Alexandra Pulwicki
August 23, 2016

Methods
Data Analysis

Transect Data Import WP to matlab + quality Converting to SWE??

Zigzag Data Import distances and vertex coordinates to matlab + quality Calculate location of measurements Options 1 and 2 SWE from tube values

Density Data Federal sampler Quality decisions Averaging of tube values Snowpit Integrated density

1 Data Processing

1.1 Linear and Curvilinear Transects

Snow depth measurements along the linear and curvilinear transects were taken at locations a certain distance from marked waypoints. Since only the coordinates of the waypoints (WP) were recorded, the measurement coordinates needed to be estimated. The measurement locations were assumed to be 10, 20, and 30 m behind the marked WP, in a straight line between the marked WP and the previous WP. In cases with only two observers, locations were assumed to be 10 and 20 m behind the marked WP. For the first marked WP of a pattern, it was assumed that the locations were along the same line as that between the first and second WPs. The following methodology was used to determine measurement locations (each step corresponds with a section of the Matlab code 'MeasurementLocations.m'):

1. Waypoint (WP) locations were exported from the GPS units using the BaseCamp program. They were then imported to QGIS and exported with UTM coordinates. This file was then used in the Matlab script and is entitled 'GlacierWP_UTM.xlsx'.
2. In order to obtain the measurement locations for the first WPs of each pattern, an "imaginary" WP was created that was along the line between the first and second WPs, but located ahead of the first WP. These waypoints were then inserted into the original data.
3. A set of 1000 equally spaced points was created along a straight line between each set of subsequent of WPs (including the "imaginary" WPs from the previous step) using the function linspaceNDim.m created by Steeve Ambroise and downloaded from the MathWorks File Exchange. The Euclidean distance between these interpolated points and the marked WP was then calculated and the points with distances closest to the assumed separation between observers were retained. The final matrix has the easting and northing of each measurement location and is labelled with the marked WP and a decimal that corresponds to the relative observer (e.g. label 45.2 means

that the location was determined from the marked WP 45 and is 20 m behind this WP because it is the second observer).

The data recorded by each observer in the field books was transcribed to a spreadsheet format and then imported and processed in Matlab according to the following steps (each step corresponds with a section of the Matlab code 'Import_Transect.m'):

1. A spreadsheet was created with a sheet for data from each field book (SD#1, SD#2, SD#3, and SWEDepth). For each reference WP there were values for all snow depth measurements and their quality (1 for good, 0 for bad), comments written, field book name, glacier name, observer, pattern, and date collected.
2. The quality, comments, book name, glacier name, observer, pattern, and date entires were categorized. This allows for efficient grouping and data searching in future analysis.
3. The depth data was then assigned the corresponding measurement location UTM from the 'MeasurementLocations.m' script. This was done by matching the WP number from the field books and that of the marked WPs and then assigning the coordinates from the WP ending with .1 to depths recorded in book SD#1, and likewise for the remaining books. The matrices for each set of observations were then made to be the same dimensions by inserting empty cells for WPs where no data was recorded in that set of observations.
4. The data was then arranged in a structure variable (called SD) with rows corresponding to each book (e.g. row 1 is data from book SD#1) and columns corresponding to the various types of data (e.g. depth values or glacier category). For example, the matrix with the glacier category for each value recorded in the book SD#1 can be accessed with 'SD(1).glacier'.

Subsets of the transect data can be pulled using the function 'pull data.m'. It employs the following steps to find desired data (corresponds with the Matlab code 'pulldata.m')::

1. The function is called with `pulldata(data, book, glacier, person, pattern, quality, format)`