the envelopes are very small for groups STEP 0 and STEP 1 for both parameters. Group STEP 2 shows a larger sensitivity. For E, this envelopes size is constant while for ξ it is decreasing for higher values. The range upper third of ξ suggests it is the driving parameter with respect to E. However its constancy reduces its final impact.

The addition of mass is clearer than anything seen before. The space between the groups is large, constant and similar between the two parameters.

The narrow envelopes, the constancy and the clear impact of mass addition are signs of the lower sensitivity of the stack height towards hardening law parameters in comparison with the yield criterion ones. Also, note the much larger range chosen for xi which values are multiplied by 100 between the lowest and highest boundaries.

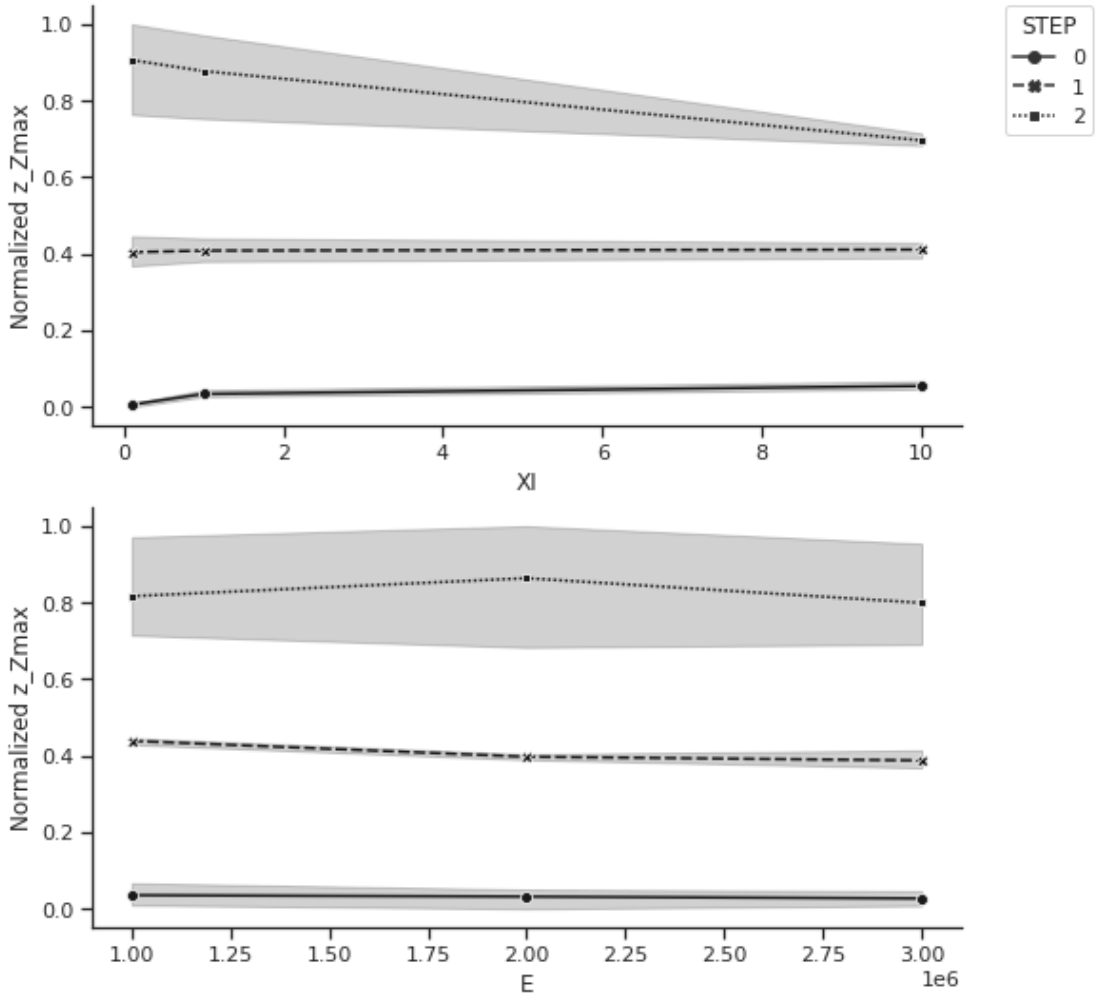

Figure 3.4.3: Stack height sensitivity to hardening law parameters ξ (hardening factor), E (Young's modulus). Constancy and the envelope width of E in group STEP 2 shows the insensitivity of the stack height towards this parameter. There is overall no clear effect of both parameter although ξ seems somewhat stronger with a stepper slope and narrower envelope in group STEP 2. The mass addition impact is very clear and constant.

Stack length

The length sensitivity to hardening law parameters is presented in Figure 3.4.4. A clear difference between ξ and E stands in the envelope size. The Young's modulus E has much wider envelopes overall. It is moreover rather constant over the whole range. The hardening factor, on the contrary, has very narrow envelopes for groups STEP 0 and STEP 1, showing its insensitivity towards E. The group STEP 2 which involves more mass, however has a much wider envelope. In its range lowest part and for all groups, xi shows a quite steep slope inversely proportionally impacting the length.

The addition of mass seems to have the same effect on the stack length as for the yield criterion sensitivity analysis. This reinforces first the assumption of the strong influence of mass involved and second the important role of the boundary condition. The pattern of a very large gap between groups STEP 0 and STEP 1 and a much smaller one between groups STEP 1 and STEP 2, supports this statement.

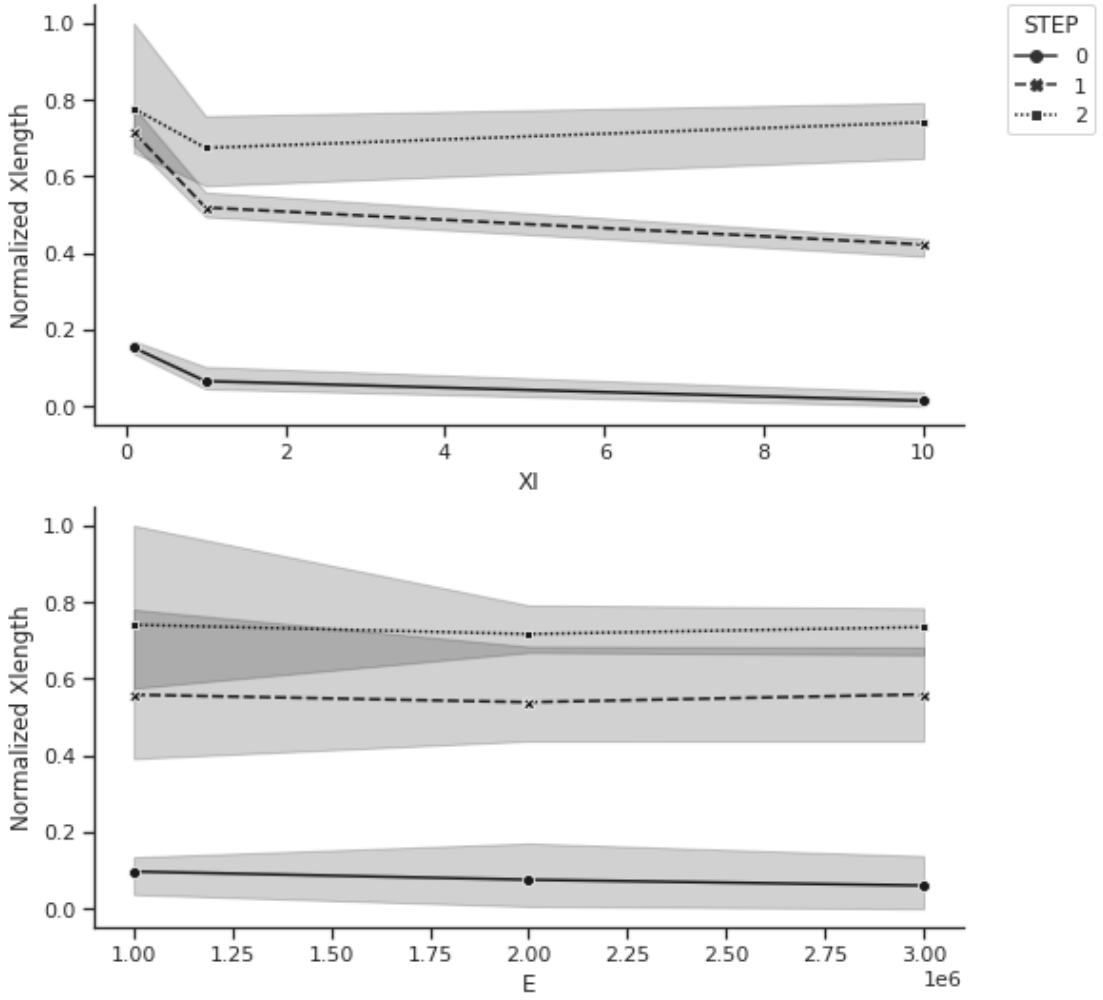

Figure 3.4.4: Stack length sensitivity to hardening law parameters ξ (hardening factor), E (Young's modulus). A clear difference in the width of envelopes suggests E has little impact. The constancy of E in all groups supports this statement. In its range lowest part, xi impacts the stack length. Mass addition shows the same impact as for the yield criterion parameters and support its first order role as well as the role of the STICKY boundary condition.

3.5 Snow parameter set reduction

Based on the previous two sections, the hardening law parameters will be fixed and not considered in the calibration. The overall smaller sensitivity towards ξ and E of, primarily, the stack height and, secondarily, its length justifies this choice. It is considered that the reported impact of ξ on the stack length will be mitigated by the yield criterion parameters. Also the fixed values of ξ and E being their reference values (Table 3.2.1), the hardening law parameters combination will maximize the stack length. Note that the small scale of the simulations may have an impact on the small implication of snow hardening. In the following, this is however not considered for simplification reasons.

KEY CHAPTER TAKEAWAYS

- **Small M and β overall produce smaller but longer stacks and inversely for large values. p_0 follows the same trend for the stack's height but has a different behavior for the length. This is probably due to its fundamental relation to the snow elasticity.**
- **More cohesive snow yield criterion parameters combinations lead to a higher sensitivity of the stack's height towards those parameters. This is not verified for the length.**
- **For β values below 0.2 and whatever the mass of snow involved, this parameter has a driving effect on the stack height.**
- **The mass of snow involved has a direct impact on both the length and height of the stack. It also overall confers a larger sensitivity towards snow parameters, especially for the height.**
- **Hardening law parameters are dropped for their identified low impact and to simplify the following steps. This might however not be the case for larger scale simulations.**
- **Most important parameters have been identified and will now be calibrated on experiments in the next Chapter.**

4 | Calibration

KEY CHAPTER FEATURES

- **The calibrated model parametrization and its performance with respect to the experiments**

Based on the sensitivity analysis results, the yield criterion parameters M, p_0 and β are calibrated on the experiments. On the path to the statistical model of Chapter 5, this calibration fixes the model snow parameters. In other words, it confers the model a specific type of snow. The model is therefore realistic but its generality is also reduced.

4.1 Workflow

The best combination selection is based on the mean rescaled Root Mean Square Error (RMSE) combining all steps of each experiment (40°, 50°). The same rescaling as used for the sensitivity analysis (equation 3.3.1) is applied to the RMSE scores to combine the two features of the stack and select a combination. Figure 4.2.1 shows the relative errors for each stack feature and the different RMSE scores are presented in Figure 4.2.2. RMSE statistics can be found in Table 4.2.1. Scores of the calibrated model is provided in Table 4.3.1.

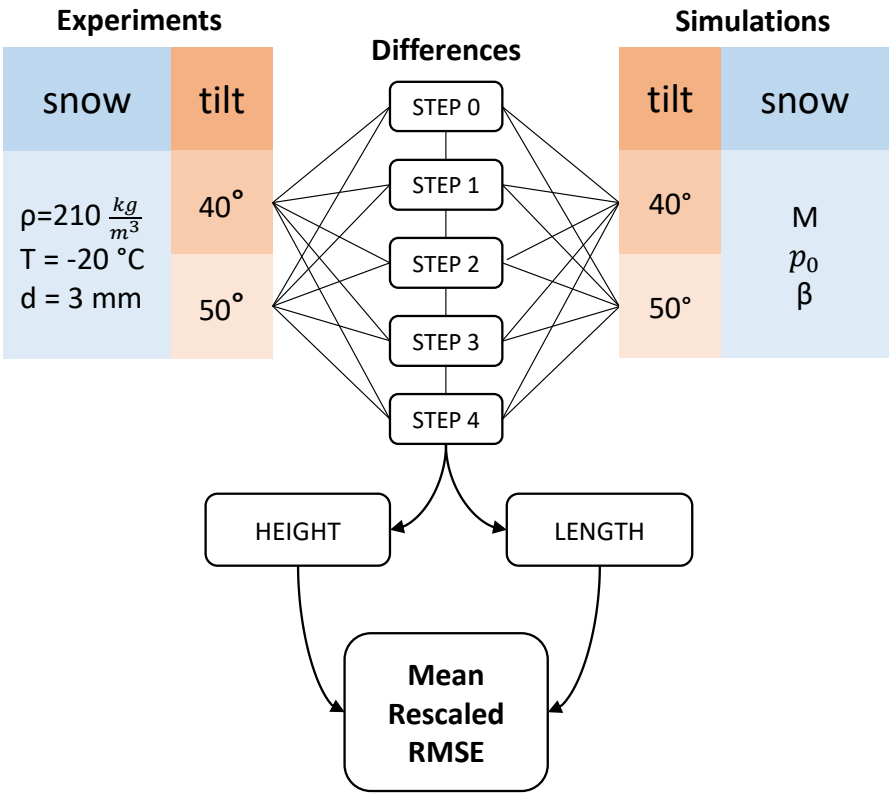

Figure 4.1.1: Calibration workflow chart. Parametrized simulations are compared (difference) to experiments measurements. While geometrical characteristics of the experiments can be perfectly simulated (tilt), snow characteristics relation is not known (M, p_0, β). Considering both the length and height of the stack, the calibration aims at finding the best snow parameter M-p_0-β triplet for the snow used in the experiments (mean rescaled RMSE). This snow is characterized by its density (ρ), its temperature (T) and the sieve grid size (d).

Table 4.1.1: Parameters used for calibration. Ranges are limited around the reference values for computational cost reasons and because, based on experience, the best combination is expected within these ranges.

Geometrical parameters			
	H_{panel}	0.33	m
	L_{panel}	0.245	m
	W_{panel}	0.213	m
	Φ	40-50	°
	Ψ	0	°
	T_{snow}	0.02	m
Snow parameters			
	ρ	210	kg/m^3
	E	1	MPa
	ν	0.3	-
	M	0.4-0.5-0.6-0.7	-
	P_0	4-5-6-7	kPa
	ξ	0.1	-
	β	0.05-0.1-0.2	-
Numerical setup			
Grid size	dx	0.01	m
Panel boundary condition	BC_{panel}	SLIP	-
Ground boundary condition	BC_{ground}	STICKY	-
Number of frames	N_{frames}	60	-
Number of loops	N_{loops}	5	-

4.2 Results

The differences between simulations and experiments are hereafter presented through different scores. All aggregate steps and tilt configurations for a given parameter set trial. While the mean relative error (Figure 4.2.1) gives a good view of the actual performance, it is more sensitive to outliers. The RMSE (Figure 4.2.2), on the other hand, gives a good feeling of the overall accuracy of the trial. Finally, the rescaled version of the RMSE allows to combine both the height and length of the stack.

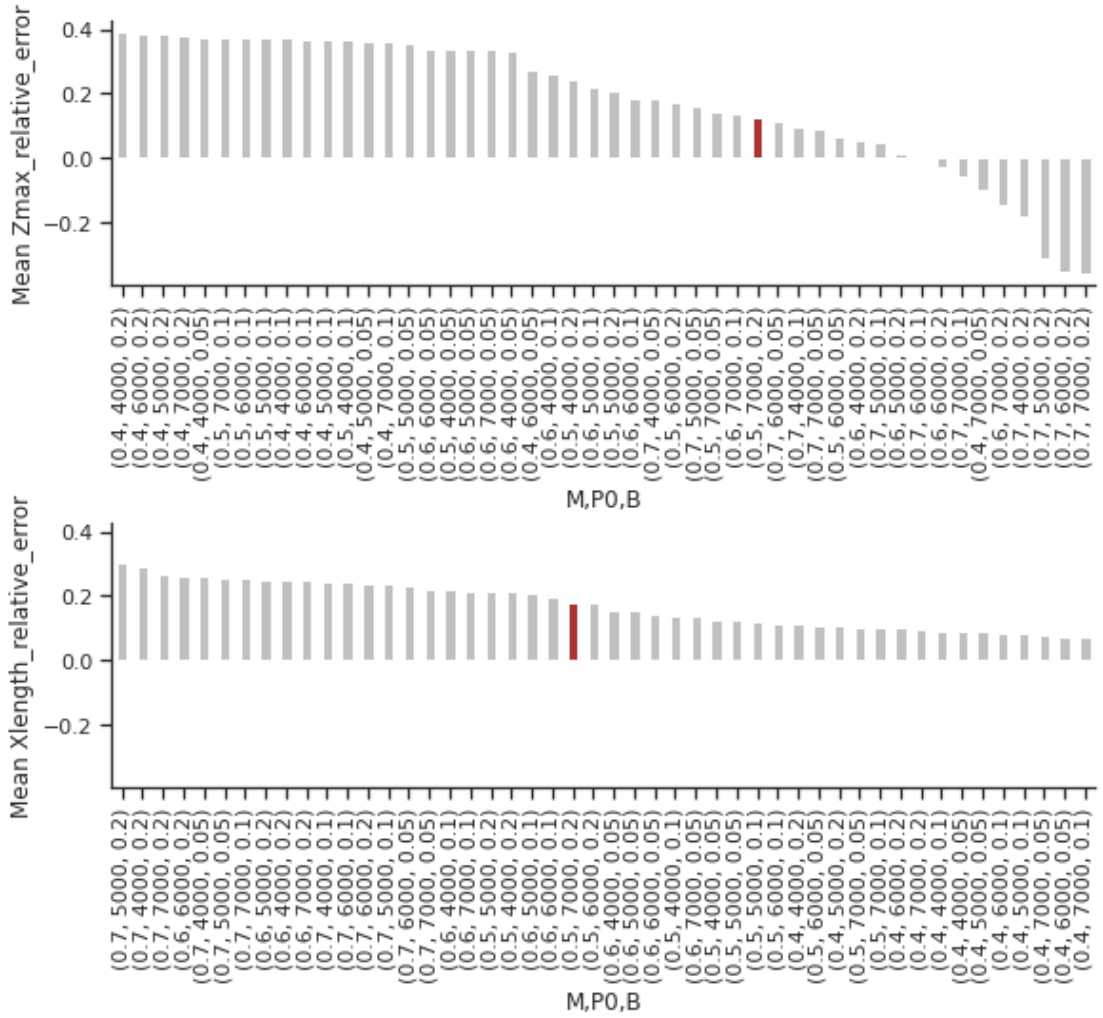

Figure 4.2.1: Mean experiment relative error for stack height (top) and length (bottom). The difference between the experiment and the simulation is divided by the experiment result. The aggregation (mean) is done on each tilt and each step of the simulation/experiment comparison to yield only one value per parameter triplet. For the height, scores close to 0 were produced. For the length the best score stands close to 0.1 (10%). Note the sensitivity to outliers of the mean over the 5 slide-off events. It however combines well with the RMSE scores presented in Figure 4.2.2. The group with the best RMSE score is higlighted in red. The absolute values of the stack features were introduced in Section 2.

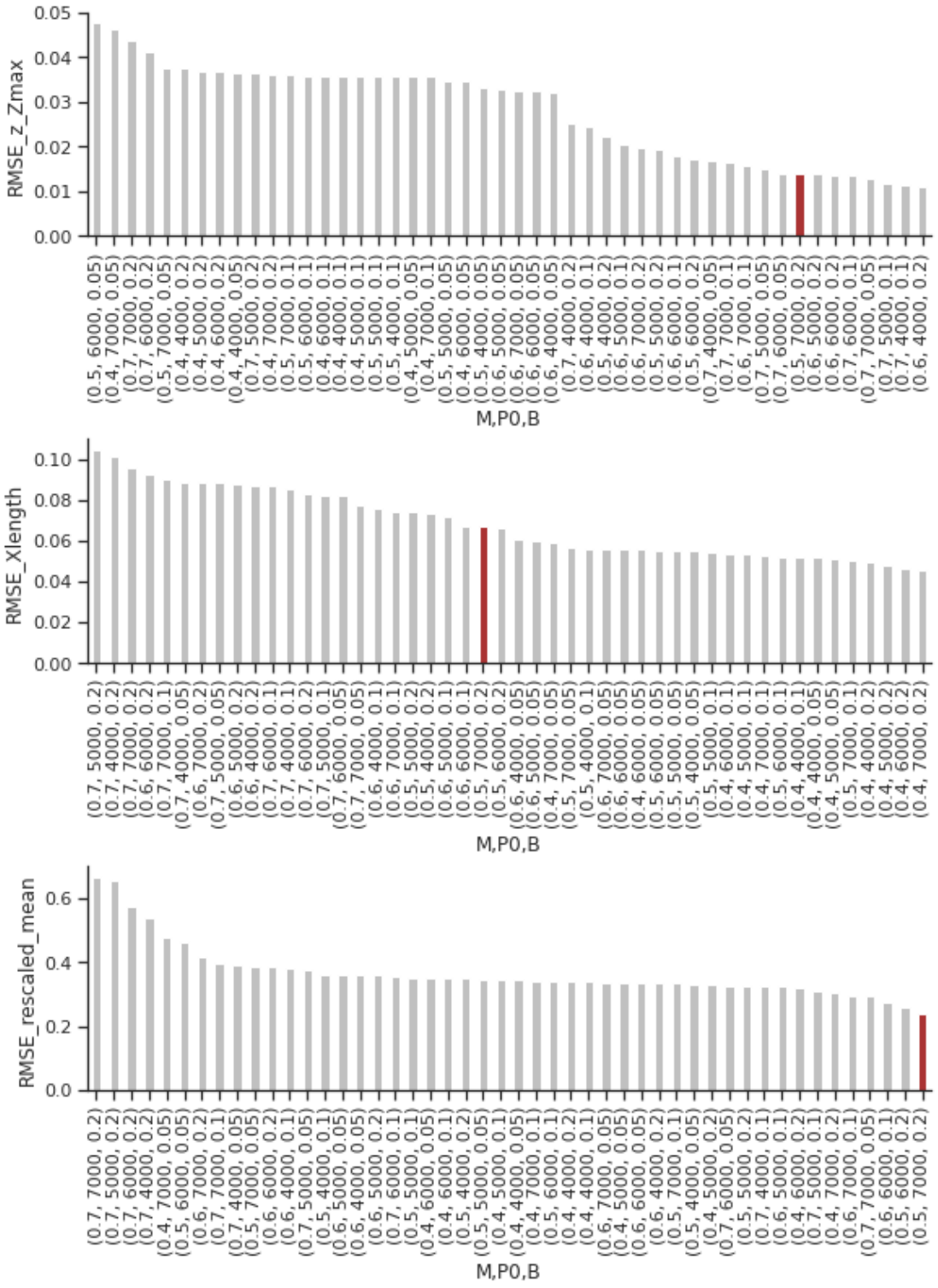

Figure 4.2.2: RMSE scores for the different M-p_0-β combinations for the stack height (top), length (middle) and the aggregation of both using the rescaled mean (bottom). The best combination in average is ranked 8 for the height and 26 for the length on a total of 48 combinations. Refer to table 4.2.1 for complementary information on the rescaling.

Table 4.2.1: Statistics of the height and length RMSE use to form the rescaled RMSE.

Feature	min	max	mean	median
RMSE Height [m]	0.009	0.067	0.028	0.029
RMSE Length [m]	0.034	0.118	0.068	0.063

4.3 Calibrated model

The calibrated model takes the parameter triplet with the lowest mean rescaled RMSE. The full parametrization and its different scores is summarized in Table 4.3.1.

Table 4.3.1: The calibrated model parametrization and its respective scores. It has a height and length mean error of 1cm and 6cm respectively over the 5 steps. The similar results for the respective RMSE scores shows there is no particular outliers.

Parameter set	M-p_0-β-ξ-E
Calibrated values	0.5-7e3-0.2-0.1-1e6
Height mean error [m]	0.010
Length mean error [m]	0.061
Height RMSE [m]	0.014
Length RMSE [m]	0.067
RMSE rescaled mean	0.237

KEY CHAPTER TAKEAWAYS

- **The mean errors lie around 10% and 20% for the stack height and length respectively.**
- **The only change between the reference and calibrated values lies in the preconsolidation pressure p_0. This emphasizes its impact on the snow stack.**
- **The calibrated model can now be used to elaborate the statistical model in the following Chapter.**

5 | Statistical model

KEY CHAPTER FEATURES

- **A sensitivity analysis of the panel geometry parameters**
- **The statistical model theoretical background and training procedure**
- **The performance comparison between a linear and non-linear approach**
- **A validation analysis for both models**

In this Chapter, snow calibrated MPM simulations are first used to build and evaluate the potential of a statistical model to predict the height and length of the stack. As mentioned in the previous Chapter, the model's input don't include snow relative data. It includes panel-geometry parameters only. The Chapter is organised as follows. First, the simulations are described and a short sensitivity analysis is carried on the geometry parameters (Section 5.1) to understand their impact on the snow stack. Second, the statistical method is introduced as well as the data set preparation (Section 5.2). Finally, two different types of model each with their own advantages are trained and their performance compared (Section 5.3).

5.1 Geometry sensitivity analysis

5.1.1 Simulation setup

The simulation domain essentially has the same setup as for the snow sensitivity analysis (Figure 3.1), but they are computed in 2D[1]. The reason for this change is the assumption of infinity for the width of the panel/module. Indeed, applications at stake, such as solar power plants, are most of the time much wider than high for mechanical constraints. Moreover, as a rule of thumb, the 3D/2D simulation ratio for the stack height is about 2/3 when height and width are comparable. When the width can be considered infinitely larger, then one can expect the ratio to reach 1.

To first analyse impact of the panel geometry parameters, the same procedure as followed for the single parameter variation of the sensitivity analysis (Section 3.3) was adopted. The panel's length, height and tilt were varied in turns and more real values are introduced. The ground tilt was chosen to remain horizontal to simplify this first approach. When a parameter is varying, the others are set to some reference value. It is thus a particular case along the analysed parameter's axis, but as first approach it provides a good idea of the stack behavior. Table 5.1.1 groups all the information on this round of simulations.

[1] In the post processing presented in Annex A.4, the 2D methods were thus used.

Table 5.1.1: All parameters used in the panel geometry varying simulations. Snow parameters are taken from calibration. Only snow thickness is changed and increased to 5cm. Note that since simulations are 2D for this round, the panel and snow slab width are not provided.

Geometry			
Panel Height	H_{panel}	1-1.5-**2**-2.5-3-3.5-4	m
Panel Length	L_{panel}	0.5-1-1.5-**2**-2.5-3-3.5-4-6-8-10	m
Panel Width	W_{panel}	-	
Panel Tilt	Φ	**40**-45-50-55-60-65-70	°
Ground Tilt	Ψ	0	°
Snow			
Slab Thickness	T_{snow}	0.05	m
Slab Length	L_{snow}	0.5-1-1.5-**2**-2.5-3-3.5-4-6-8-10	m
Slab Width	W_{snow}	-	
Density	ρ	210	kg/m^3
Young's modulus	E	1	MPa
Poisson's ratio	ν	0.3	-
Internal friction	M	0.5	-
Preconsolidation pressure	P_0	7	kPa
Hardening factor	ξ	0.1	-
Cohesion factor	β	0.2	-
Numerical setup			
Grid size	dx	0.01	m
Panel boundary condition	BC_{panel}	SLIP	-
Ground boundary condition	BC_{ground}	STICKY	-
Number of frames	N_{frames}	100	-
Number of loops	N_{loops}	5	-

The analysed features of the stack are still its height and length. However, the ratio height/length is also computed and analysed. In fact, this type of non dimensional quantity is often preferred in modelling because it reduces the variations of multiple (here 2) response variables into a single one. Results are presented on Figure 5.1.1.

5.1.2 Single geometry parameter variation analysis

For the panel length (Figure 5.1.1, top), H/L shows a logarithmic behavior with a steep slope in lower values of the analysed range and a more linear and flat behavior in the higher values. The behavior is shifted and repeated for each group STEP. Looking at the length and height separately, it seems the length drives the strong logarithmic trend in the lower values. The height is also considerably growing. The panel length is in fact directly related to the mass of snow involved just as much as the grouping parameter STEP. As seen in the sensitivity analysis (Chapter 3), it is thus no surprise that PANEL_L has such an influence on the stack features.

The slopes of each of them support this statement.

The panel height plot (Figure 5.1.1, center) shows an interesting linear relation for H/L and for each group STEP with the same slope. Due to the increase of end-of-fall kinetic energy for larger panel heights, the spread and thus the length of the stack shows a strong proportional dependence. The height is also proportionally affected but to a lesser extend. This results in H/L inversely proportional to the panel height.

The panel tilt constantly shows less influence on the stack features and contains more noise. The length shows a slight increase with a steeper panel. The height and H/L are slightly decreasing with increasing tilt, while the length increases relatively strongly between 40° and 50° and is rather constant then. Overall the stack's features are noisier towards the panel tilt variation.

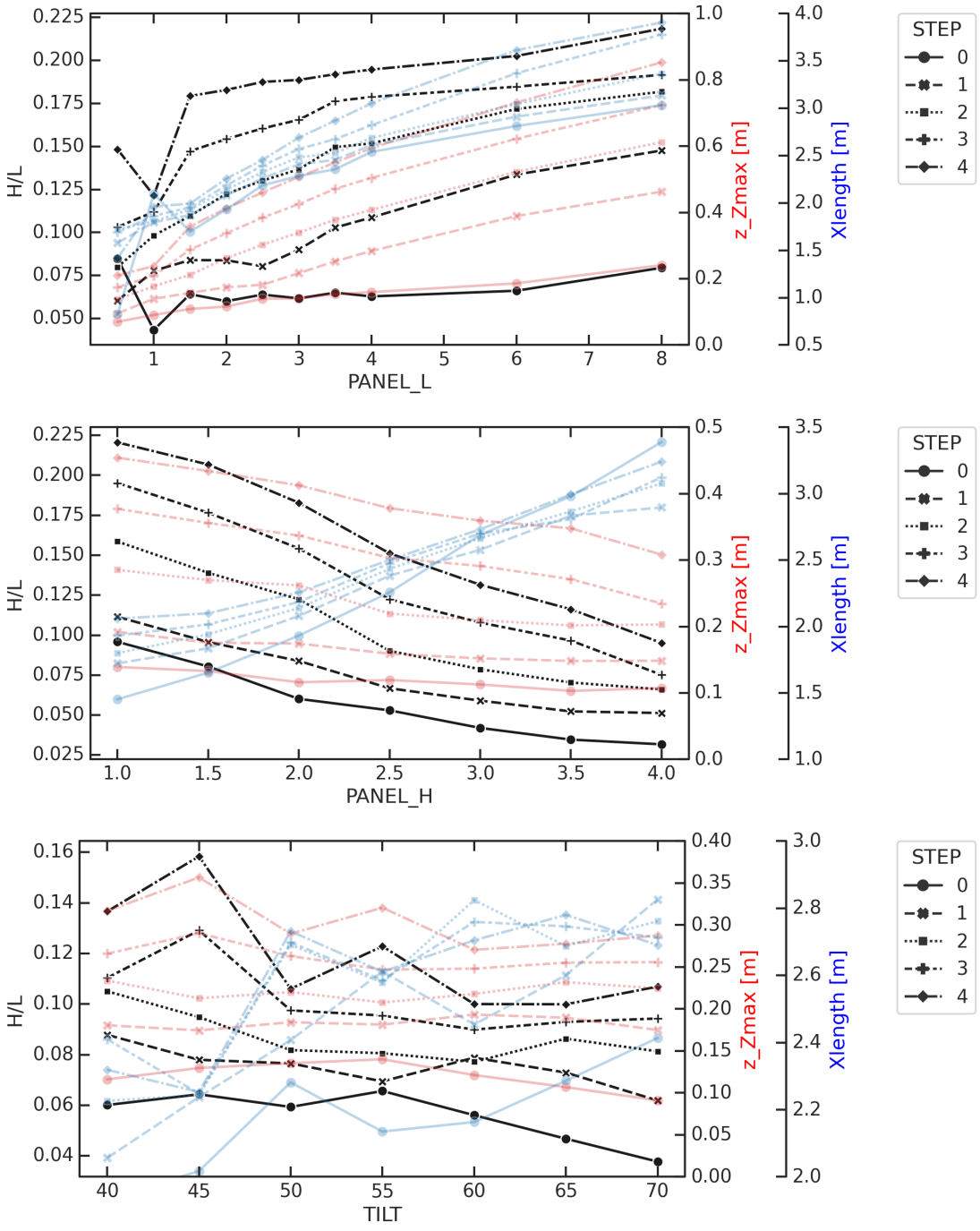

Figure 5.1.1: Single panel's geometry parameters variation. Panel length and height plots show two interesting behaviors, respectively logarithmic and linear, of the stack features and their combination (H/L). The panel tilt effect is more noisy and shows less effect. These plots provide with insights on the stack features behavior under geometrical variation of the panel. It is however only a partial view of that behavior.

5.1.3 Geometry parameter set reduction

Based on the previous section, the panel height and length seem most important and the tilt impact looks as a second order one. However, as previously said, the above partial sensitivity analysis doesn't provide a complete view of each parameter. Therefore and based on trial and error, the tilt is kept for the building of the statistical model.

5.2 Regression method

The non linearity and multivariate nature of the data set to be regressed, as well as its relative small size, directed the choice of the method towards the Support Vector Machine (SVM) [19] generalized for regression (ϵ-SVR). Among the family of supervised learning techniques, both for classification and regression purposes, it has been proved one of the most powerful method [20] in general. The following method details are inspired from [21].

5.2.1 Support Vector Regression principle

In its hard-margin form, the idea of ϵ-SVR is to find a function f as flat as possible that deviates at most by ϵ from the target values (y_i) of all the training data.
In the linear case, the function takes the following form

$$f(x) = \boldsymbol{w} \bullet \boldsymbol{x} + b \tag{5.2.1}$$

with $\boldsymbol{w} \in \mathbb{X}$, the space of input $\boldsymbol{x}$ and b $\in \mathbb{R}$. The optimization thus seeks the smallest $\boldsymbol{w}$ for flatness and makes sure the difference between y_i, the model's targets and $f(x_i)$, the predictions, is not larger than ϵ. Formally, it takes the form of a convex optimization problem

$$\text{minimize} \quad \frac{1}{2}||\boldsymbol{w}||^2 \tag{5.2.2}$$

$$\text{subject to} \quad \begin{cases} y_i - \boldsymbol{w}x_i - b \leq \epsilon \\ \boldsymbol{w}x_i + b - y_i \leq \epsilon \end{cases} \tag{5.2.3}$$

This ϵ strong requirement might however make the optimization insolvable. Therefore, the soft-margin version [22] introduces some tolerance towards deviations of f larger than ϵ. The parameter C is introduced to tune the trade-off between the flatness of f and the deviations over ϵ, through so-called slack variables (ξ_i). The optimization problem becomes

$$\text{minimize} \quad \frac{1}{2}||\boldsymbol{w}||^2 + \sum_{i=1}^{l}(\xi_i + \xi_i^*) \tag{5.2.4}$$

$$\text{subject to} \quad \begin{cases} y_i - \boldsymbol{w}x_i - b & \leq \epsilon + \xi_i \\ \boldsymbol{w}x_i + b - y_i & \leq \epsilon + \xi_i^* \\ \xi_i, \xi_i^* & \geq 0 \end{cases} \tag{5.2.5}$$

The slack variables derive from a loss function which attributes a cost to deviations that imply

a slack variable absolute value larger than ϵ

$$|\xi|_\epsilon := \begin{cases} 0 & \text{if} |\xi| \leq \epsilon \\ |\xi| - \epsilon & \text{otherwise} \end{cases} \tag{5.2.6}$$

Now, to solve the optimization problem 5.2.5 and to extend the method to non-linear functions, it is transformed utilizing Lagrange multipliers. This yields a dualized version of the first (primal) optimization problem. Refer to [21] for its full derivation.

$$\text{maximize} \quad \begin{cases} -\frac{1}{2}\sum_{i,j=1}^{l}(\alpha_i - \alpha_i^*)(\alpha_j - \alpha_j^*)(\boldsymbol{x}_i \bullet \boldsymbol{x}_j) \\ \epsilon\sum_{i=1}^{l}(\alpha_i + \alpha_i^*) + \sum_{i=1}^{l} y_i(\alpha_i - \alpha_i^*) \end{cases} \tag{5.2.7}$$

$$\text{subject to} \sum_{i=0}^{l}(\alpha_i - \alpha_i^*) = 0 \quad \text{and} \quad \alpha_i, \alpha_i^* \in [0, C] \tag{5.2.8}$$

where, $\boldsymbol{\alpha}$ and $\boldsymbol{\alpha}^*$ are Lagrange multipliers. Note that the optimization problem doesn't contain any $\boldsymbol{w}$ anymore. It is only expressed in terms of input dot product. This is a major advantage when it comes to the solution computation.

Until here, only linear ϵ-SVR was considered. To extend it to nonlinear functions, the input $\boldsymbol{x} \in \mathbb{X}$ is mapped into a higher dimension feature space $\mathbb{F}$. If the data input seem fuzzy in its original space, a good mapping (ϕ) into a higher dimension usually makes it easier to segregate them. This mapping may however be very computationally expensive. But, as seen before, the optimization problem doesn't require more than computing the dot product of that mapping. The problem is reformulated as follows

$$\text{maximize} \quad \begin{cases} -\frac{1}{2}\sum_{i,j=1}^{l}(\alpha_i - \alpha_i^*)(\alpha_j - \alpha_j^*)k(\boldsymbol{x}_i, \boldsymbol{x}_j) \\ \epsilon\sum_{i=1}^{l}(\alpha_i + \alpha_i^*) + \sum_{i=1}^{l} y_i(\alpha_i - \alpha_i^*) \end{cases} \tag{5.2.9}$$

$$\text{subject to} \sum_{i=0}^{l}(\alpha_i - \alpha_i^*) = 0 \quad \text{and} \quad \alpha_i, \alpha_i^* \in [0, C] \tag{5.2.10}$$

where, k is a function resulting from the dot product of the mapping ϕ. The function k is called the ϵ-SVR kernel. The conditions a kernel must fulfill so that it is indeed the result of a dot product of some ϕ are given in [21]. In the non-linear case, the regression function (equ. 5.2.1) thus takes the form of a linear combination of the training data

$$f(x) = \sum_{i=l}^{l}(\alpha_i - \alpha_i^*)k(x_i, x) + b \tag{5.2.11}$$

Also, refer to [21], for an entire description of the algorithmic towards the optimization problem solution.

In the present case, the 2 following kernels are applied:

1. Linear: $k(\boldsymbol{x}, \boldsymbol{x}') = (\boldsymbol{x} \bullet \boldsymbol{x}')$
2. Radial Basis Function (RBF): $k(\boldsymbol{x}, \boldsymbol{x}') = exp(-\gamma||\boldsymbol{x} - \boldsymbol{x}'||^2)$

The linear kernel is chosen for the simple model expression it will yield. Also the rather linear behavior of H/L for varying panel heights, supports this choice. As seen, the linear kernel doesn't have any further parameter. Therefore the linear ϵ-SVR hyperparameters are: ϵ and C. The RBF is chosen to have a nonlinear function in the test and because it fits application for which no *a priori* knowledge about the data is available. Although such knowledge exists in this case, the RBF kernel ϵ-SVR is a general approach with famous high performances. The RBF brings an additional parameter (γ) into the ϵ-SVR model. The RBF ϵ-SVR hyperparameters are thus: ϵ, C and γ.

5.2.2 Data set

To train the model, the data set consists in the cross product of all values simulated already presented in Table 5.1.1 for the panel length and height, with a fixed reference value of 40° for the tilt. For computational reasons the tilt was then only crossed with a selection of the two other parameters as presented in Table 5.2.1. Also, less tilts were simulated but the range between 40° and 70° is kept. From the [STEP×PANEL_H×PANEL_L] cross product, 385 tuples of the MPM simulations geometry parameters and stack's feature are produced. From the [STEP×PANEL_H×PANEL_L×PANEL_TILT] cross product, 225 tuples are produced.

Table 5.2.1: Geometrical parameters used for the dot product that includes the panel tilt. These simulations are added to the one from Table 5.1.1 in which the panel tilt were only simulated with reference values of length and height.

Geometry			
Panel Height	H_{panel}	2-3-4	m
Panel Length	L_{panel}	0.5-1.5-2-4-8	m
Panel Width	W_{panel}	-	
Panel Tilt	Φ	50-60-70	°
Ground Tilt	Ψ	0	°

In total, the data set contains 610 tuples and their headers are:

INPUT:	STEP	PANEL_H	PANEL_L	OUTPUT:	H/L

Each tuple is rescaled to obtain a Gaussian distribution with zero mean and unit variance applying the following transformation

$$X' = \frac{X - \mu}{\sigma} \tag{5.2.12}$$

where, μ and σ are the mean and standard deviation of X (all tuples) respectively. Note that the data is assumed Independant and Identically Distributed (i.i.d.). This is a strong yet reasonable assumption thanks to the measurement idealization provided by the MPM simulations.

5.2.3 Model selection procedure

In supervised machine learning techniques such as SVR, the supervision consists in providing the model with a discrete number of input/output pairs ($\boldsymbol{x}$, $\boldsymbol{y}$). The goal is subsequently to predict outputs for input that are not available. Consequently, when choosing a model, the main objective is a good generality. In other words, a regression model is useless if it is good on the data it was trained with, but performs poorly on any other data. Such a model is said to be overfitting. In order to obtain generality, an independent assessment of performance is required. To that end, the learning procedure is specially organised.
For the SVR, the framework commonly consists in two steps: the tuning of the hyperparameters (A) and the regression performance validation (B). Each step must be carried on a distinct part of the data. This precaution prevents the overfitting [23]. The portion of the data given to (A) and (B) are respectively called the development and validation sets, both subsets of the original data set. To obtain good statistics, steps (A) and (B) are repeated several times. The whole process was implemented by [24] under the appellation of nested cross-validation (CV). Its implementation consists of 2 loops described below and illustrated in Figure 5.2.1. The whole process is repeated n times.

1. **The outer loop**: the original data set is split using a K-fold randomised CV (1.1) and fed to the inner loop (1.2). Then the inner loop's result is returned (1.3) and regression performance is assessed (1.4). Then it restarts at (1.1) until the end of the K-fold randomised CV. After (K=10) outer loops, each fold's best model and its performance score are returned (1.5) and the algorithm stops.

2. **The inner loop**: the data fed to the inner loop is presented to a hyperparameters grid search[2] (2.1). Using the same type of cross-validation as in the outer loop but with a smaller K, the (ϵ,C,γ) 3-dimensional space (for RBF) is entirely assessed (2.3) based on the regression performance of the trained model (2.2). The number of inner loops is the product of the size of the (ϵ,C,γ) space and the number K of the CV. At the end of each best CV, the (ϵ,C,γ) run mean performance score is stored. In the end of the grid search, the inner loop is over and the best (ϵ,C,γ) model is finally returned to the outer loop (1.3).

For the RBF kernel, the (ϵ,C,γ) space searched was: $[\epsilon]\times[\mathrm{C}]\times[\gamma]=[10^{-4}-10^{2}]\times[10^{-3}-10^{4}]\times[10^{-4}-10^{2}]$, with each range composed of the in-between orders of magnitudes. For the linear kernel, the space was reduced to $[\epsilon]\times[\mathrm{C}]$, with the same ranges. The outer loop CV was (K=10)-fold, the inner loop was (K=5)-fold and the process was repeated (n=10) times.

[2] The name "grid search" refers to a type of parameter combination check. The grid search assesses all points of the grid formed by the parameters it is constructed with. Other, more effective techniques exist.

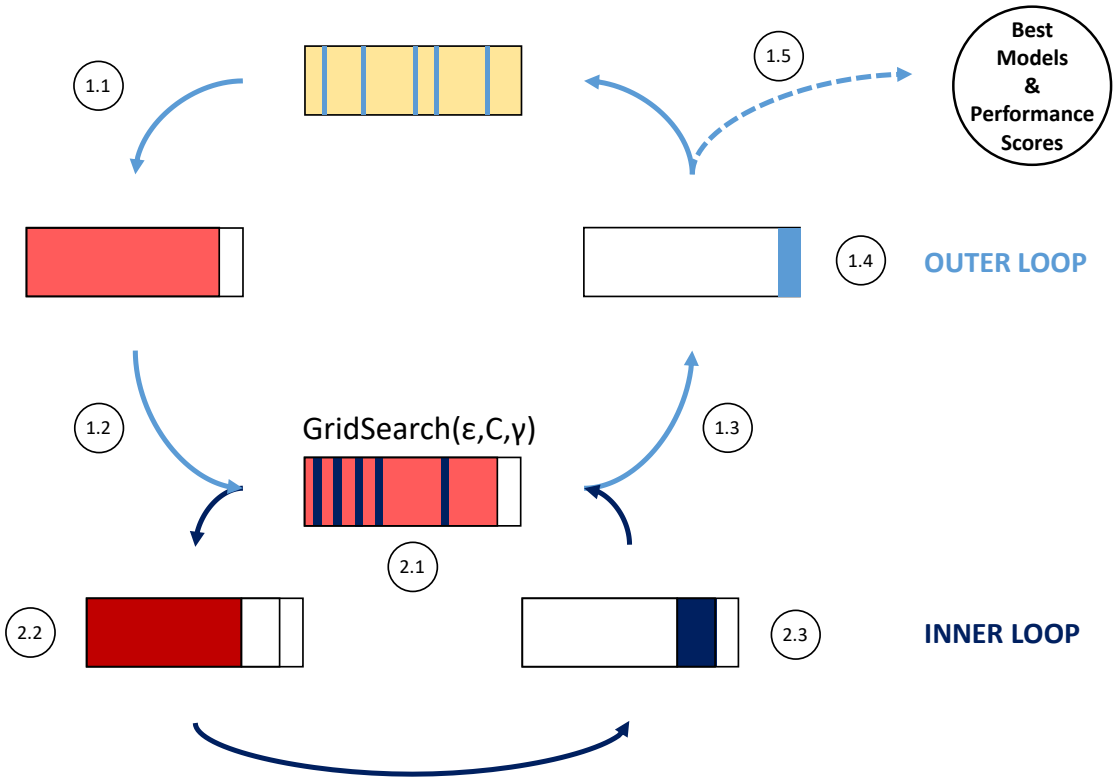

Figure 5.2.1: Illustration of the nested cross validation learning process. The outer loop takes care of randomly splitting the original data set (yellow) into the development (light red) and validation (light blue) sets. This splitting is implemented by a K-fold cross-validation. This makes sure that all the data set was once considered as validation set. The average or median score of the outer loop is therefore more representative. The inner loop looks for the best hyperparameters. It's implemented, for RBF, by an exhaustive grid search on the (ϵ,C,γ) space. Each node of the grid uses a K-fold cross-validation to assess the node's regression score. Each fold is first trained on a randomly split part of the development set, called the training set (dark red) and is assessed on the test set (dark blue). The best model with the best average score in the inner loop is finally passed to the outer loop.

5.2.4 Performance measure

The performance measure is the descriptor used for the regression scoring. It is the value cross validations rely on. The median absolute error (MAE) was chosen as performance measure. It provides a good vision of accuracy and bias while being more robust than its mean counterpart.
The different performance information provided are:

- From the inner loop: mean of (K=5)-fold CV scores of the best model found on the grid search. This means a model, with its hyperparameters and the mean of its 5 CV scores, are passed to the outer loop at each end of the inner loop.
- From the outer loop: (K=10) objects containing the model, hyperparameters, and mean

score given by the inner loop.

- From the n repetitions: (n=10) objects produced by the outer loop.

In the end, (n=10×K=10) = 100 models with their respective hyperparameters and score are produced. This is particularly interesting to provide statistics on the predicted data. Moreover, the regression fit of each of the models can finally be done on the entire data set, since their respective constructions (hyperparameters choice) were shown not to overfit on the entire data set. This yields a mean model on the full range of available data.

5.3 Results

In this section, model selection scores, derived from the nested CV are first presented. Then, the selected model performance is assessed, computed on the average of the 100 models the nested CV outputs. Finally, predictions are visualized together with ground truth data.

5.3.1 Model selection

The mean scores and their deviation for both kernels are presented in Figure 5.3.1. The selected hyperparameters for each grid search and each round are summarized in Figure 5.3.2. The RBF kernel shows a score about 2.5 times better than the linear kernel on validation sets around 0.005m. The difference between training and validation sets is larger for the RBF, suggesting a bit of overfitting. The validation score is however acceptable.

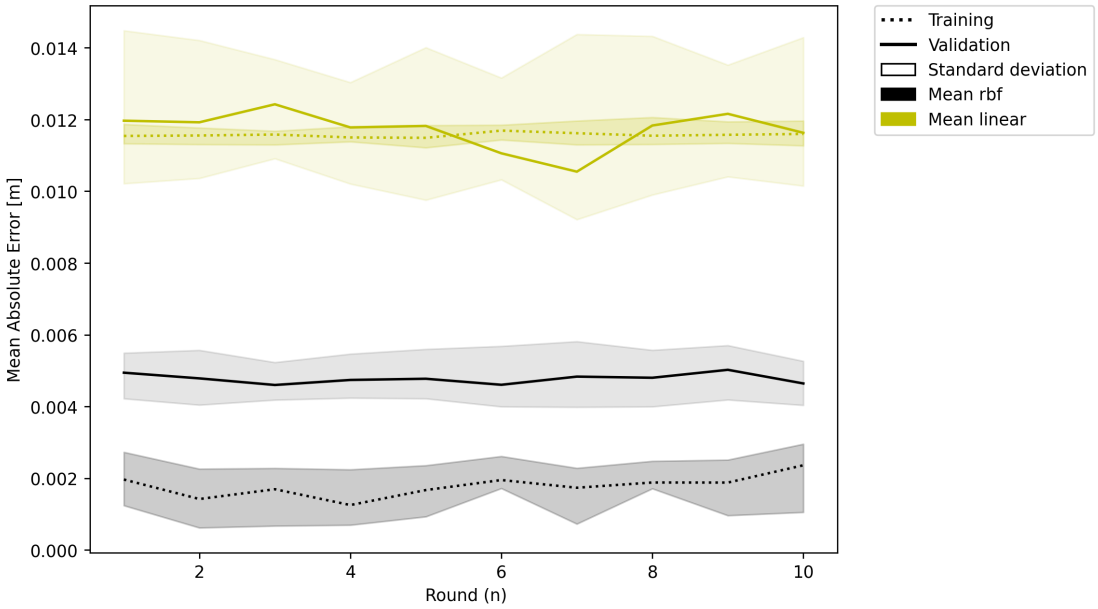

Figure 5.3.1: Evolution of the best models performance statistics over the 10 repetitions of the model nested CV. Statistics are computed over each rounds (K=10)-fold validation and development scores.

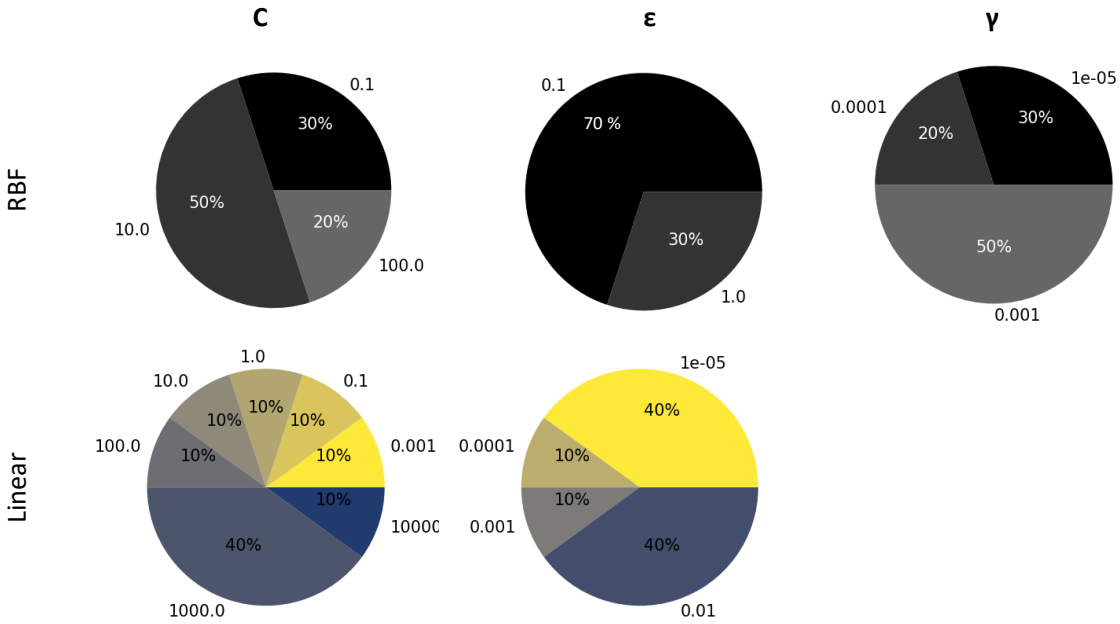

Figure 5.3.2: Summary of the values the grid search chose for each parameter over the nested cross validation. The linear kernel shows more sensitivity on each CV with multiple values taken, for C especially. For the RBF kernel, parameters take less different values and when they do, they are less different than for the linear kernel. To some extent, this shows the RBF kernel is better for this data set.

5.3.2 Model performance

If each model is fed with all the original data set input, they respectively produce corresponding predictions. Plotting these prediction ordered by quartile versus the original data set output ordered in the same fashion, results in the QQ-plot presented in Figure 5.3.3. If a model predictions and the "ground truth" data correspond, points should align on a line of slope 1. Depending on the deviation from that line, this type of QQ-plot allows to detect where the model shows most errors and whether a clear bias exist. The values of each prediction set quartile are presented in Table 5.3.1.

The linear model shows the worst performance. It over-estimates for small and large values. The distribution however on average follows the 1:1 slope. The RBF model performs much better with no detectable bias on this QQ-plot. Some few outliers distribute further from the 1:1 line. However those points show a large standard deviation, meaning the 100 models from which these prediction are computed don't agree well on them. It has to be remembered that the original data set itself contains some noise. This also counts for the few furthest outliers of the linear model.

To build the QQ-plot, the models are presented all the original data set and predictions are confronted to their 'ground-truth' counterparts. All errors measurements until now were carried during the nested CV, meaning the models were never fed all the data set. Hereafter, in Table 5.3.2, the absolute difference of models predictions and the 'ground-truth' data are averaged and their standard deviation computed. These are the models final performance

results.

Table 5.3.1: Prediction quartiles for both models

Quartile	Q1	Q2	Q3	Q4	
RBF	0.057	0.092	0.142	0.240	m
Linear	0.064	0.101	0.139	0.255	m

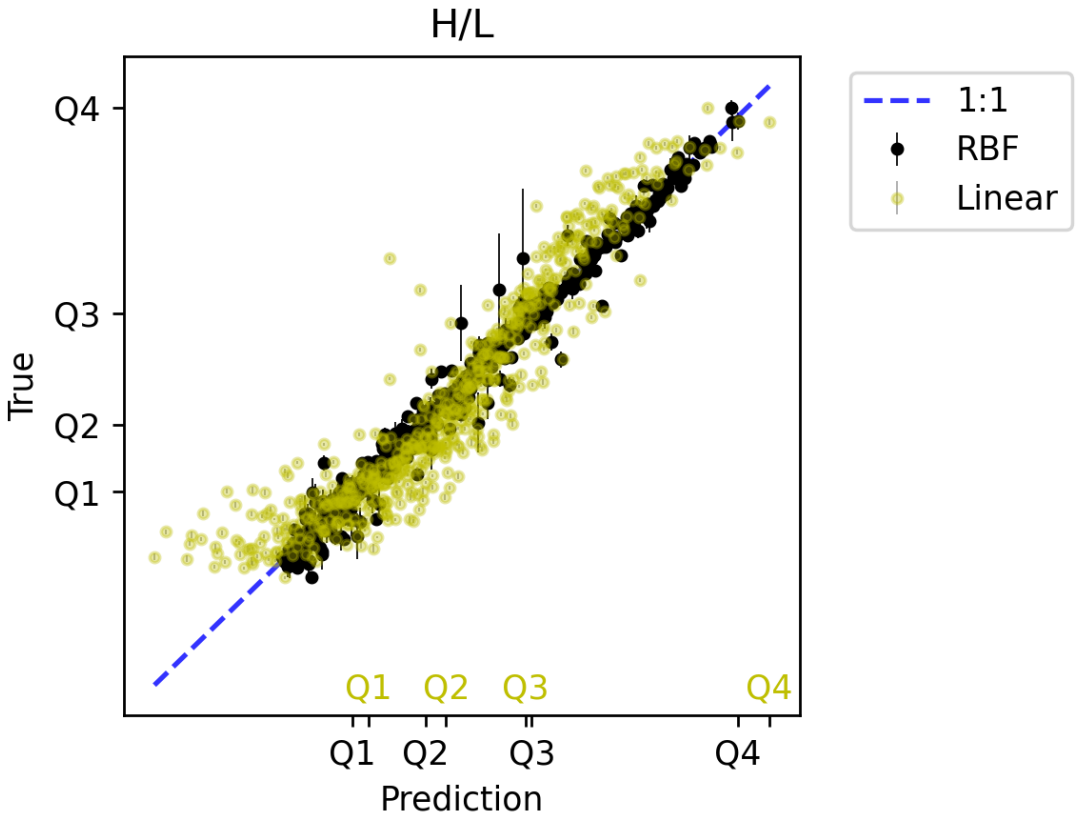

Figure 5.3.3: QQ-plot. Mean predictions of all models elected over the (K=10)-fold CV and (n=10) repetitions are plotted versus the original 'true' data set. Good position along 1:1 line, interesting large std for points further from the 1:1 line.

Table 5.3.2: Errors statistics of models predictions. The mean absolute error and its standard deviation is computed on all the predictions versus the 'ground truth' data.

Model	RBF	Linear	
Mean Abolute Error	0.004	0.015	m
Standard Deviation	0.005	0.013	m

Model predictions

To better visualize the model predictions, the same type of plot as in the geometry sensitivity analysis (Figure 5.1.1) are presented in Figure 5.3.4 on a single plot for the linear and RBF models predictions as well as the 'ground truth' data. This is repeated for all 3 geometry parameters.
For the panel length the RBF kernel fits well the logarithmic behavior in the lower range, except for the smallest value. The error bars get larger towards the upper length values but mean predictions are rather satisfying.
There are similar predictions for the panel height. Its impact on H/L being more linear, the linear kernel also performs well. Overall, there are only small error bars, showing the models' high confidence. The mean values are close to 'ground truth' points.
The worse predictions precision and accuracy of the RBF kernel shows the tilt effect was harder to isolate compared to the other parameters. This confirms the observation in the above sensitivity analysis. The trend is however followed. The linear kernel is not grasping the impact of the tilt on H/L at all. Its prediction are far off and it still has a high confidence.

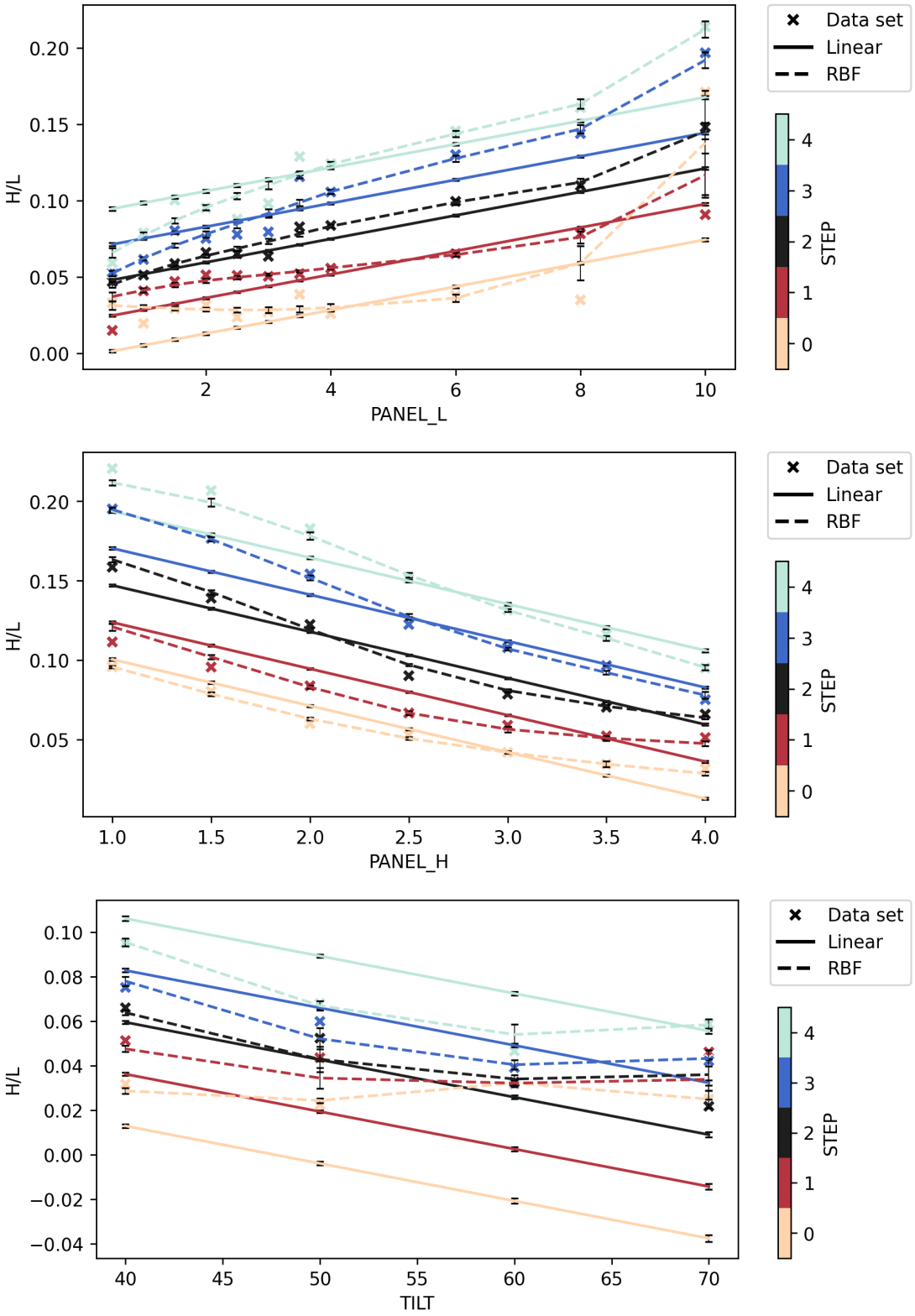

Figure 5.3.4: Visualization of the models predictions for all 3 parameters separately. When one parameter is varying, the two others are fixed on their reference value (Table 5.1.1).

KEY CHAPTER TAKEAWAYS

- **The SVR method is well adapted to the input data type and the stacks features behavior under geometrical variation.**
- **The behavior of H/L with respect to the panel length and tilt are much better predicted by the RBF kernel. The panel height showed a more linear behavior, yet the RBF kernel still performs better.**
- **The regression has a sub-centimeter accuracy and precision using the RBF kernel on meter-scale simulations. Those numbers are tripled using the linear kernel.**

Conclusion and Outlook

This thesis has assessed the large potential of the Material Point Method (MPM) for the study and modeling of snow slide-off from solar panels and subsequent snow stacks formation. In that context, to the best of our knowledge, snow stacks were quantified and characterised for the first time, both through simulations and observations. A multi-events workflow for the MPM and the snow constitutive model adopted was developed. Then, the work was separated in two parts. First, the understanding of the snow parametrization of the MPM and second, the modeling of the snow stack height and length variation with respect to the panel's geometry. In the first part, a sensitivity analysis on the snow parameters was carried and yielded an appreciation of each parameter's behavior and impact's order of magnitude. Then, to link this first part to the second, the MPM snow model was calibrated on the experiments that had been carried to that end and also produced controlled real measurements for the stack's characterization. In the second part, a MPM model with the calibrated fixed snow parameters was employed to carry a second sensitivity analysis on the panel's geometry parameters. It then served in the production of a relatively large data set for the building and comparison of two statistical models. The RBF approach showed a mean absolute error and standard deviation respectively around 3 and 2 times smaller than the linear approach. The RBF kernel was most adapted for the studied data set. The RBF regression showed sub-centimeter accuracy and precision on meter scale simulation data. Overall, this thesis has thus brought promising results on the path to a more general model of solar panels in alpine environments.

On that path, this thesis suggests multiple future works. First, on the technical side, the multi-events workflow developed needs more automatisation and a more robust transfer between 2 subsequent simulations. Also, the transformation of the stack between the slide-off events was not addressed but might have a significant role. Processes such as settling or sublimation may have impacts that need to be quantified. Second, regarding the modeling, a more general approach of some geometrical parameters should be studied. The snow slab thickness and the panel length could be linked in a single volume parameter. Even better, the snow density could be added to the list and integrated in the snow mass as single parameter. The different nature of these parameters however could produce non-trivial effects and the study should focus on them. Third, the major simplification of the present study is the fixation of the snow parameters in the statistical model. To reach the objective of a model that takes common meteorological data in addition to the geometrical one, there is the need of closing the gap between the MPM snow parameters and those meteorological variables. This includes a good

understanding of radiation processes in alpine environments. Finally and in combination with the third point, the slide-off triggering condition as well as the impact of multiple layering of the snow slab are two key characteristics. The development of the complete solar panel model should tackle these topics as well.

In more general terms, the developed method for simulation of sliding snow from a solar panel can be applied to a wide range of different structures. In this regard, the MPM used in this work could be applied to situations such as the slide-off from roofs or the prediction of preferential snow accumulation due to gravity, on and around natural or constructed structures.

References

Reviewed Literature

[1] O. Edenhofer, R. Pichs-Madruga, Y. Sokona, K. Seyboth, P. Eickemeier, P. Matschoss, G. Hansen, S. Kadner, S. Schlömer, T. Zwickel, and C. V. Stechow, "IPCC, 2011: Summary for Policymakers. In: IPCC Special Report on Renewable Energy Sources and Climate Change Mitigation," Tech. Rep., 2011. DOI: 10.5860/CHOICE.49-6309.

[2] A. Kahl, J. Dujardin, and M. Lehning, "The bright side of PV production in snow-covered mountains," *Proceedings of the National Academy of Sciences of the United States of America*, 2019, ISSN: 10916490. DOI: 10.1073/pnas.1720808116.

[3] J. Dujardin, A. Kahl, B. Kruyt, S. Bartlett, and M. Lehning, "Interplay between photovoltaic, wind energy and storage hydropower in a fully renewable Switzerland," *Energy*, vol. 135, 2017, ISSN: 03605442. DOI: 10.1016/j.energy.2017.06.092.

[6] F. von Rütte, A. Kahl, J. Rohrer, and M. Lehning, "How Forward-Scattering Snow and Terrain Change the Alpine Radiation Balance With Application to Solar Panels," *Journal of Geophysical Research: Atmospheres*, vol. 126, no. 15, 2021, ISSN: 21698996. DOI: 10.1029/2020JD034333.

[7] A. Stomakhin, C. Schroeder, L. Chai, J. Teran, and A. Selle, "A material point method for snow simulation," *ACM Transactions on Graphics*, vol. 32, no. 4, 2013, ISSN: 07300301. DOI: 10.1145/2461912.2461948.

[8] J. Gaume, T. Gast, J. Teran, A. van Herwijnen, and C. Jiang, "Dynamic anticrack propagation in snow," *Nature Communications*, vol. 9, no. 1, 2018, ISSN: 20411723. DOI: 10.1038/s41467-018-05181-w.

[9] C. Jiang, C. Schroeder, J. Teran, A. Stomakhin, and A. Selle, "The material point method for simulating continuum materials," in *ACM SIGGRAPH 2016 Courses, SIGGRAPH 2016*, 2016. DOI: 10.1145/2897826.2927348.

[11] X. Li, B. Sovilla, C. Jiang, and J. Gaume, "Three-dimensional and real-scale modeling of flow regimes in dense snow avalanches," *Landslides*, 2021, ISSN: 16125118. DOI: 10.1007/s10346-021-01692-8.

[13] F. H. Harlow and M. W. Evans, "A Machine Calculation Method for Hydrodynamic Problems," *Los Alamos Scientific Laboratory*, 1956.

[14] J. U. Brackbill, D. B. Kothe, and H. M. Ruppel, "Flip: A low-dissipation, particle-in-cell method for fluid flow," *Computer Physics Communications*, vol. 48, no. 1, 1988, ISSN: 00104655. DOI: 10.1016/0010-4655(88)90020-3.

[15] C. Jiang, C. Schroeder, A. Selle, J. Teran, and A. Stomakhin, "The affine Particle-In-Cell method," in *ACM Transactions on Graphics*, vol. 34, 2015. DOI: 10.1145/2766996.

[16] S. Filhol and M. Sturm, "Snow bedforms: A review, new data, and a formation model," *Journal of Geophysical Research F: Earth Surface*, vol. 120, no. 9, 2015, ISSN: 21699011. DOI: 10.1002/2015JF003529.

[18] O. Sus, M. Williams, C. Bernhofer, P. Béziat, N. Buchmann, E. Ceschia, R. Doherty, W. Eugster, T. Grünwald, W. Kutsch, P. Smith, and M. Wattenbach, "A linked carbon cycle and crop developmental model: Description and evaluation against measurements of carbon fluxes and carbon stocks at several European agricultural sites," *Agriculture, Ecosystems and Environment*, vol. 139, no. 3, 2010, ISSN: 01678809. DOI: 10.1016/j.agee.2010.06.012.

[19] B. E. Boser, I. M. Guyon, and V. N. Vapnik, "Training algorithm for optimal margin classifiers," in *Proceedings of the Fifth Annual ACM Workshop on Computational Learning Theory*, 1992. DOI: 10.1145/130385.130401.

[20] D. Meyer, F. Leisch, and K. Hornik, "The support vector machine under test," *Neurocomputing*, vol. 55, no. 1-2, 2003, ISSN: 09252312. DOI: 10.1016/S0925-2312(03)00431-4.

[21] A. J. Smola and B. Schölkopf, "A tutorial on support vector regression," *Statistics and Computing*, vol. 14, no. 3, 2004, ISSN: 09603174. DOI: 10.1023/B:STCO.0000035301.49549.88.

[22] V. N. Vapnik, *The Nature of Statistical Learning Theory*. 1995. DOI: 10.1007/978-1-4757-2440-0.

[23] G. C. Cawley and N. L. Talbot, "On over-fitting in model selection and subsequent selection bias in performance evaluation," *Journal of Machine Learning Research*, vol. 11, 2010, ISSN: 15324435.

[24] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and É. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, 2011, ISSN: 15324435.

Theses

[12] Alexander Maximilian Nilles, "Multi-material simulation with the Material Point Method," Ph.D. dissertation, Universität Koblenz Landau, Koblenz, May 2020.

[17] J. Gaume, "Evaluation of avalanche release depths," Ph.D. dissertation, Université de Grenoble, Grenoble, Oct. 2012.

Web pages

[4] Axpo Holding AG, *The largest alpine solar plant in Switzerland*, 2021. [Online]. Available: https://www.axpo.com/ch/en/energy-knowledge/pioneer-project-in-the-swiss-alps.html.

[5] Alpic Holding AG, *Gondosolar*, 2021. [Online]. Available: https://www.gondosolar.ch/.

[10] B. McGinty, *Continuum Mechanics*, 2012. [Online]. Available: continuummechanics.org.

A | Appendix

A.1 Small strain tensor limitation

If a pure rotation is taken as deformation gradient

$$\boldsymbol{F} = \begin{vmatrix} cos\theta & -sin\theta \\ sin\theta & cos\theta \end{vmatrix} \tag{A.1.1}$$

Substituting into the small strain definition as function of **F**

$$\epsilon = \frac{1}{2}(\boldsymbol{F} + \boldsymbol{F}^T) - \boldsymbol{I} = \begin{vmatrix} cos\theta - 1 & 0 \\ 0 & cos\theta - 1 \end{vmatrix} \tag{A.1.2}$$

The strain is thus not zero in case of $\theta \neq 2n\pi, n \in \mathbb{Z}$, even though no deformation occurs under any pure rotation.

A.2 Experiments setup

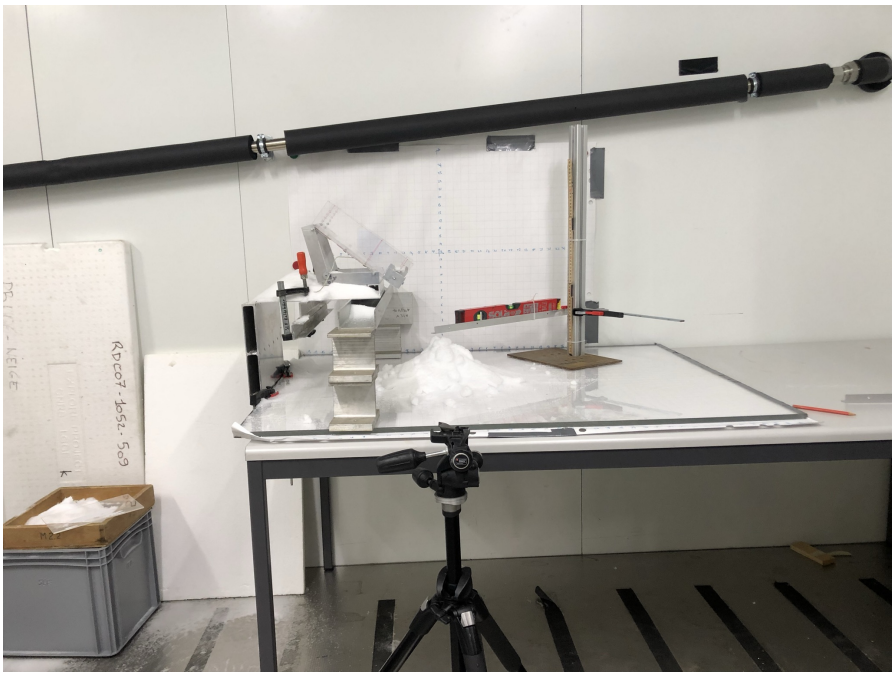

Figure A.2.1: The experiments setup. The experiments took place in the cold lab of the SLF in Davos (CH).

Figure A.2.2: Profile of a stack centered on its top.

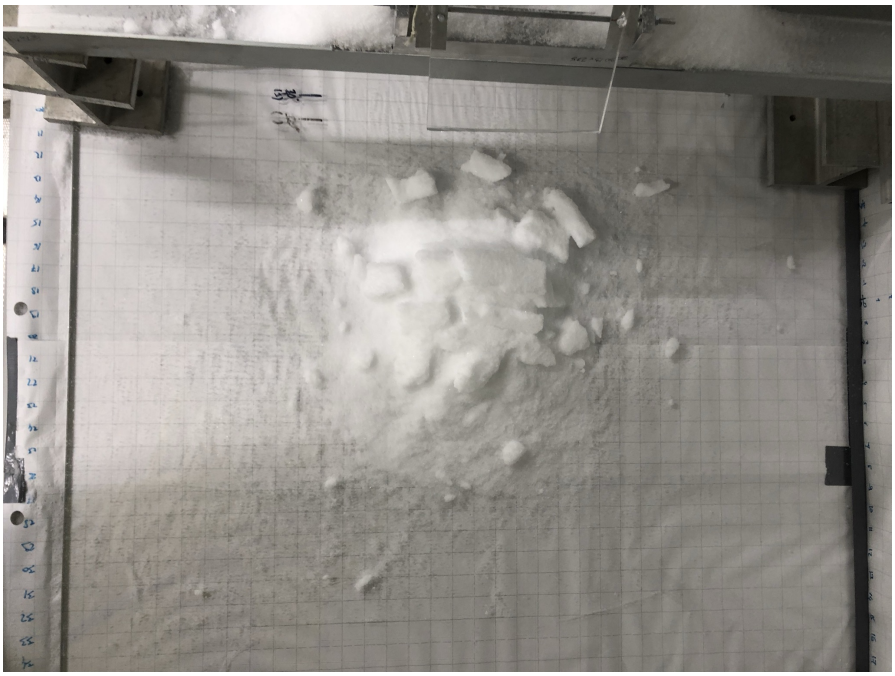

Figure A.2.3: Top view of the stack. The underlying grid allows to measure its limit. This image witnesses from the boundary delimitation problem.

A.3 Management of MPM and Houdini for multi-events simulations

Avalanche modelling with MPM has a single-event workflow. First pre-processing is performed in Houdini[1], a 3D animation tool, then the MPM simulation is run and finally results are visualized and post-processed in Houdini and potentially exported for further analysis.
In the case of snow slide-off from solar panel, a multi-events workflow is necessary to, for instance, simulate a whole winter season. It takes the form of a script that step by step runs different tools responsible for pre-processing, running the MPM and post-processing. In addition it can loop on those steps using the output of the previous loop as input of the next one. The following sections describe how the MPM is used in this study.

A.3.1 Lua binding

The MPM algorithm is a cpp code for which a Lua binding was developed to easily setup simulations. The Lua script can be seen as a remote controller which allows to define MPM parameters and initialize the domain and its content. Important for automatization and looping, it allows to append updated values of parameters which overwrites precedent instances. The new values are then taken for the initialization. Two Lua script were used for the initialization and for the updating loops.

A.3.2 Simulation domain

The simulated domain contains the discretized material and collision objects. A constant acceleration field is applied to the whole domain, locally mimicking gravitation. It is parallel to the z-axis and directed towards z-axis negative values.

Materials

The modelled material is imported into the MPM first as a voxel (*volumetric pixel*) based object. The animation tool Houdini, allows to shape any type of continuous volume and then discretize it into smaller regular volumes called voxels. Their number and consequently their size ultimately depend on each simulation. Each voxel is attributed a number of particles. So the smaller the voxels the more particles the material gets descretized into. As a rule of thumb, the voxel size should be chosen equal to the Eulerian grid size (dx). To avoid numerical fracture when computing the weights for particles transfer (Section 1.2.2), a minimum of 4 and 6 particles per voxel is needed for 2D and 3D simulations respectively (reference??). In addition, material object must be given a density and a volume to attribute a mass to each

[1] It is also possible to form simple geometries with the Lua library. This option was not used in this project.

particle

$$V_{slab} = L_{slab} \times W_{slab} \times T_{slab} \tag{A.3.1}$$

$$N_{voxel} = \frac{V_{material}}{(dx)^3} \tag{A.3.2}$$

$$N_{particles} = N_{voxel} \times \frac{N_{particles}}{voxel} \tag{A.3.3}$$

$$m_{particles} = \frac{\rho \times V}{N_{particles}} \tag{A.3.4}$$

While it is easy to compute the snow slab volume, the task is more complex for the stack. The same volume as the original is attributed to the fallen snow slab. The stack gets a volume equal to the number of events multiplied by the original volume

$$V_{stack} = n \times V_{slab} \tag{A.3.5}$$

with n = 0,1,2... the number of slide-off events the stack is made of.

Collision objects

Objects constraining the domain are exported in the exact same format as the material particles. In the Lua script they are however imported as level set which sets them as collision object for the MPM. Each object has to be attributed a boundary condition: STICKY, SLIP or SEPARATE (Section 1.2).

A.3.3 Houdini

Houdini is a 3D fully procedural animation tool developed by SideFX. In relation with MPM it usually has two different roles of pre- and post-processing:

1. Pre-processing: prepare the material and domain geometry.
2. Post-processing: visualize, analyse and render particle properties.

In this study, Houdini was used in its scripted format, using the python module "hou" which allows to script any action available in the user interface (UI). Different options for the use of this module exist. The external editor opened from within Houdini's UI was chosen. Once the external editor opened, the user has access to the full library and capacity of Houdini while keeping all usual python functionalities. This powerful tool perfectly matches the needs of a multi-event circular MPM workflow.

A.3.4 Workflow

The developed workflow is presented in the chart in Figure A.3.1. It allows to run multi-events simulations (multi-event loop) and automatically iterate over all parameters combinations possible based on the input (parameters update loop). For instance, if 3 values for the internal

friction M and 3 values for the ground tilt are given while all other parameters are given 1 value, the parameters update loop will iterate $\Pi_p^n(V_p)$ times, with n the number of parameters and V the number of values the parameter p can take. In the previous example this means 9 combinations will be tested. As mentioned in the example both geometrical and snow parameter can be modulated. The total number of MPM simulations is the product of the parameter update loop and the multi-event one. Keeping the same example, if 3 slide-offs are required, a total of 27 MPM simulations will be run.

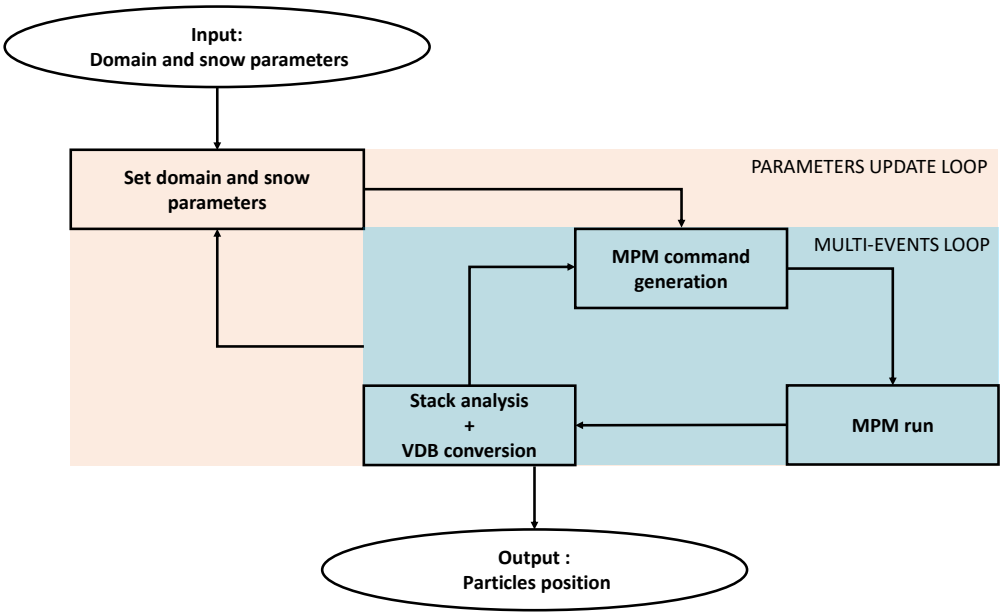

Figure A.3.1: Caption

Tools

In Houdini, python scripted list of commands can be encapsulated in a tool. By calling this tool, the whole script it contains is run by the internal python interpreter. In order to achieve the workflow presented in Figure A.3.1, the following tools were developed:

1. Controller: the main script from which all commands are given. It contains the parameters and different functions as well. The parameters update loop takes the form of multiple nested for-loops and the multi-events loop is an additional loop nested within the parameters update one. This tool also contains the Unix commands that trigger the MPM simulation.

2. Geometry processor: called by the controller, it takes the geometry parameters as arguments, builds and saves the simulation domain constraints, i.e. levelsets and the material VDB.

3. Post-processor: called after a MPM simulation is finished to transform and save the result into material VDB.

Stack conversion

The conversion of the stack to a VDB levelset is the key step of the circular workflow which allows to use the output of one simulation as input of the next one. Ideally, the conversion should not change the morphology of the stack in the subsequent simulation. Also, the same amount of particles should constitute the stack. The "VDB from particles" houdini node was chosen to stick as much as possible to these requirements. This nodes detect particles of a minimum radius given as parameter and creates a voxel around it. Since the particles size changes in the MPM simulation, the nodes parameters are adapted to each simulation so that the previous requirements are satisfied within a reasonable range.

A.4 Post processing of simulation data

A.4.1 Outlier 3D mitigation algorithm

In order to remove outliers and have a common definition of the stack's edge, mass is integrated circularly up to 97% of the mass. In fact, since particles carry all the same mass in MPM, only the count of particles must be integrated. Also, the developed approach rather removes the outermost 3% of particles for time optimisation. It computes the center of mass X-Y coordinates and uses polar coordinates to circularly, each lap closer to the mass center, detect and count particles lying in a arc-window of 2 times the increment of angle in width and 2 times the increment in radius. The initial radius is derived from the maximum coordinates in X and Y which could be from 2 distinct particles as $R_{ini} = \sqrt{max(X)^2 + max(Y)^2}$. This way it is certain that the first integration circle doesn't miss any particle. The increments of angle and radius are based on the resolution chosen for the data reduction process (sectionA.4.3).

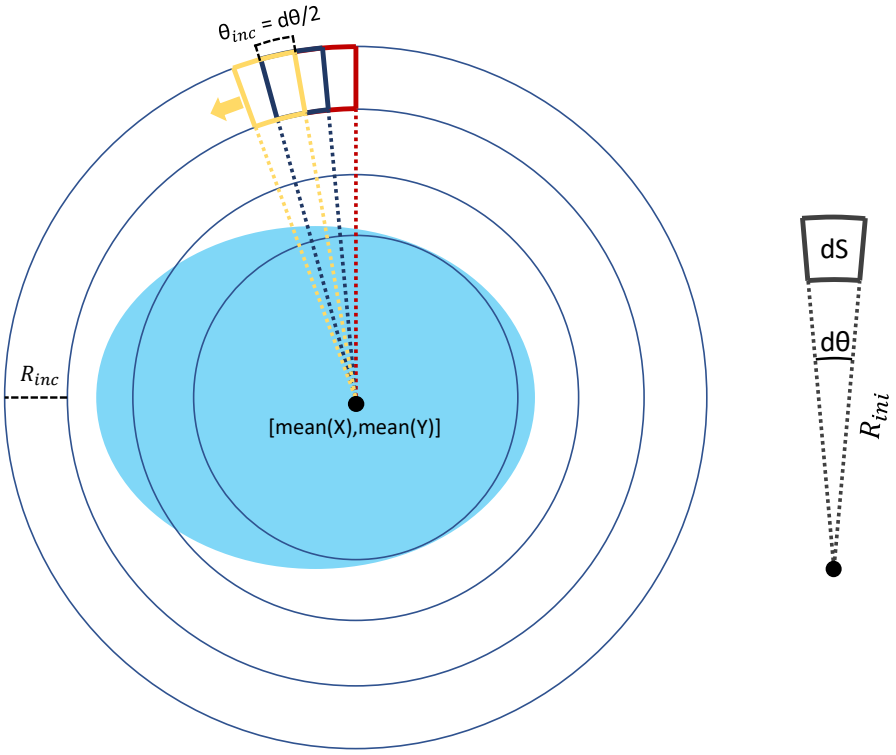

Figure A.4.1: Caption

A.4.2 Outlier 2D mitigation algorithm

In 2D, the algorithm has the same principle as the 3D one, but doesn't need to rotate. Instead, it starts from the 2 ends of the stack and starts to integrate by a increment ∂x towards the center of mass. The threshold is also set to 3%, keeping the 97% other particle for the analysis

of the stack.

A.4.3 Data reduction

After the outlier mitigation, the shell of each stack was extracted to reduce the number of particles. To do so, the stacks were split on a cartesian grid forming cells which contain a fraction of the stack. At each cell, only the particle with the maximum z coordinate is kept. The smoothing effect of the algorithm depends on the grid size. The resolution adopted splits the x and y ranges in 30 cells, giving grids of 900 cells in total. For stacks of about 0.3m long, this is a grid size of 0.01m on the x axis. It is even finer on the y axis.

A.4.4 Data transformation

When ground tilt (Ψ) is different from 0, the stack height and length are not aligned on the referential. In order to compute the these features, all particles coordinates are rotated by the appropriate angle. The ground is tilted about the Houdini's Z axis which is the usual horizontal Y axis directly converted at data import to avoid further confusion. All particles coordinates triplets are hence multiplied by the adequate rotation matrix

$$R_y(\Psi) = \begin{vmatrix} cos(\Psi) & 0 & sin(\Psi) \\ 0 & 1 & 0 \\ -sin(\Psi) & 0 & cos(\Psi) \end{vmatrix} \tag{A.4.1}$$