ISSM moulin flow routing

Tim Hill

September 2025

1 Overview

Moulins are placed based on steady-state flow routing, enforcing a minimum moulin discharge and optional external conditions. Once moulins have been placed, time-dependent moulins are calculated by transferring melt from elements to the corresponding moulins.

2 Notation

Table 1: Notation for equations and corresponding Matlab variable names

Symbol	Matlab variable	Description	Units
$\dot{m}$	`melt`	Surface melt rate	$\mathrm{m\,w.e.\,s^{-1}}$
Q_m	`discharge`	Discharge into moulin	$\mathrm{m^3\,s^{-1}}$
Q_min	`discharge_threshold`	Minimum discharge threshold	$\mathrm{m^3\,s^{-1}}$
$I(x,y)$	`condition`	Indicator function for external condition	N/A
f	`decay_factor`	Relative tolerance in minimum moulin discarge	-
p	`p`	Acceptance probability for sequential moulin simulation	N/A
z_s	`phi`	Surface elevation	m

3 Algorithm for moulin placement

Moulin positions are simulated to satisfy conditions:

1. Moulin discharge is not less than a prescribed threshold: $Q_\mathrm{m} \geq Q_\mathrm{min}$
2. External conditions (e.g., based on the strain-rate field): vertex i is only a candidate moulin position where the indicator function $I(x_i, y_i) = 1$.

3.1 Flow routing

The discharge condition (1) requires that we estimate supraglacial flow paths to compute discharge into each candidate moulin. Following Hill and Dow (2021), we deterministically route water according to the surface elevation gradient. Starting from each triangular element, all water produced within the element is routed through the lowest elevation node. From this first node, we sequentially route flow to the lowest elevation neighbouring node. At each step, the flow path is terminated if the node is a moulin, on the boundary, and/or on the ice shelf. By looping through all elements in the mesh, we obtain flowpaths and flow accumulation maps for all surface meltwater produced in the catchment. This will be referred to as a single pass or iteration of flow routing. Flow routing is computed by the function `compute_flow_accumulation`.

3.2 Simulating moulin positions

The strategy to simulate moulin positions is different here than Hill and Dow (2021) in two main ways: (a) we have added condition (2), and (b) we now build up the list of moulins from an initial moulin-free state, rather than culling moulins from a large initial list.

The algorithm begins with the flow routing and accumulation map with no moulins. For each iteration:

1. Randomly propose a candidate moulin at node i among all nodes where conditions (1) and (2) are both satisfied

2. Recompute the flow accumulation and routing map including the candidate moulin and all previously placed moulins

3. REJECT the candidate moulin (node i) if, by adding this moulin, the discharge into previously placed moulins drops below a specified threshold, fQ_{min}, where $0 \leq f \leq 1$ controls how far below the minimum threshold we allow the discharge to drop.

4. ACCEPT the candidate with probability p otherwise. Append the candidate to the list of moulins.

The simulation is stopped after a fixed number of iterations. An ad-hoc stopping criteria is to terminate the iterations when the acceptance rate for candidate moulins is low. As more iterations are run, the moulin discharge converges to the range $[fQ_{\mathrm{min}}, 2Q_{\mathrm{min}}]$. For a mesh with $\mathcal{O}(50,000)$ nodes, this may take an unreasonable number of iterations. The random choice in step (1) means that a different moulin configuration will be reached each time the algorithm is run. This algorithm differs from Siu (2022), who used an additional constraint that the algorithm starts from the highest elevation in the domain where both conditions are satisfied and searches downhill. This elevation-based search may be a more efficient approach, since a candidate moulin is unlikely to be rejected based on its precense cutting off flow to previously located moulins. The tradeoff

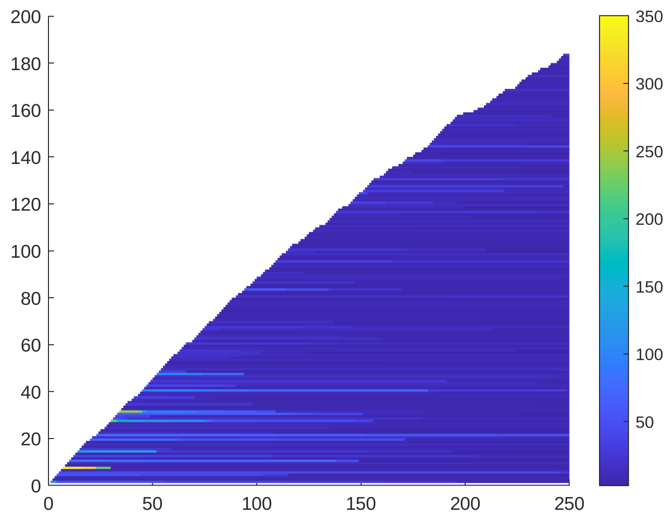

Figure 1: Trace of moulin discharge for each iteration in the simulation.

may be that Siu (2022) has less stochasticity in the selected moulins and a narrower distribution of moulin discharges. Function `place_moulins` performs the iterative search described by steps (1)–(4).

4 Time-dependent moulin inputs

The flow-routing code (`compute_flow_accumulation.m`) returns the moulin "catchments": that is, the list of elements that drain through each moulin. Since these catchments do not change with time, time-dependent moulin inputs can be easily computed by summing the melt generated in each catchment. See the example `time_dep_moulin_inputs.m`.

5 Example

This example considers moulins on grounded parts of the Amundsen Sea Sector of west Antarctica. For moulin placement, steady-state surface melt is represented by a highly idealized linear interpolation between $2\,\mathrm{m\,w.e.\,a^{-1}}$ at sea level and $0\,\mathrm{m\,w.e.\,a^{-1}}$ above $1500\,\mathrm{m\,asl}$. This example does not enforce an external condition, i.e. $I(x, y) = 1$ everywhere.

$$\dot{m} = \max\left(0, \frac{1500 - z_{\mathrm{s}}}{1500}\right) \tag{1}$$

For this example, we also set:

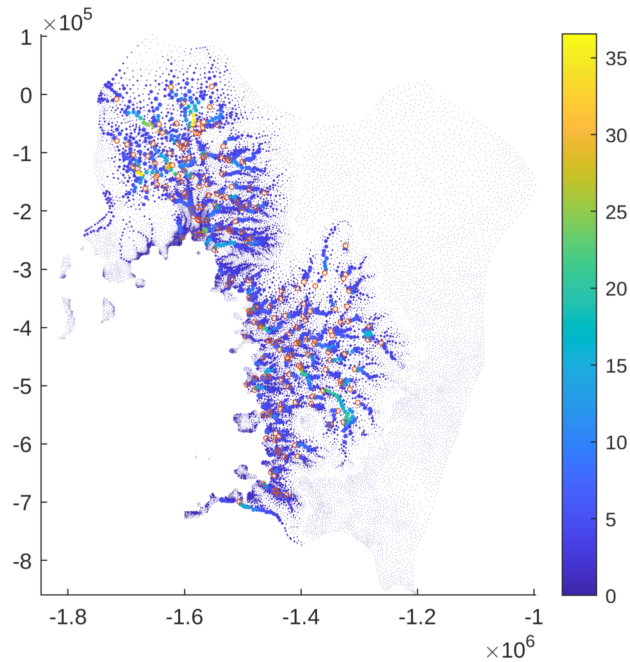

Figure 2: Flow accumulation and moulins (marked with circles) for the idealized Amundsen Sea Sector example. Colormap indicates the accumulated discharge ($\mathrm{m^3\,s^{-1}}$).

- $Q_{\min}$ = `discharge_threshold` $= 5\,\mathrm{m^3\,s^{-1}}$
- f = `decay_factor` $= 0.5$
- `maxiter` $= 250$
- No external condition, i.e. moulins are allowed to form anywhere that the minimum discharge threshold is met

Figure 1 traces when each moulin was added through the simulation. The slope of the "staircase" indicates the rate at which moulins are added. Decreases in moulin discharge indicate that the flow into the original moulin has been split between two moulins. Over 250 iterations, 184 moulins are placed, with the rate of placement slowing down in the later iterations.

Figure 2 maps the simulated moulins. Moulins tend to form densely where lots of surface flow accumulates, especially along the Pine Island trough. Flow is more dispersed across the width of Thwaites Glacier, leading to a lower moulin density.

Now that moulins are placed, we use the supraglacial catchments to determine seasonally varying moulin inputs. We multiply the steady-state inputs by

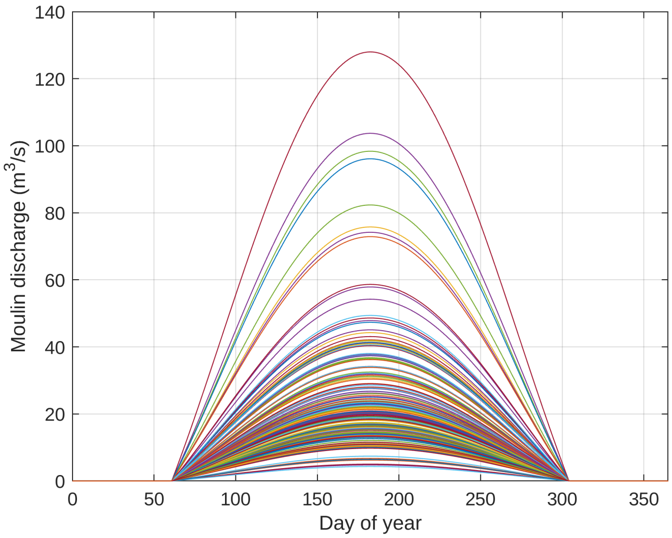

Figure 3: Discharge into the 184 moulins with idealized seasonal forcing.

a truncated sinusoid:

$$\dot{m}(t) = \max\left(0, \left(0.5 - \cos\frac{2\pi t}{T_{\text{year}}}\right)\frac{1500 - z_{\text{s}}}{1500}\right). \tag{2}$$

Figure 3 shows the resulting seasonal moulin inputs, computed by multiplying the melt rate defined on mesh elements by the element area, and summing across each moulin catchment.

References

Hill, T. and Dow, C. F. (2021). Modeling the dynamics of supraglacial rivers and distributed meltwater flow with the Subaerial Drainage System (SaDS) model. *Journal of Geophysical Research: Earth Surface*, 126(12):e2021JF006309.

Siu, K. (2022). *Modelling Subglacial Hydrology under Future Climate Scenarios in Wilkes Subglacial Basin, Antarctica*. PhD thesis, University of Waterloo.