# GEO2332 Numerical Methods for Physical Geography

## How will the Haut Glacier d'Arolla respond to future climate change?

### Introduction

The Haut Glacier d'Arolla is a 4km long, temperature valley glacier in the Swiss Alps (Goodsell *et al*., 2018). Figure 1 shows the 1990 glacier 1990 flows northwards from two basins. Since the Little Ice Age, the glacier has been steadily retreating (GLAMOS, 2022), due to surface heat fluxes (Arnold, 2018). It is important to predict the future changes, to understand the glacier's response under climate change scenarios. In 1990, Sharp *et al.* (1993) studied the glacier, the results of which were used to create the parameters of the of the model analysed in this report. The model uses parameters of mass balance gradient (M) ($yr^{-1}$), flow parameter (A) ($yr^{-1}$ $Pa^{-3}$), and equilibrium line altitude (ELA) (m), to aim to create valid projections.

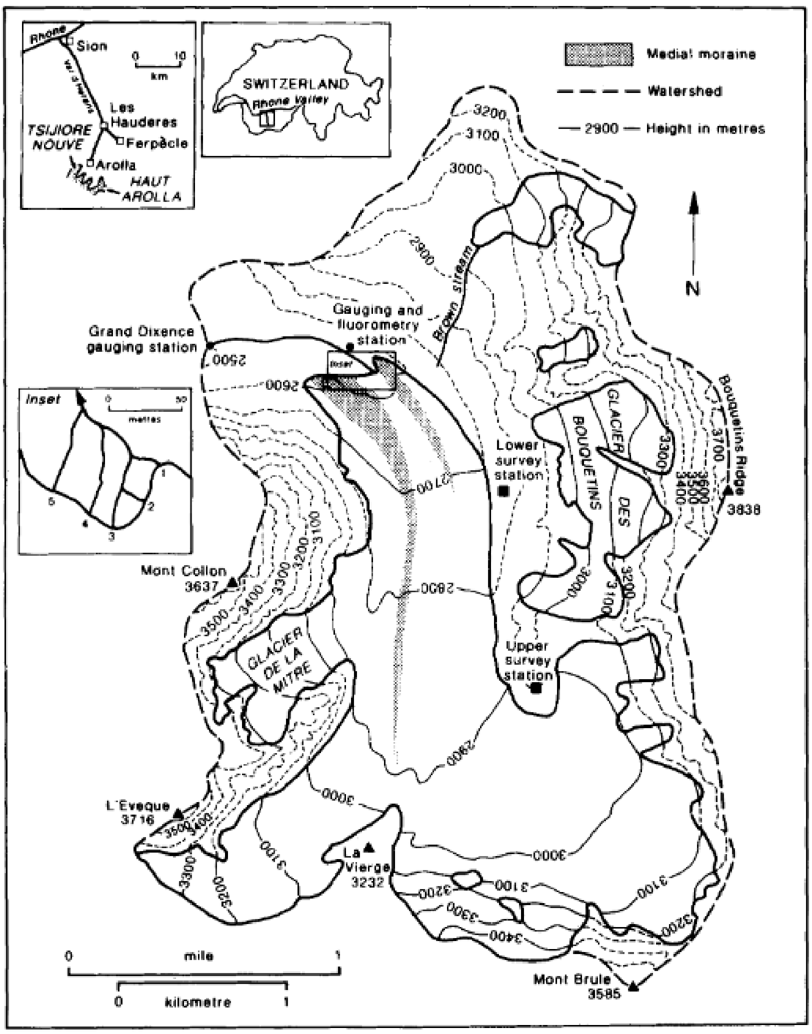



*Figure 1. Figure from Sharp et al. (1993). Map of the 1990 Haut Glacier d'Arolla, the time the glacier model parameters were created.*

## **Sensitivity Analysis**

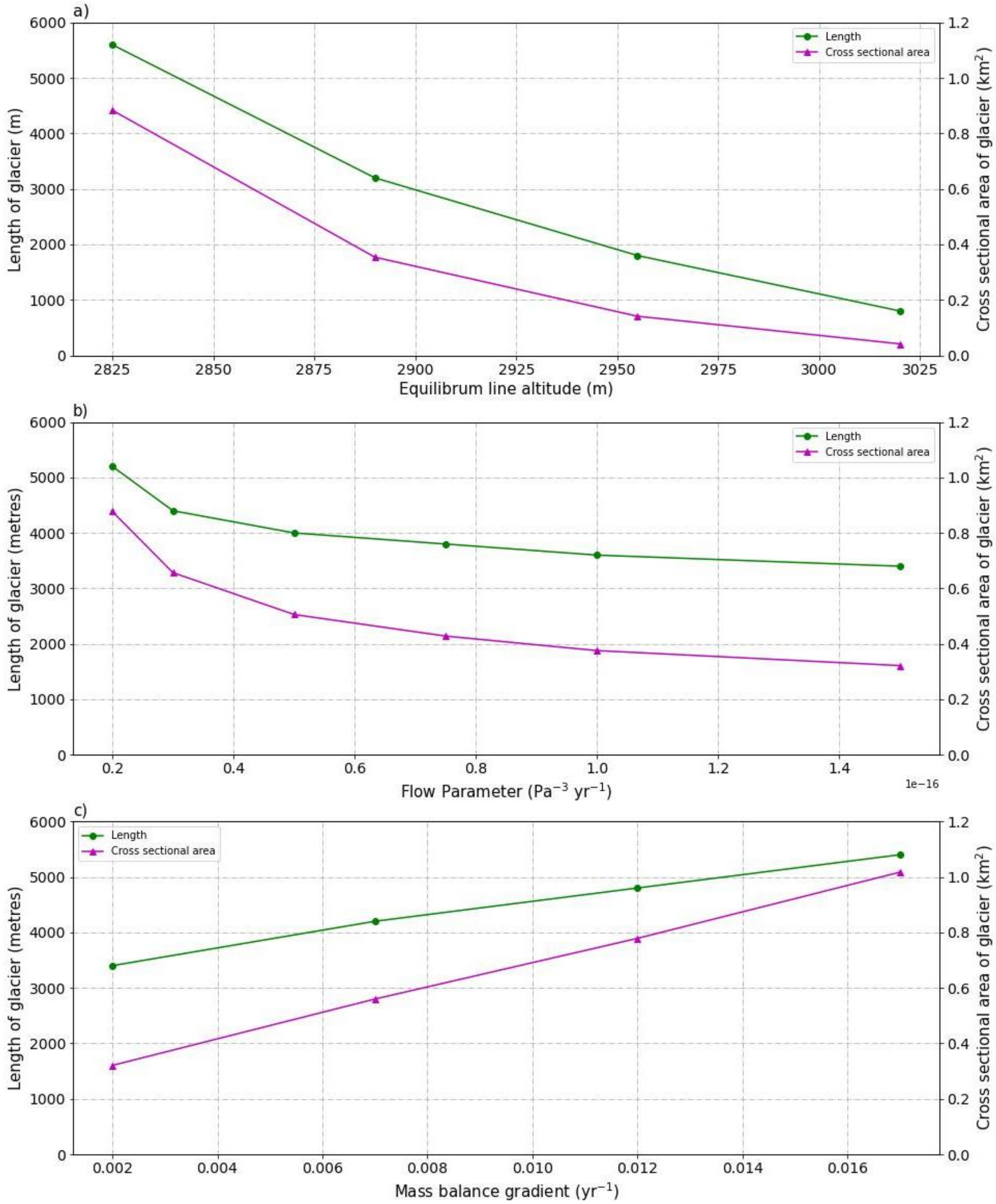



*Figure 2. Sensitivity analysis of parameters against the length and cross-sectional area of Haut Glacier d'Arolla. Graph 'a' shows ELA (m) parameter, graph 'b' shows A ($yr^{-1}$ $Pa^{-3}$) and graph 'c' shows M ($yr^{-1}$). The green line refers to the length of the glacier and the purple to the cross-sectional area.*

| Ice temperature (°C) | A ($yr^{-1}$ $Pa^{-3}$) |
|---|---|
| 0 | $2.14 \times 10^{-16}$ |
| -2 | $0.75 \times 10^{-16}$ |
| -5 | $0.50 \times 10^{-16}$ |
| -10 | $0.15 \times 10^{-16}$ |
| -15 | $0.09 \times 10^{-16}$ |
| -20 | $0.05 \times 10^{-16}$ |

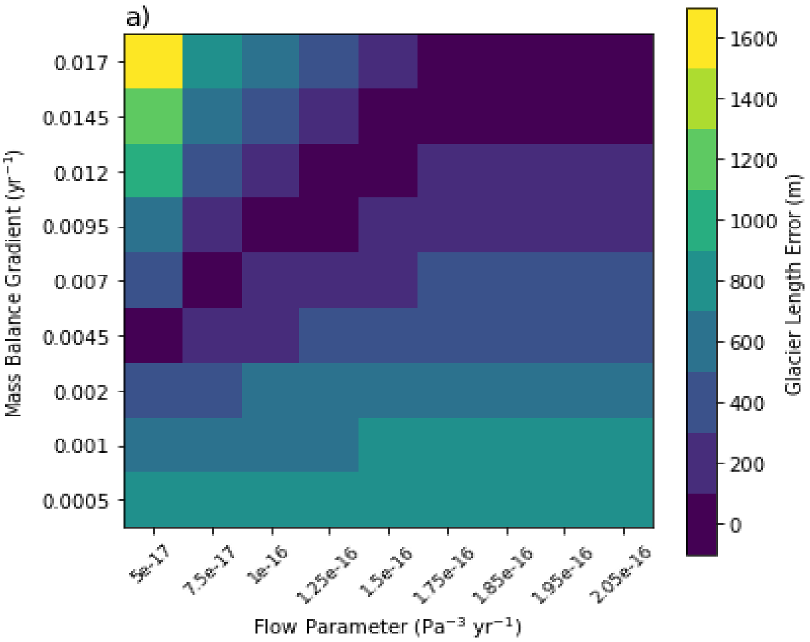
*Figure 3. Ice temperature and the corresponding flow parameter values*

Figure 2, graph 'a' shows the model is most sensitive to changes in the ELA. Decreasing the ELA from 2825 to 3025m, resulted in the length going from >5000m to 500m, and the cross-sectional area decreasing by >90%. ELA determines the point where mass change is zero, so it's impact determines the length of the glacier. A higher ELA means ablation starts at a higher altitude, so the glacier doesn't reach as far down the valley due to warmer temperatures. This explains why when the model is run with a higher ELA parameter, the glacier is shorter and has lower cross-sectional area.

Graph 'b', shows increased sensitivity for lower A values, where the ice temperature is <-20°C (refer to figure 3). The glacier is a temperature glacier, so ice temperatures are higher as it is made of 'warm ice', whereby some parts are at melting point (Douglas, 2014). It is therefore more appropriate to look at A values where ice temperature is 0 to -2°C. For values of 0.02 x $10^{-16}$ and 0.03 x $10^{-16}$ cross-sectional area decreases by 0.5km$^2$ and length decreases gradually by 950m. Size decreases because higher temperature of the ice causes more ablation, which means the glacier does not slide as far. Decreases in accumulation can also explain

Graph 'c' shows how outputs both increase steadily over parameter change from 0.002 to 0.017 yr$^{-1}$, causing length to increase by 2025m, and the cross-sectional area to increase by 69%. As M increases there is a greater difference between accumulation and ablation, steeping the mass balance gradient. The glacier responds by lengthening and becoming thicker to adjust to a new steady state. The Haut Glacier d'Arolla is characterised by high M as it is a temperate glacier, which is characterised by high winter snowfall and summer melt rates (Douglas, 2014).

**Calibration**



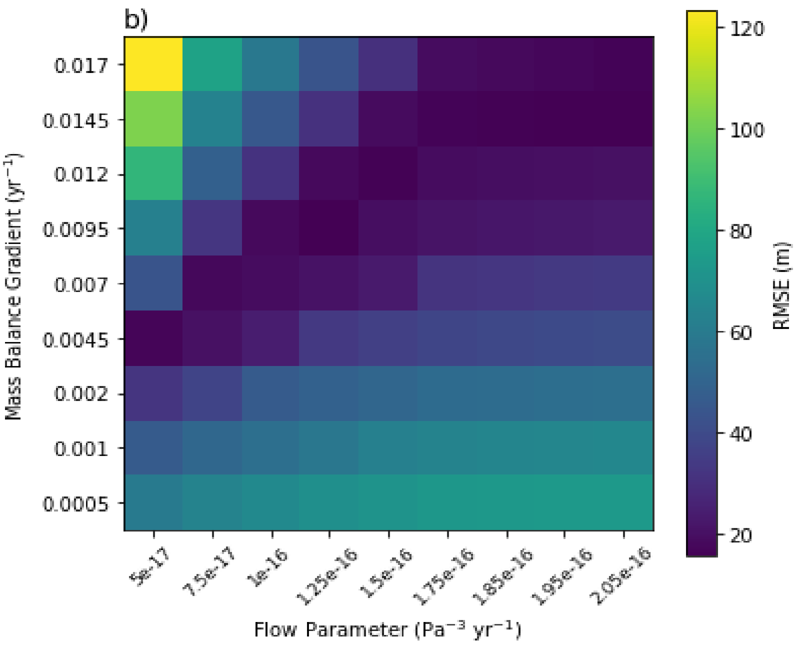




*Figure 4. Model calibration outcome. Flow parameter (yr$^{-1}$ Pa$^{-3}$) and mass balance gradient (yr$^{-1}$) plotted. Showing Glacier Length Error (m) in graph 'a', and Root Mean Square Error (RMSE) (m) in graph 'b', in relation to the 1990 Haut Glacier d'Arolla. Dark blue shows parameter combinations with least error and yellow shows combinations with most error.*

Figure 4 shows multiple optimum parameters sets which are very similar to the 1990 glacier with minimal error, this is equifinality. Graph 'a' shows <100m of glacier length error with multiple parameter combinations, for example when M is 0.012$yr^{-1}$($yr^{-1}$) and A is 1.5 x $10^{-16}$ ($yr^{-1}$ $Pa^{-3}$). Graph 'b' has similar optimum parameters. Optimal combinations occur when M exceeds 0.002 ($yr^{-1}$), because M is higher for Haut Glacier d'Arolla due to it being a temperate glacier. 'A' values vary as the model needs to only be similar to that of the 1990 glacier in size (not flow speed), to display minimal error. The addition of ELA would allow for a better indication to the optimal parameters. By plotting the ELA, in a 3D graph (with a Z-axis), fewer optimal parameters combinations would be found, which would allow a precise representation of the 1990 glacier from a more in-depth model calibration.

**Validation**

Validity would increase if basal sliding was included as the glacier is underlain by a subglacial network of meltwater streams (Sharp et al., 1993), which lubricate the bed increasing basal sliding. Changes in length are not accounted for due to increased flow velocity at the bed.

The model calibration may be applicable to other glaciers the Alps for the 1990s. Many glacial systems will experience similar A and M values as they are in relatively close proximity, so would have a similar accumulation rate.

Model validity would improve if observations from other years were included, not only 1990. This would allow easier comparison between 1990 predictions and recent observations. It would be unsuitable to use 1990 parameters as a basis for future glacier predictions, as A changes as ice temperature changes, and M changes when atmosphere changes. The model has low temporal validity as it does not account for glacier changes due to global warming.

**Future Projections**

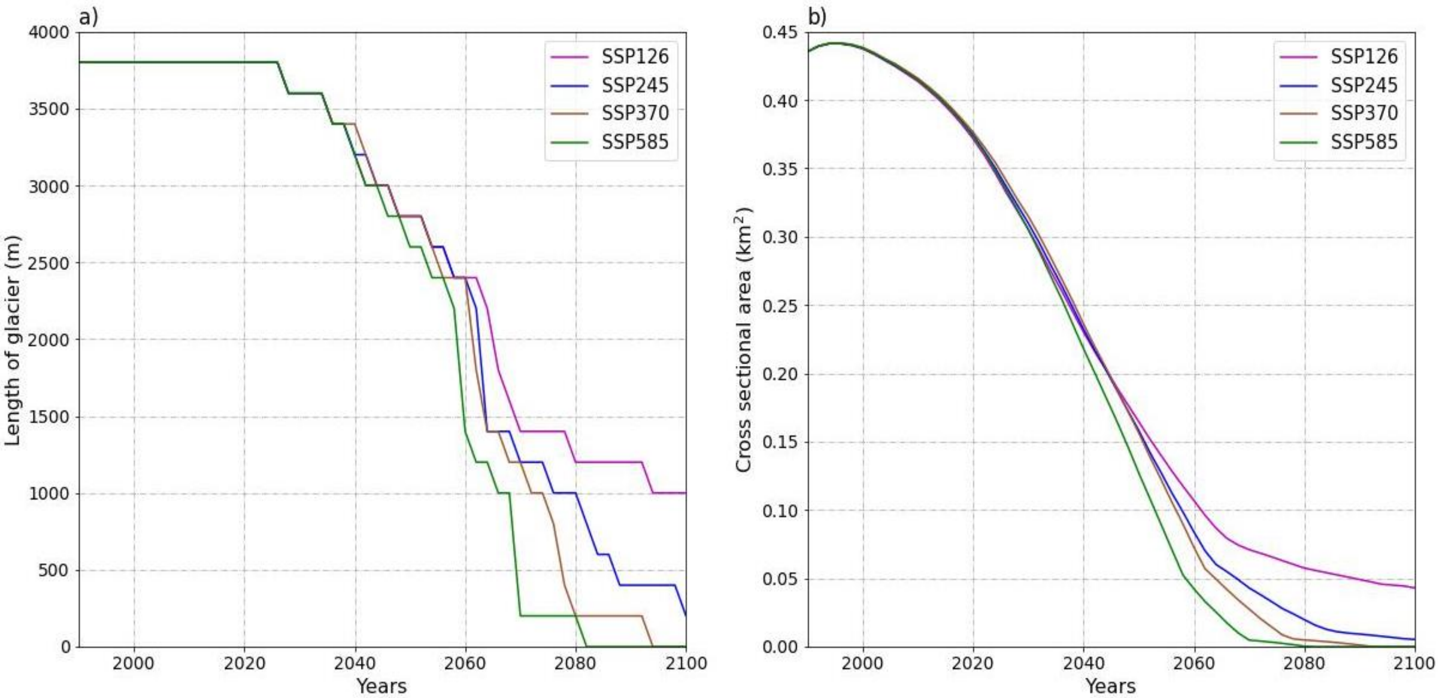



*Figure 5. Future projections of the length (graph 'a') and cross-sectional area (graph 'b') of the Haut Glacier d'Arolla under SSP126, SSP245, SSP370 and SSP585 (business as usual) climate projections. Up to year 2100, based on the 1990 glacier model values.*

Graph 'a' in Figure 5 shows and rapid decreases in the length of the glacier after 2027 under all scenarios. For SSP126, the glacier length decreases by 73.7% by 2100, and for SSP585 length decreases by 100% by year 2082, with the other scenarios distributed between. Graph 'b' shows a decrease of 100% for SSP370 and SSP585

The model predicts for the glacier to completely melt by 2082 under SSP585, and all other projections show steady mass loss, it can be expected that the glacier will be mostly gone within a century. The current summer melt length is 100 days (Sharp et al., 1993), but as mountain region climates warm this will become. This means that the ELA will increase, shortening the glacier because 1°C temperature increase causes the ELA to rise by 125m (Wallinga & Van de Wal, 1998), this will have varying spatial affects across different temperate glaciers.

**Uncertainty**

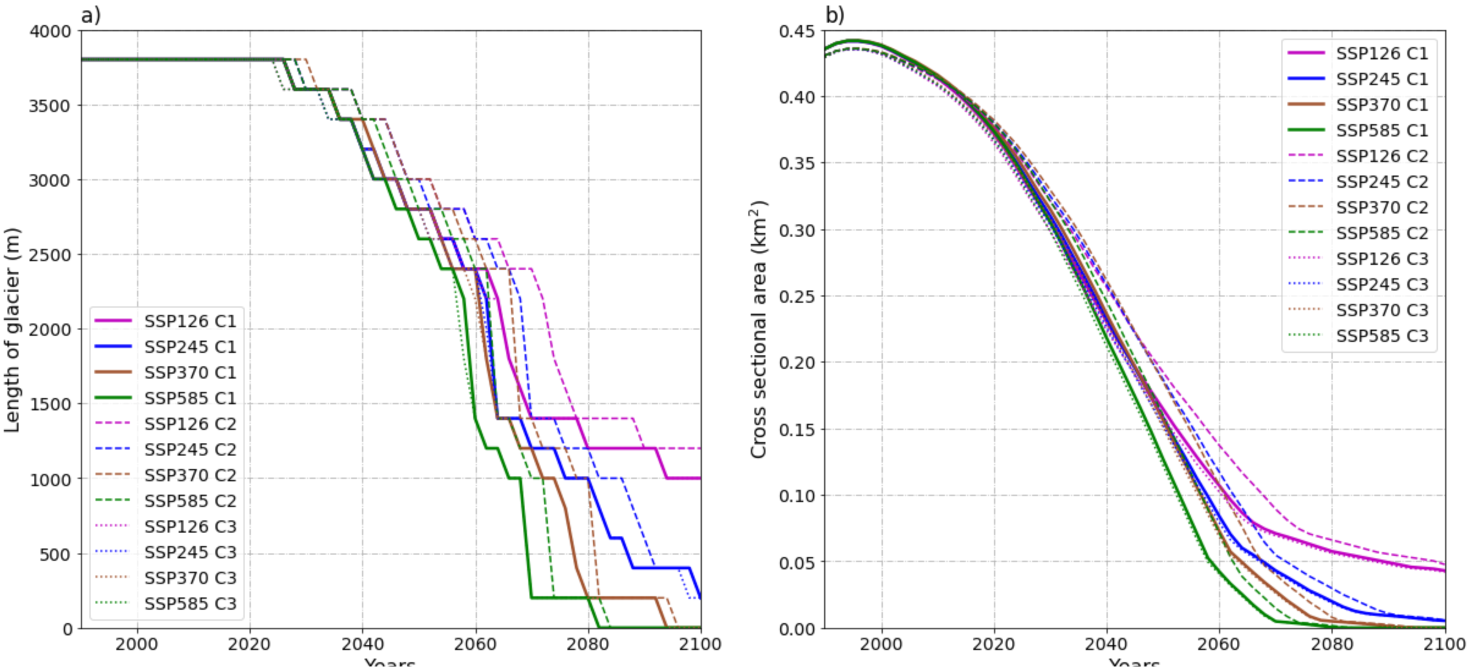



*Figure 6. Future projection uncertainty of the length (graph a) and cross-sectional area (graph b) of the Haut Glacier d'Arolla under SSP126, SSP245, SSP370 and SSP585 projections. Up to year 2100, based on the 1990 glacier model values. Three optimum parameter combinations are plotted, combination 1 is the solid line, combination 2 is the dashed line, and combination 3 is the dotted line.*

*Figure 7. Optimum parameter combinations inputted into the model for future glacier predictions*

| Parameter combination | Flow parameter (A) ($yr^{-1}$ $Pa^{-3}$) | Mass balance gradient (M) ($yr^{-1}$) |
|---|---|---|
| Combination 1 (C1) | 1.75e – 16 | 0.0145 |
| Combination 2 (C2) | 1.5e – 16 | 0.012 |
| Combination 3 (C3) | 1.85e – 16 | 0.0145 |

Figure 7 is based off parameter combination 1 (C1) as C1 has the lowest error value in the calibration analysis. For both graph 'a' and 'b' all C2s compared to C1s project overall that there will be not as rapid glacier melt. This may be due to C2 having the lowest A value (1.5 x $10^{-16}$ $yr^{-1}$ $Pa^{-3}$); a slower flow rate meaning less of the glacier reaches the ablation zone, so less ice mass will be lost. C3 has very similar projections to C1, due to it having the same M value, and A value being very similar. For graph 'b', all projections predict that cross-sectional area will decrease <0.05$km^2$.

## References


Goodsell, B., Hambrey, M. J., Glasser, N. F., Nienow, P., and mair, M. (2018) 'The structural Glaciology of a temperature Glacier: Haut Glacier d'Arolla, Switzerland', *Arctic, Antarctic and Alpine Research*, Vol. 37, pp. 218-232

Arnold, N. (2018) 'Investigating the Sensitivity of Glacier Mass Balance/Elevation Profiles to Changing Meteorological Conditions: Model Experiments for Haut Glacier D'Arolla, Valais, Switzerland', *Arctic, Antarctic and Alpine Research,* Vol. 37, pp. 139-145

GLAMOS Glacier Modelling Switzerland (2022) *Swiss Glaciers*, Available at: https://www.glamos.ch/en/#/E23-16 (Accessed 23/11/2022)

Sharp, M., Richards, K., Willis, I., Arnold, N., Nienow, P., Lawson, W., and Tison, J.L. (1993) 'Geometry, Bed Topography and Drainage System Structure of the Haut Glacier d'Arolla, Switzerland', *Earth Surface Processes and Landforms, Vol.* 18(6), pp. 557-571

Douglas, B. (2014), *Glaciers and Glaciation*. Available at: https://www.vlebooks.com/Product/Index/347630?page=0 (Accessed 23/11/2022)


## Appendix

Appendix 1: Code to produce Figure 2 – Sensitivity Analysis

```
# sensitivity analysis plot for ELA, flow parameter and mass balance gradient of
the Haut Glacier d'Arolla
# parameters are plotted against length of glacier and cross sectional area of
glacier
# code written by 710013703
# thanks to David H-G

#import python packages
import numpy as np
import matplotlib.pyplot as plt

## Load data from saved arrays into numpy
sensitivity_ELA_table = np.load('sensitivity_ELA_table.npy')
sensitivity_FlowParamA_table = np.load('sensitivity_FlowParamA_table.npy')
sensitivity_mBalGrad_table = np.load('sensitivity_mBalGrad_table.npy ')


# create figures to hold the subplots
fig = plt.figure(figsize=(15, 20))

### add a 3 row 1 column subplot
## plot subplots 1 to 3 for length and cross sectional area
# ax(1-3)Plot2 is the second y axis on the right
# label the axis
# set axis tick limits and font size
# create and label legend
# add gridlines
# add title

# subplot 1 - ELA Length and cross sectional area sensitivity subplot
ax1 = fig.add_subplot(3,1,1)
axPlot1 = ax1.plot(sensitivity_ELA_table[:,0],sensitivity_ELA_table[:,1],
marker="o",color='g',label = 'Length')
```

```
ax1_dual = ax1.twinx()
axPlot2 = ax1_dual.plot(sensitivity_ELA_table[:,0],sensitivity_ELA_table[:,2],
marker="^",color='m',label ='Cross sectional area')
ax1.set_xlabel('Equilibrum line altitude (m)',size=15)
ax1.set_ylabel('Length of glacier (m)',size=15)
ax1_dual.set_ylabel('Cross sectional area of glacier (km$^2$)', size=15)
ax1_dual.set_ylim([0,1.2])
ax1.set_ylim([0,6000])
ax1.tick_params(axis='y', labelsize=14)
ax1_dual.tick_params(axis='y', labelsize=14)
ax1.tick_params(axis='x', labelsize=14)
ax1.legend(handles=axPlot1+axPlot2)
ax1.grid(True,linestyle = "-.")
plt.title('a)',loc = 'left',fontsize=16)

# subplot 1 - Flow parameter Length and cross sectional area sensitivity subplot
ax2 = fig.add_subplot(3,1,2)
axPlot1 = ax2.plot(sensitivity_FlowParamA_table[:,0],sensitivity_FlowParamA_ta-
ble[:,1], marker="o",color='g',label = 'Length')
ax2_dual = ax2.twinx()
axPlot2 = ax2_dual.plot(sensitivity_FlowParamA_table[:,0],sensitivity_FlowPar-
amA_table[:,2], marker="^",color='m',label ='Cross sectional area')
ax2.set_xlabel('Flow Parameter (Pa$^{-3}$ yr$^{-1}$)',size=15)
ax2.set_ylabel('Length of glacier (metres)',size=15)
ax2_dual.set_ylabel('Cross sectional area of glacier (km$^2$)', size=15)
ax2_dual.set_ylim([0,1.2])
ax2.set_ylim([0,6000])
ax2.tick_params(axis='y', labelsize=14)
ax2_dual.tick_params(axis='y', labelsize=14)
ax2.tick_params(axis='x', labelsize=14)
ax2.legend(handles=axPlot1+axPlot2)
ax2.grid(True,linestyle = "-.")
plt.title('b)',loc = 'left',fontsize=15)

# subplot 3 - Mass balance gradient Length and cross sectional area sensitvity
subplot
ax3 = fig.add_subplot(3,1,3)
axPlot1 = ax3.plot(sensitivity_mBalGrad_table[:,0],sensitivity_mBalGrad_ta-
ble[:,1], marker="o",color='g',label = 'Length')
ax3_dual = ax3.twinx()
axPlot2 = ax3_dual.plot(sensitivity_mBalGrad_table[:,0],sensitivity_mBalGrad_ta-
ble[:,2], marker="^",color='m',label ='Cross sectional area')
ax3.set_xlabel('Mass balance gradient (yr$^{-1}$)',size=15)
ax3.set_ylabel('Length of glacier (metres)',size=15)
ax3_dual.set_ylabel('Cross sectional area of glacier (km$^2$)', size=15)
ax3_dual.set_ylim([0,1.2])
ax3.set_ylim([0,6000])
ax3.tick_params(axis='y', labelsize=14)
ax3_dual.tick_params(axis='y', labelsize=14)
ax3.tick_params(axis='x', labelsize=14)
ax3.legend(handles=axPlot1+axPlot2)
ax3.grid(True,linestyle = "-.")
plt.title('c)',loc = 'left',fontsize=15)

#save the figures as jpg file
plt.savefig('sensitivity_analysis_subplots.jpg')
```

Appendix 2: Code to produce Figure 4 – Model calibration

```
#colour plots of length error and RMSE error measures compared to 1990 observed
Haut glacier d'Arolla
#showing different parameter value combinations
```

```
#code written Anne Le Brocq


#import python packages
import numpy as np
import matplotlib.pyplot as plt

#Load parameter values array into numpy
FlowParamA_table = np.load('FlowParamA_table.npy')
mBalGrad_table = np.load('mBalGrad_table.npy')
length_error_table = np.load('length_error_table.npy')
rmse_table = np.load('rmse_table.npy')

#create variable for the highest non nan value in the length error table
length_error_max = np.nanmax(length_error_table)
dx = 200

#set up the colour bar, including colour scale and tick marks
colorbar_intervals_num = int(length_error_max/dx)+1
colorbar_ticks = range(0,int(length_error_max+dx),dx)
colorbar_llimit = 0-(dx/2)
colorbar_ulimit = length_error_max+(dx/2)

#setting up parameter labels variables and tick marks starting from 0
FlowParamA_labels = FlowParamA_table[0,:]
FlowParamA_ticks = np.arange(0,len(FlowParamA_labels))
mBalGrad_labels = mBalGrad_table[:,0]
mBalGrad_ticks = np.arange(0,len(mBalGrad_labels))

#create the contour subplots and plot size
fig = plt.figure(figsize=(6, 12))

##add a 1 row 2 column subplots
#plot subplots for length error and RMSE
#set colour bar colour and tick mark intervals
#plot colour bar chart (cbar1-2) with tick marks and labels
#set the loaction and names of the parameter tick marks on the axis of the plot
#rotate x axis tick marks by 45 degrees for easier reading
#label the axis and add a title for analysis of figures

##assigning vairable ax1 to plot suplot 1
ax1 = fig.add_subplot(2,1,1)
cmap1 = plt.cm.get_cmap('viridis', colorbar_intervals_num)
plt.imshow(length_error_table,origin='lower',cmap=cmap1)
cbar1 = plt.colorbar(ticks=colorbar_ticks)
cbar1.set_label('Glacier Length Error (m)')
plt.clim(colorbar_llimit,colorbar_ulimit)
ax1.set_xticks(FlowParamA_ticks)
ax1.set_xticklabels(FlowParamA_labels,size=9)
ax1.set_yticks(mBalGrad_ticks)
ax1.set_yticklabels(mBalGrad_labels)
plt.xticks(rotation=45)
ax1.set_xlabel('Flow Parameter (Pa$^{-3}$ yr$^{-1}$)')
ax1.set_ylabel('Mass Balance Gradient (yr$^{-1}$)')
plt.title('a)',loc = 'left',fontsize=14)

##assigning variable ax2 to plot subplot 2
ax2 = fig.add_subplot(2,1,2)
plt.imshow(rmse_table,origin='lower')
cbar2 = plt.colorbar()
cbar2.set_label('RMSE (m)')
ax2.set_xticks(FlowParamA_ticks)
```

```
ax2.set_xticklabels(FlowParamA_labels,size=9)
ax2.set_yticks(mBalGrad_ticks)
ax2.set_yticklabels(mBalGrad_labels)
plt.xticks(rotation=45)
ax2.set_xlabel('Flow Parameter (Pa$^{-3}$ yr$^{-1}$)')
ax2.set_ylabel('Mass Balance Gradient (yr$^{-1}$)')
plt.title('b)',loc = 'left',fontsize=14)

#save the figures as jpg file
plt.savefig('calibration_figure.jpg')
```

Appendix 3: Code to produce figure 5 – Future Projections

```
#Future projections of the 1990 Haut Glacier d'Arolla length and cross sectional area
#Showing SSP126, SSP245, SSP370, and SSP585 climate scenarios
#The model was run using parameter combination of flow parameter 1.75-16 (yr-1Pa-3) and

#import python packages
import numpy as np
import matplotlib.pyplot as plt

### Load data for projected length and cross sectional area arrays into numpy
Future_Length126_table = np.load('FutureLength126.npy')
Future_Length245_table = np.load('FutureLength245.npy')
Future_Length370_table = np.load('FutureLength370.npy')
Future_Length585_table = np.load('FutureLength585.npy')
Future_X126_table = np.load('FutureXArea126.npy')
Future_X245_table = np.load('FutureXArea245.npy')
Future_X370_table = np.load('FutureXArea370.npy')
Future_X585_table = np.load('FutureXArea585.npy')

# create figures to hold the subplots
fig = plt.figure(figsize=(20, 9))

##plot subplots 1 and 2
#label the x and y axis and set the font size
#create a legend for each subplot
#label each subplot
#set limits for x and y axis
#set the font size of the tick marks

#this is subplot 1 for glacier length
ax1 = fig.add_subplot(1,2,1)
ax1.grid()
ax1.plot(Future_Length126_table[:,0],Future_Length126_table[:,1], color='m')
ax1.plot(Future_Length245_table[:,0],Future_Length245_table[:,1], color='b')
ax1.plot(Future_Length370_table[:,0],Future_Length370_table[:,1], color='sienna')
ax1.plot(Future_Length585_table[:,0],Future_Length585_table[:,1], color='g')
ax1.set_xlabel('Years',size=16)
ax1.set_ylabel('Length of glacier (m)',size=16)
ax1.legend(['SSP126','SSP245','SSP370','SSP585'],fontsize=15)
plt.title('a)',loc = 'left',fontsize=17)
ax1.set_xlim([1990,2100])
ax1.set_ylim([0,4000])
plt.xticks(fontsize=14)
plt.yticks(fontsize=14)
ax1.grid(True,linestyle = "-.")

#this is subplot 2 for glacier length
ax2 = fig.add_subplot(1,2,2)
ax2.grid()
ax2.plot(Future_X126_table[:,0],Future_X126_table[:,1], color='m')
```

```
ax2.plot(Future_X245_table[:,0],Future_X245_table[:,1], color='b')
ax2.plot(Future_X370_table[:,0],Future_X370_table[:,1], color='sienna')
ax2.plot(Future_X585_table[:,0],Future_X585_table[:,1], color='g')
ax2.set_xlabel('Years',size=16)
ax2.set_ylabel('Cross sectional area (km$^2$)',size=16)
ax2.legend(['SSP126','SSP245','SSP370','SSP585'],fontsize=15)
plt.title('b)',loc = 'left', fontsize=17)
ax2.set_xlim([1990,2100])
ax2.set_ylim([0,0.45])
plt.xticks(fontsize=14)
plt.yticks(fontsize=14)
ax2.grid(True,linestyle = "-.")

#save the figures as jpg file
plt.savefig('future_scenarios_subplots.jpg')

```

Appendix 4: Code to produce figure 6 – Uncertainty

```
#Future projections of the 1990 Haut Glacier d'Arolla length and cross sectional
area
#Showing SSP126, SSP245, SSP370, and SSP585 climate scenarios

##Model was run using three parameter combinations that have minimal error to
1990 glacier
#parameter combination 1 (C1) is flow parameter 1.75-16 (yr-1Pa-3) and mass bal-
ance gradient 0.0145 (yr-1)
#parameter combination 2 (C2) is flow parameter 1.5-16 (yr-1Pa-3) and mass bal-
ance gradient 0.012 (yr-1)
#parameter combination 3 (C3) is flow parameter 1.85-16 (yr-1Pa-3) and mass bal-
ance gradient 0.0145 (yr-1)


#import python packages
import numpy as np
import matplotlib.pyplot as plt

### Load data for projected length and cross sectional area arrays into numpy

#load parameter combination 1
Future_Length126_table = np.load('FutureLength126.npy')
Future_Length245_table = np.load('FutureLength245.npy')
Future_Length370_table = np.load('FutureLength370.npy')
Future_Length585_table = np.load('FutureLength585.npy')
Future_X126_table = np.load('FutureXArea126.npy')
Future_X245_table = np.load('FutureXArea245.npy')
Future_X370_table = np.load('FutureXArea370.npy')
Future_X585_table = np.load('FutureXArea585.npy')

#load parameter combination 2
C2Future_Length126_table = np.load('C2FutureLength126.npy')
C2Future_Length245_table = np.load('C2FutureLength245.npy')
C2Future_Length370_table = np.load('C2FutureLength370.npy')
C2Future_Length585_table = np.load('C2FutureLength585.npy')
C2Future_X126_table = np.load('C2FutureXArea126.npy')
C2Future_X245_table = np.load('C2FutureXArea245.npy')
C2Future_X370_table = np.load('C2FutureXArea370.npy')
C2Future_X585_table = np.load('C2FutureXArea585.npy')

#load parameter combination 2
C3Future_Length126_table = np.load('C3FutureLength126.npy')
```

```
C3Future_Length245_table = np.load('C3FutureLength245.npy')
C3Future_Length370_table = np.load('C3FutureLength370.npy')
C3Future_Length585_table = np.load('C3FutureLength585.npy')
C3Future_X126_table = np.load('C3FutureXArea126.npy')
C3Future_X245_table = np.load('C3FutureXArea245.npy')
C3Future_X370_table = np.load('C3FutureXArea370.npy')
C3Future_X585_table = np.load('C3FutureXArea585.npy')

# create figures to hold the subplots
fig = plt.figure(figsize=(20, 9))

##plot subplots, from collumn 1 for the x axis and collumn 2 for y axis of the
array
#label the x and y axis and set the font size
#create a legend for each subplot
#label each subplot
#set limits for x and y axis
#set the font size of the tick marks
#add a dashed gridlines

#this is subplot 1, showing all parameter combinations and scenarios for glacier
length
ax1 = fig.add_subplot(1,2,1)
ax1.grid()
ax1.plot(Future_Length126_table[:,0],Future_Length126_table[:,1], color='m',lin-
ewidth=2.5)
ax1.plot(Future_Length245_table[:,0],Future_Length245_table[:,1], color='b',lin-
ewidth=2.5)
ax1.plot(Future_Length370_table[:,0],Future_Length370_table[:,1], color='sien-
na',linewidth=2.5)
ax1.plot(Future_Length585_table[:,0],Future_Length585_table[:,1], color='g',lin-
ewidth=2.5)
ax1.plot(C2Future_Length126_table[:,0],C2Future_Length126_table[:,1],
color='m',linestyle='dashed')
ax1.plot(C2Future_Length245_table[:,0],C2Future_Length245_table[:,1],
color='b',linestyle='dashed')
ax1.plot(C2Future_Length370_table[:,0],C2Future_Length370_table[:,1], color='si-
enna',linestyle='dashed')
ax1.plot(C2Future_Length585_table[:,0],C2Future_Length585_table[:,1],
color='g',linestyle='dashed')
ax1.plot(C3Future_Length126_table[:,0],C3Future_Length126_table[:,1],
color='m',linestyle='dotted')
ax1.plot(C3Future_Length245_table[:,0],C3Future_Length245_table[:,1],
color='b',linestyle='dotted')
ax1.plot(C3Future_Length370_table[:,0],C3Future_Length370_table[:,1], color='si-
enna',linestyle='dotted')
ax1.plot(C3Future_Length585_table[:,0],C3Future_Length585_table[:,1],
color='g',linestyle='dotted')
ax1.set_xlabel('Years',size=16)
ax1.set_ylabel('Length of glacier (m)',size=16)
ax1.legend(['SSP126 C1','SSP245 C1','SSP370 C1','SSP585 C1','SSP126 C2','SSP245
C2','SSP370 C2','SSP585 C2', 'SSP126 C3','SSP245 C3','SSP370 C3','SSP585
C3'],fontsize=14)
plt.title('a)',loc = 'left',fontsize=17)
ax1.set_xlim([1990,2100])
ax1.set_ylim([0,4000])
plt.xticks(fontsize=14)
plt.yticks(fontsize=14)
ax1.grid(True,linestyle = "-.")
```

```
#this is subplot 2, showing all parameter combinations and scenarios for glacier
length
ax2 = fig.add_subplot(1,2,2)
ax2.grid()
ax2.plot(Future_X126_table[:,0],Future_X126_table[:,1], color='m',linewidth=2.5)
ax2.plot(Future_X245_table[:,0],Future_X245_table[:,1], color='b',linewidth=2.5)
ax2.plot(Future_X370_table[:,0],Future_X370_table[:,1], color='sienna',lin-
ewidth=2.5)
    ax2.plot(Future_X585_table[:,0],Future_X585_table[:,1], color='g',linewidth=2.5)
    ax2.plot(C2Future_X126_table[:,0],C2Future_X126_table[:,1], color='m',lin-
    estyle='dashed')
    ax2.plot(C2Future_X245_table[:,0],C2Future_X245_table[:,1], color='b',lin-
    estyle='dashed')
    ax2.plot(C2Future_X370_table[:,0],C2Future_X370_table[:,1], color='sienna',lin-
    estyle='dashed')
    ax2.plot(C2Future_X585_table[:,0],C2Future_X585_table[:,1], color='g',lin-
    estyle='dashed')
    ax2.plot(C3Future_X126_table[:,0],C3Future_X126_table[:,1], color='m',lin-
    estyle='dotted')
    ax2.plot(C3Future_X245_table[:,0],C3Future_X245_table[:,1], color='b',lin-
    estyle='dotted')
    ax2.plot(C3Future_X370_table[:,0],C3Future_X370_table[:,1], color='sienna',lin-
    estyle='dotted')
    ax2.plot(C3Future_X585_table[:,0],C3Future_X585_table[:,1], color='g',lin-
    estyle='dotted')
    ax2.set_xlabel('Years',size=16)
    ax2.set_ylabel('Cross sectional area (km$^2$)',size=16)
    ax2.legend(['SSP126 C1','SSP245 C1','SSP370 C1','SSP585 C1','SSP126 C2','SSP245
    C2','SSP370 C2','SSP585 C2', 'SSP126 C3','SSP245 C3','SSP370 C3','SSP585
    C3'],fontsize=14)
    plt.title('b)',loc = 'left',fontsize=17)
    ax2.set_xlim([1990,2100])
    ax2.set_ylim([0,0.45])
    plt.xticks(fontsize=14)
    plt.yticks(fontsize=14)
    ax2.grid(True,linestyle = "-.")


    #save the figures as jpg file
    plt.savefig('uncertainty_merging_graphs_subplots.jpg')
```