CRONUScalc Function Appendix

Shasta Marrero
University of Edinburgh
shastamarrero@gmail.com

June 12, 2015

This document was written to accompany the CRONUScalc MATLAB code published in Quaternary Geochronology. This document details each function included in the MATLAB code and THIS DOCUMENT IS NOT WHOLLY APPLICABLE TO THE ONLINE INTERFACE. All references to input/output parameters apply to the internal, raw code and not to the online interface. For information about the interface, see the notes on the website. All functions are described in detail (inputs, outputs, how each functions), except where functions are already described in detail in Balco et al. (2008). The CRONUScalc program matching this paper (version 2.0) and all future updated code can be freely downloaded from the bitbucket repository (`https://bitbucket.org/cronusearth/cronus-calc`).

The references for the original Balco et al. (2008) paper and the accompanying supplemental information:

Balco, G., Stone, J. O., Lifton, N., and Dunai, T. J. (2008). A complete and easily accessibly means of calculating surface exposure ages or erosion rates from ^{10}Be and ^{26}Al measurements. Quaternary Geochronology, 3:174–195.

Balco, G. (2007). CRONUS-Earth ^{26}Al-^{10}Be exposure age calculator MATLAB function reference Version 2. `http://hess.ess.washington.edu/math/docs/al_be_v2/al_be_fctn_desc.pdf`

1 Inputs & Outputs

List of inputs for all nuclides. Lists of parameters necessary for each nuclide are included after the general list of possible inputs. For calculations of most sample exposure ages and erosion rates, a main variable is required with the nominal inputs (listed below for each nuclide) and a second variable

with the uncertainties associated with each input (in the same order) is also required. In general, we assume that the inputs/uncertainties are uncorrelated. For ^{36}Cl, there is an additional covariance variable to describe the correlation between Cl concentration and ^{36}Cl concentration in the sample (see `spiketoconc.m` for more details).

1. Sample concentration (atoms/g of target) - This should be the blank-subtracted concentration of the sample. For Be, samples are assumed to be normalized to standard 07KNSTD and for Al, samples are assumed to be normalized to standard KNSTD. See Balco et al. (2008) and Nishiizumi et al. (2007) for detailed discussions concerning these normalizations.

2. Inheritance (atoms/g of target) - This is zero except in some rare instances.

3. Erosion-rate epsilon ((g/cm^2)/kyr) - This is the erosion rate in mass depth units. Do not use mm/kyr or similar units.

4. Bulk density (g/cm^3) - This is the density of the sample. For depth profiles, use the individual layer density here.

5. Sample thickness (cm) - This is the thickness of the sample that was used for analysis. For depth profiles, use the thickness of the collected layer here.

6. Latitude (decimal degrees)

7. Longitude (decimal degrees)

8. Elevation (meters) - This is only used at one point to calculate muon scaling. Pressure is the input used to determine atmospheric depth and scaling.

9. Pressure (hPa) - The pressure should be calculated from the elevation unless it has been directly measured. The current recommended model is ERA40 (see MATLAB function `ERA40atm.m`).

10. Shielding factor for terrain, snow, etc. (unitless) - This is the shielding correction factor. If necessary, use the `topooriginal.m` file to calculate this value.

11. $\lambda_{f,e}$ Effective neutron attenuation length (g/cm^2) - The effective attenuation length should account for the location of the sample (use the `attenuationlength.m` or `attenuationlengthraw.m` file as an initial estimate of the attenuation length. This value should then be adjusted to account for the changes in the horizon.)

12. Depth to top of sample (g/cm^2) - This is the depth to the top of the sample in units of mass depth. For depth profiles, the depth to top of sample should incorporate any changes in density throughout the profile. This should be zero for surface samples.

13. Year Collected (AD) - This is the calendar year the sample was collected.

14. Fractional volumetric water-content (unitless) - This is the pore water content of the rock given as a fraction. Example: A rock that is 1% saturated would have an input value of 0.01.

15. Weight % CO_2 - This is the bulk rock weight %, typically measured using XRF.

16. Weight% Na_2O - This is the bulk rock weight %, typically measured using XRF.

17. Weight% MgO - This is the bulk rock weight %, typically measured using XRF.

18. Weight% Al_2O_3 - This is the bulk rock weight %, typically measured using XRF.

19. Weight% SiO_2 - This is the bulk rock weight %, typically measured using XRF.

20. Weight% P_2O_5 - This is the bulk rock weight %, typically measured using XRF.

21. Weight% K_2O - This is the bulk rock weight %, typically measured using XRF.

22. Weight% CaO - This is the bulk rock weight %, typically measured using XRF.

23. Weight% TiO_2- This is the bulk rock weight %, typically measured using XRF.

24. Weight% MnO - This is the bulk rock weight %, typically measured using XRF.

25. Weight% Fe_2O_3 - This is the bulk rock weight %, typically measured using XRF.

26. Cl (ppm) - This is the blank-subtracted chlorine concentration for the bulk rock. If doing whole-rock chlorine analyses, this should be the same value as the target Cl concentration.

27. B (ppm) - This is the trace element composition for the bulk rock, typically measured using ICP.

28. Sm (ppm) - This is the trace element composition for the bulk rock, typically measured using ICP.

29. Gd (ppm) - This is the trace element composition for the bulk rock, typically measured using ICP.

30. U (ppm) - This is the trace element composition for the bulk rock, typically measured using ICP.

31. Th (ppm) - This is the trace element composition for the bulk rock, typically measured using ICP.

32. Cr (ppm) - This is the trace element composition for the bulk rock, typically measured using ICP.

33. Li (ppm) - This is the trace element composition for the bulk rock, typically measured using ICP.

34. Target element %K_2O - This is the target element composition and should be exactly the composition of the dissolved material. This is typically measured by XRF or ICP.

35. Target element %CaO - This is the target element composition and should be exactly the composition of the dissolved material. This is typically measured by XRF or ICP.

36. Target element %TiO_2- This is the target element composition and should be exactly the composition of the dissolved material. This is typically measured by XRF or ICP.

37. Target element %Fe_2O_3- This is the target element composition and should be exactly the composition of the dissolved material. This is typically measured by XRF or ICP.

38. Target element Cl (ppm) - This is the blank-subtracted chlorine concentration for the sample. If doing whole-rock chlorine analyses, this should be the same value as the bulk rock Cl concentration.

1.1 Beryllium-10/Aluminum-26 Inputs

1. Latitude (decimal degrees)
2. Longitude (decimal degrees)
3. Elevation (meters)
4. Pressure (hPa)
5. Sample thickness (cm)
6. Bulk density (g/cm^3)
7. Shielding factor (unitless)
8. Erosion rate ($(g/cm^2)/kyr$)
9. ^{10}Be concentration (at ^{10}Be/g target)
10. ^{26}Al concentration (at ^{26}Al/g target)
11. ^{10}Be Inheritance (at ^{10}Be/g target)
12. ^{26}Al Inheritance (at ^{26}Al/g target)
13. Effective attenuation length (g/cm^2)
14. Depth to top of sample (g/cm^2)
15. Year collected (AD)

1.2 Helium-3 Inputs

1. Latitude (decimal degrees)
2. Longitude (decimal degrees)
3. Elevation (meters)
4. Pressure (hPa)
5. Sample thickness (cm)
6. Bulk density (g/cm^3)
7. Shielding factor (unitless)
8. Erosion rate ($(g/cm^2)/kyr$)
9. ^3He concentration (at ^3He/g target)
10. Inheritance (at ^3He/g target)
11. Effective attenuation length (g/cm^2)
12. Depth to top of sample (g/cm^2)
13. Year collected (AD)

1.3 Carbon-14 Inputs

1. Latitude (decimal degrees)
2. Longitude (decimal degrees)
3. Elevation (meters)
4. Pressure (hPa)
5. Sample thickness (cm)
6. Bulk density (g/cm^3)
7. Shielding factor (unitless)
8. Erosion rate ($(g/cm^2)/kyr$)
9. ^{14}C concentration (at ^{14}C/g target)

10. Inheritance (at ^{14}C/g target)
11. Effective attenuation length (g/cm^2)
12. Depth to top of sample (g/cm^2)
13. Year collected (AD)

1.4 Chlorine-36

The inputs for chlorine-36 include three variables for regular samples and an additional independent age variable for calibration samples. The four possible inputs include `nominal36`, `uncerts36`, `cov36`, and `indages36`. `nominal36` is set up with one row for each sample and each column is a different input parameter, as defined below.

1. Sample ^{36}Cl concentration (atoms of ^{36}Cl/g of target)
2. Inheritance (atoms 36-Cl/g of target)
3. Erosion-rate epsilon ((g/cm^2)/kyr)
4. Fractional volumetric water-content (unitless)
5. Bulk density (g/cm^3)
6. Sample thickness (cm)
7. Latitude (decimal degrees)
8. Longitude (decimal degrees)
9. Elevation (meters)
10. Pressure (hPa)
11. Shielding factor (unitless)
12. $\lambda_{f,e}$ Effective neutron attenuation length (g/cm^2)
13. Weight % CO_2, bulk rock
14. Weight% Na_2O, bulk rock
15. Weight% MgO, bulk rock
16. Weight% Al_2O_3, bulk rock

17. Weight% SiO_2, bulk rock
18. Weight% P_2O_5, bulk rock
19. Weight% K_2O, bulk rock
20. Weight% CaO, bulk rock
21. Weight% TiO_2, bulk rock
22. Weight% MnO, bulk rock
23. Weight% Fe_2O_3, bulk rock
24. Cl (ppm), bulk rock
25. B (ppm), bulk rock
26. Sm (ppm), bulk rock
27. Gd (ppm), bulk rock
28. U (ppm), bulk rock
29. Th (ppm), bulk rock
30. Cr (ppm), bulk rock
31. Li (ppm), bulk rock
32. Target element %K_2O
33. Target element %CaO
34. Target element %TiO_2
35. Target element %Fe_2O_3
36. Target element Cl (ppm)
37. Depth to top of sample (g/cm^2)
38. Year Collected (AD)

2 MATLAB Functions

The MATLAB code, named CRONUScalc, is organized into several folders based on the type of calculation being performed. In order to use the complete code, you must use the `addpath` command to add all the folders or select them via the Matlab interface. The function list here follows the structure of the code and is organized into the same categories with specific descriptions of each function included. In many cases, there are similar code pieces for chlorine-36, aluminum-26, beryllium-10, etc. When the descriptions for the same, the nuclide number will be replaced with 'XX' to indicate that it applies to all the nuclides. For example, there are several codes for calibrating each nuclide including `calibrate10`, `calibrate26`, `calibrate36`, etc. These codes are discussed as a group under the heading `calibrateXX`. A similar method, only using 'aa' instead of 'XX', is used for code that is named using the two-letter indicator for the nuclide. For example, `al26uncert.m` becomes `aaXXuncert.m` when the generic code is discussed.

2.1 Calib Folder

The `calib` folder contains code used to perform calibrations and age the primary and secondary datasets after calibration.

2.1.1 `agecalibsetXX.m`

This code computes ages for the calibration dataset or another dataset formatted in the same way.

Inputs: There are no inputs to this code. The calibration dataset is hard-coded at the beginning in the line that starts with "load." The required input is the "calibration input" that consists of three variables. The three variables are nominalXX, uncertsXX, and indagesXX. For chlorine-36, there is an additional variable called cov36. These are covered in detail in the excelformatting section.

Details: The code first collects some basic information about the independent ages of the samples. The uncertainties on the concentrations are updated using the `uncertsXX` function. The ages and nominal uncertainties are computed using the `aaXXage.m` function. Other output parameters that are recorded are the scaling factors for the spallation production rate and the percent production from muons based on the contemporary production rates.

The code also calculates the RMSE (root mean square error) for the entire calibration dataset. This is calculated by the following formula: For each sample: percenterror=100*(computedages-INDAGES)/INDAGES; The vector containing the all the percenterror values is the percenterrors variable.

RMSE=sqrt(mean(percenterrors^2))

A variable named 'total' is created that contains one column for each of the following variables:

1. Computed ages
2. Computed uncertainties
3. Spallation scaling factor
4. Average muon scaling factor
5. Percent production from muons

2.1.2 `calibrateXX.m`

This script calibrates the production rate from the calibration dataset. The production rate has varying units that depend on the nuclide. For carbon, beryllium, and aluminum, the units are atoms of nuclide per gram of quartz per year. For helium, the units are atoms of nuclide per gram of target mineral (either pyroxene or olivine). For chlorine, the units are dependent on the pathway that is being calibrated. Units for the calcium and potassium pathways are atoms of chlorine-36 per gram of Ca per year and atoms of chlorine-36 per gram of K per year, respectively. For the low-energy neutron pathway, the units are fast neutrons per gram of air per year (see Marrero (2012) for a more detailed description).

Input: There is no input to this script. The calibration dataset is hard-coded at the beginning in the line that starts with "load." The required input is the "calibration input" that consists of three variables. The three variables are nominalXX, uncertsXX, and indagesXX. For chlorine-36, there is an additional variable called cov36. These are covered in detail in the excelformatting section.

Output: The production rates for the appropriate reaction.

Details: The code first stores some variables into global variables and gathers information about the independent ages of the samples. The uncertainties on the concentrations are updated using the `uncertsXX` function, based on the CRONUS interlaboratory comparison exercises. The

necessary factors are precomputed by calling the `sampparsXX`, `physpars`, `scalefacsXX`, and `compparsXX`. The calibration values are initialized using the variable pinit. The first value in pinit is the starting value for the production rate. The other values in pinit are the concentration residuals (difference between the predicted and measured concentrations). The calibrated production rate is displayed on the screen when the calibration is complete.

The `lm.m` code fits the parameters. The covariance matrix is calculated using `odrfunXX.m` and `odrjacXX.m` and a p-value is calculated.

`maxiter` is the variable that determines the maximum number of iterations to be performed during the calibration. It is set to 100 for final calibrations, but can be modified if desired.

2.1.3 lm.m

Syntax: `[pstar,iter]=lm(func,jac,p0,tol,maxiter)`

Use the Levenberg-Marquardt algorithm to minimize

$$f(x) = sum(F_i(x)^2) \tag{1}$$

Input:

func	Name of the function F(x)
jac	Name of the Jacobian function J(x)
x0	Initial guess
tol	Stopping tolerance
maxiter	Maximum number of iterations allowed

Output:

xstar	Best solution found
iter	Iteration count

2.1.4 odrfunXX.m

Syntax: `fvec=odrfun10(p)`

This function computes the differences between forward predicted ^{10}Be concentrations and the measured ^{10}Be concentrations, normalized by the standard deviations of the individual measured concentrations.

2.1.5 odrjacXX.m

Syntax: `J=odrjac10(p)`

Computes the Jacobian for the calibration problem.

2.1.6 plotcalibsetXX.m

These are all hard-coded to plot the predicted ages and the actual ages for each nuclide and for each particular calibration dataset. These were primarily used for the CRONUS-Earth calibrations and papers.

2.2 predcalibsetXX.m

This script is used to predict the expected concentration of nuclide in a sample with an independent age and then compare this value to the measured value using ratios of concentrations. These were primarily used for the CRONUS-Earth calibrations and papers.

2.3 Excelformatting Folder

This folder contains the Excel template for each nuclide as well as the matlab code that converts from the Excel information block to the appropriate matlab files. The data in the included Excel sheet is already properly formatted to be used with these codes. Instructions are included in each file. In general, the `createageXX.m` functions create only the nominal input parameters and the uncertainties. `createcalibXX.m` creates an additional parameter that contains the independent ages for each sample and the uncertainties on those ages. Finally, `createcalibXXxval.m` creates all the variables from the `createcalibXX.m` code, but adds additional datasets for cross-validation purposes.

Input:

A single block of values, formatted as 1 row per sample and the nominal values and uncertainties (no empty cells in between) in the columns and in the order listed below for each nuclide. If independent ages and uncertainties are used, these are placed at the end of the uncertainties (and after covariance in the case of ^{36}Cl).

2.3.1 createageXX.m

This file creates a properly formatted set of MATLAB variables for aging samples from an Excel sheet based on the templates (discussed below). This file does not create the variable called indagesXX. The template varies based on nuclide. There are also calculations performed on the input data in order

to convert some commonly used units to those used in the MATLAB code. These two issues are covered on a nuclide-by-nuclide basis below.

2.3.2 createcalibXX.m

This file creates a properly formatted set of MATLAB variables for a calibration. This file creates exactly what is created in createageXX.m, but adds the indagesXX variable.

2.3.3 createcalibXXxval.m

This file creates a properly formatted set of MATLAB variables for calibration, but also creates additional datasets that can be used in cross-validation exercises. The datasets are created by removing one site and creating a calibration dataset from the remaining datasets. This is done for each site. The individual sites are named using the three- or four-letter abbreviation for the site and the word 'only', as seen in the following example: PERUonly10.mat. The cross-validation datasets are named by labeling using the following convention: calib10minusPERU.mat.

Beryllium-10 and Aluminum-26 For beryllium-10 and aluminum-26, most inputs are directly copied into the variables during the Excel-to-MATLAB conversion. The code also determines if data is present for aluminum, beryllium, or both and formats the dataset appropriately. The only calculation performed in the code is to convert the given erosion rates (originally in mm/kyr) to new units needed by the code ($g/cm^2/kyr$) using the density provided by the user. The Excel sheet for the Al/Be data uses one sample per row and requires the columns described in Table 1.

Table 1: Excel to MATLAB formatting for ^{10}Be and ^{26}Al

Column	Parameter	Units	Description
1	Latitude	Decimal degrees	Latitude of the sample location.
2	Longitude	Decimal degrees	Longitude of the sample location.

Continued on next page

Table 1 – continued from previous page

Column	Parameter	Units	Description
3	Elevation	masl	Elevation of the sample. The sample elevation is used only for scaling of muons. Sample pressure is used as an indication of atmospheric depth.
4	Pressure	hPa	Sample pressure should be calculated from the elevation using the ERA40 conversion (see section on ERA40atm.m). ERA40 is recommended for use because it was used in the calibration of production rates. For advanced users, the sample pressure can be independently calculated and entered here.
5	Sample thickness	cm	Average sample thickness should be measured for each sample.
6	Bulk density	g/cm^3	The bulk density of the sample. This should be measured for each sample or estimated for each batch of samples.
7	Shielding	unitless	Topographic shielding of the sample. Shielding can be calculated using the topooriginal.m file included in CRONUScalc.
8	Erosion rate	mm/kyr	Erosion rate for the sample surface.
9	Be Conc.	atoms ^{10}Be g^{-1}	Measured concentration of beryllyium-10 after all appropriate blank subtractions
10	Al Con.	atoms ^{26}Al g^{-1}	Measured concentration of aluminum-26 after all appropriate blank subtractions
11	Be Inheritance	atoms ^{10}Be g^{-1}	This is a value for inherited nuclide. This is not used except in unusual cases.

Continued on next page

Table 1 – continued from previous page

Column	Parameter	Units	Description
12	Al Inheritance	atoms ^{26}Al g^{-1}	This is a value for inherited nuclide. This is not used except in unusual cases.
13	Lambda effective	g/cm^2	Effective attenuation length for the sample should include effects of sample dip and topographic shielding. This is also an output of the topooriginal.m file.
14	Depth to top of sample	g/cm^2	This measurement is the depth to the top of the sample. For surface samples, this should be zero. Note the units are not cm, but g/cm^2.
15	Year Collected	Year AD	The year the sample was collected. This is used to properly integrate the production through time.
16-30	Uncerts		Uncertainties for each of the parameters listed above. Important parameters for uncertainties include the concentration and target element concentrations, especially chlorine.
31	Independent age	Years before 2010	Independent age given in years before 2010. This field is required if you are performing a calibration.
32	Independent age uncert	Years	Uncertainty on the independent age in years. This field is required if you entered an independent age for the sample.

Helium-3 and Carbon-14 Carbon-14 and helium-3 are formatted using different codes, but are identical in the formatting and function of the individual code pieces. The only calculation performed in the code is to convert the given erosion rates (originally in mm/kyr) to new units needed by the code (g/cm^2/kyr) using the density provided by the user. The Excel sheet for both nuclides uses one sample per row and requires the columns described in table 2.

Table 2: Excel to MATLAB formatting for ^{3}He and ^{14}C

Column	Parameter	Units	Description
1	Latitude	Decimal degrees	Latitude of the sample location.
2	Longitude	Decimal degrees	Longitude of the sample location.
3	Elevation	masl	Elevation of the sample. The sample elevation is not used directly in the code, but is entered for reference.
4	Pressure	hPa	Sample pressure should be calculated from the elevation using the ERA40 conversion (see section on ERA40atm.m). ERA40 is recommended for use because it was used in the calibration of production rates. For advanced users, the sample pressure can be independently calculated and entered here.
5	Sample thickness	cm	Average sample thickness should be measured for each sample.
6	Bulk density	g/cm^3	The bulk density of the sample. This should be measured for each sample or estimated for each batch of samples.
7	Shielding	unitless	Topographic shielding of the sample. Shielding can be calculated using the topooriginal.m file included in CRONUScalc.
8	Erosion rate	mm/kyr	Erosion rate for the sample surface.
9	Conc.	atoms nuclide g^{-1}	Measured concentration of nuclide after all appropriate blank subtractions
10	Inheritance	atoms nucide g^{-1}	This is a value for inherited nuclide. This is not used except in unusual cases.

Continued on next page

Table 2 – continued from previous page

Column	Parameter	Units	Description
11	Lambda effective	g/cm^2	Effective attenuation length for the sample should include effects of sample dip and topographic shielding. This is also an output of the topooriginal.m file.
12	Depth to top of sample	g/cm^2	This measurement is the depth to the top of the sample. For surface samples, this should be zero. Note the units are not cm, but g/cm^2.
13	Year Collected	Year AD	The year the sample was collected. This is used to properly integrate the production through time.
14-26	Uncerts		Uncertainties for each of the parameters listed above. Important parameters for uncertainties include the concentration and target element concentrations, especially chlorine.
27	Independent age	Years before 2010	Independent age given in years before 2010. This field is required if you are performing a calibration.
28	Independent age uncert	Years	Uncertainty on the independent age in years. This field is required if you entered an independent age for the sample.

Chlorine-36 For chlorine-36, there are many input parameters and more calculations performed in the Excel-to-MATLAB conversion than for the other nuclides. The properly formatted Excel sheet has the columns (one row per sample) described in Table 3.

Table 3: Excel to MATLAB formatting for ^{36}Cl

Column	Parameter	Units	Description
1	Latitude	Decimal degrees	Latitude of the sample location.
2	Longitude	Decimal degrees	Longitude of the sample location.
3	Elevation	masl	Elevation of the sample. The sample elevation is not used directly in the code, but is entered for reference.
4	Pressure	hPa	Sample pressure should be calculated from the elevation using the ERA40 conversion (see section on ERA40atm.m). ERA40 is recommended for use because it was used in the calibration of production rates. For advanced users, the sample pressure can be independently calculated and entered here.
5	Sample thickness	cm	Average sample thickness should be measured for each sample.
6	Bulk density	g/cm^3	The bulk density of the sample. This should be measured for each sample or estimated for each batch of samples.
7	Shielding	unitless	Topographic shielding of the sample. Shielding can be calculated using the topooriginal.m file included in CRONUScalc.
8	Erosion rate	mm/kyr	Erosion rate for the sample surface.
9	Conc.	atoms ^{36}Cl g^{-1}	Measured concentration of chlorine-36 after all appropriate blank subtractions
10	Inheritance	atoms ^{36}Cl g^{-1}	This is a value for inherited nuclide. This is not used except in unusual cases.

Continued on next page

Table 3 – continued from previous page

Column	Parameter	Units	Description
11	Lambda effective	g/cm^2	Effective attenuation length for the sample should include effects of sample dip and topographic shielding. This is also an output of the topooriginal.m file.
12	Depth to top of sample	g/cm^2	This measurement is the depth to the top of the sample. For surface samples, this should be zero. Note the units are not cm, but g/cm^2.
13	Year Collected	Year AD	The year the sample was collected. This is used to properly integrate the production through time.
14	Water content	volume %	The volume percent water is the percentage of the sample by volume that is water. This is typically the pore water and is estimated based on the pore volume of the rock and the climate.
15	SiO_2	weight %	Weight percent based on whole-rock composition, typically determined by XRF.
16	TiO_2	weight %	Weight percent based on whole-rock composition, typically determined by XRF.
17	Al_2O_3	weight %	Weight percent based on whole-rock composition, typically determined by XRF.
18	Fe_2O_3	weight %	Weight percent based on whole-rock composition, typically determined by XRF.
19	MnO	weight %	Weight percent based on whole-rock composition, typically determined by XRF.
20	MgO	weight %	Weight percent based on whole-rock composition, typically determined by XRF.

Continued on next page

Table 3 – continued from previous page

Column	Parameter	Units	Description
21	CaO	weight %	Weight percent based on whole-rock composition, typically determined by XRF.
22	Na_2O	weight %	Weight percent based on whole-rock composition, typically determined by XRF.
23	K_2O	weight %	Weight percent based on whole-rock composition, typically determined by XRF.
24	P_2O_5	weight %	Weight percent based on whole-rock composition, typically determined by XRF.
25	Analytical H_2O	weight %	Weight percent based on whole-rock composition.
26	CO_2	weight %	Weight percent based on whole-rock composition, typically determined by XRF.
27	Cl	ppm	Whole-rock chlorine concentration.
28	B	ppm	Bulk rock boron concentration.
29	Sm	ppm	Bulk rock samariam concentration.
30	Gd	ppm	Bulk rock gadolinium concentration.
31	U	ppm	Bulk rock uranium concentration.
32	Th	ppm	Bulk rock thorium concentration.
33	Cr	ppm	Bulk rock chromium concentration. Typically determined by XRF or ICP.
34	Li	ppm	Bulk rock lithium concentration. Typically determined by XRF or ICP.
35	Target K_2O	weight %	Target (mineral separate) concentration, typically determined by XRF or ICP.
36	Target CaO	weight %	Target (mineral separate) concentration, typically determined by XRF or ICP.
37	Target TiO_2	weight %	Target (mineral separate) concentration, typically determined by XRF or ICP.

Continued on next page

Table 3 – continued from previous page

Column	Parameter	Units	Description
38	Target Fe_2O_3	weight %	Target (mineral separate) concentration, typically determined by XRF or ICP.
39	Target Cl	ppm	Target (mineral separate) concentration, typically calculated from AMS results.
40-78	Uncertainties		Uncertainties for each of the parameters listed above. Important parameters for uncertainties include the concentration and target element concentrations, especially chlorine.
79	Covariance	Unitless	This is the covariance between the uncertainties on chlorine-36 concentration and total chlorine concentration. This is an output of the `spiketoconc.m` file.
80	Independent age	Years before 2010	Independent age given in years before 2010. This field is required if you are performing a calibration.
81	Independent age uncert	Years	Uncertainty on the independent age in years. This field is required if you entered an independent age for the sample.

The Excel table is pasted into MATLAB as a variable. The MATLAB code copies most of the inputs directly into the correct location. The code also does several conversions. The analytical water content is converted from weight percent to volumetric water content. This is added to the estimated volumetric water content and used in the MATLAB code as a single input for water. The uncertainties for these two parameters are also mathematically combined. The major element sample composition is normalized to include the volumetric water content as part of the composition (only applies to the samples measured by weight% ; not applicable to samples measured in ppm). Finally, the erosion rate (originally provided in mm/kyr) is converted to g/cm^2/kyr using the density of the sample.

The sample output from these codes produces the inputs described for the main MATLAB code in section 1.

2.3.4 Excel Templates

There are different template spreadsheets for each of the nuclides. Each file has several tabs corresponding to different types of templates. All the nuclides have at least two tabs: one for the online interface and one for the code itself. Other templates may also be included here. For example, `spiketoconc.m` also has a template included in the chlorine-36 template.

2.3.5 testsamplesXX.mat

These files are taken from the CRONUS-Earth calibrations to serve as examples of formatted data sets. These are already in the format needed to run the `be10age.m` code, for example.

2.4 Muoncalib Folder

This folder contains files used to calibrate the Al, Be, Cl from Ca, and Cl from K muon parameters using the depth profiles discussed in the CRONUS publications. In the CRONUScalc calibration, the chlorine-36 profiles are used to fit several necessary parameters simultaneously. These parameters are erosion rate, attenuation length for spallation reactions, and the muon parameters, σ_0 and f*. The files used to perform these calculations are `calClCaprofile.m` and `calClKprofile.m`. To plot the entire profile, use the `plotcoreCa.m` and `plotcoreK.m` codes. Many of the files in this folder have corresponding functions in the 'calib' folder and more complete descriptions can be found there.

2.5 Production Folder

This folder contains the heart of the code, including the pieces necessary to predict the nuclide concentration and perform scaling.

2.5.1 `aaXXuncert.m`

This code is called prior to performing a calibration in order to apply the uncertainties appropriate for the nuclide based on interlaboratory comparisons performed by the CRONUS Project. In most cases, the uncertainty changes depending on the concentration, with higher concentrations having

a lower percentage of uncertainty. The interlaboratory intercomparisons associated with CRONUS include: Jull et al. (2011, 2013); Vermeesch et al. (2012); Schaefer et al. (2014); Blard et al. (2014).

Syntax: `uncert=aaXXuncert(c)`

Input: c - concentration (atoms/gram) **Output:** uncert - 1-sigma uncertainty (atoms/gram)

Details: Given a nuclide concentration in atoms/gram, the function computes the uncertainty based on interpolation of the standard deviations of the interlaboratory comparison exercises or uses an average percentage for all concentrations. If a nuclide concentration is higher than the value from the interlaboratory comparison, then the uncertainty remains unchanged.

2.5.2 `andist.m`

This code is discussed in detail in Balco et al. (2008).

2.5.3 attenuationlength.m

Syntax: `l=attenuationlength(lat,lon,elev,pressure)`

This code uses the Sato model (Sato et al., 2008) to compute an approximate attenuation length for spallation reactions. This attenuation length is typically within about 10% of attenuation lengths that have been obtained by fitting to depth profiles. It should generally be acceptable for surface samples, where a small error in the attenuation length will not be significant. The attenuation length produced by this code may need to be further corrected for topography. The attenuation length is computed based on the long term average rigidity cutoff for the site. This will not necessarily be appropriate for relatively young sites. To calculate attenuation lengths without age information (long-term average rigidity cutoff), use `usingattenuationlengthnormal.m`. In order to account for the shorter exposure time (actual rigidity cutoff), use `usingattenuationlength.m`.

The code uses the `rawattenuationlength.m` file to determine the attenuation length. The code simply interpolates from a table using the elevation and latitude. The table is based on the Sato (PARMA) Excel sheet, but

using a correction to adjust the atmospheric attenuation lengths to those appropriate for rocks.

Input:

lat	Latitude (degrees)
lon	Longitude (degrees east)
elev	Elevation (m)
pressure	Atmospheric pressure (hPa)

Output: Fast neutron effective attenuation length (g/cm^2)

2.5.4 `compparsXX.m`

Syntax: `cp=comppars36(physpars,samppars,scalefactors,maxdepth)`

This code generates a complete set of parameters for dating from the physical parameters, sample-specific parameters, and scaling factors for a given sample.

Maxdepth, the maximum depth for which production parameters are calculated, can be adjusted so that it is appropriate for the type of reaction. The default for maxdepth is 25 g/cm^2, which is appropriate for surface samples.

Input:

physpars	Physics parameters (see physpars.m). This is constant for all nuclides.
samppars	Sample-specific parameters (see samppars.m)
scalefactors	Scaling factors for a given sample (see get_tdsf.m for details)
maxdepth	maximum depth for the calculation of parameters

Output: A structure named cp with the names specified below.

cp.Lambdafe	The effective attenuation length is simply copied from samppars so it is available in comppars.
cp.depthvector	The individual depths used to calculate all depth-dependent values.
cp.muonXX	Muon production rates. For Al/Be pair, this contains an extra row for Al production rates.
cp.SFmufast	Fast muon scaling factors, broken up by different energies
cp.SFmuonslow	Slow muon scaling factors, broken up by different energies
cp.negflux	Flux of negative muons
cp.totalflux	Total flux of muons.

For chlorine-36, there are additional outputs. These outputs are generally calculations for intermediate parameters necessary for production calculations. The variable names follow the naming convention of Gosse & Phillips (2001) very closely. For example, the variable cp.DeltaPhistareth is equivalent to $\Delta\Phi^*_{eth}$.

Table 4: Output parameters for comppars.m for ^{36}Cl

Parameter	**Description**
cp.ni	Concentrations for the major elements in units of at/g. This consists of all the bulk composition elements in the order they are taken in the input to the MATLAB code.
cp.tni	Concentration of the elements in the target (mineral separate) in units of at/g. These are also in the order they are taken in the input to the MATLAB code.
cp.fi	These are the bulk element concentrations converted from cp.ni into fraction of composition.
Continued on next page	

Table 4 – continued from previous page

Parameter	Description
cp.Ass	The average atomic weight of the subsurface material (See Gosse & Phillips (2001)).
cp.Sigmascss	Equation 3.22 in Gosse and Phillips (2001).
cp.Sigmathss	Equation 3.6 in Gosse and Phillips (2001).
cp.Ieff	Effective resonance integral for absorption of epithermal neutrons (See equation 3.9 in Gosse and Phillips (2001)).
cp.Xss	From Gosse and Phillips (2001).
cp. Sigmaethss	From Gosse and Phillips (2001).
cp.Lambdaetha	Attenuation length for epithermal neutrons in the air.
cp.Lambdaethss	Attenuation length for epithermal neutrons in the subsurface.
cp.Lambdathss	Attenuation length for thermal neutrons in the subsurface.
cp.PsCa	Production rate of calcium in this particular sample, so it accounts for the concentration of the target elements within a given sample.
cp.PsK	Production rate of potassium in this particular sample, so it accounts for the concentration of the target elements within a given sample.
cp.PsTi	Production rate of titanium in this particular sample, so it accounts for the concentration of the target elements within a given sample.
cp.PsFe	Production rate of iron in this particular sample, so it accounts for the concentration of the target elements within a given sample.
cp.B	Scattering rate parameter (see Gosse and Phillips (2001)).
cp.pEthss	See Gosse and Phillips (2001).
cp.Reth	Production rate of epithermal neutrons from fast neutrons in the air at the land/atmosphere boundary (see p. 1498 in Gosse and Phillips (2001)).

Continued on next page

Table 4 – continued from previous page

Parameter	Description
cp. Rth	Equation 3.2 in Gosse and Phillips (2001).
cp.Detha	Equation 3.16 in Gosse and Phillips (2001).
cp.Da	Equivalent to cp.Detha, see Equation 3.16 in Gosse and Phillips (2001).
cp.Dthss	Equation 3.33 in Gosse and Phillips (2001).
cp.Dethss	Equivalent to cp.Dthss, see Equation 3.33 in Gosse and Phillips (2001).
cp.Lambdatha	
cp.Letha	Epithermal diffusion length in the air, see Equation 3.21 in Gosse and Phillips (2001).
cp.Lethss	Epithermal diffusion length in the subsurface, see Equation 3.21 in Gosse and Phillips (2001).
cp.Lthss	Thermal neutron diffusion length, Equation 3.34 in Gosse and Phillips (2001).
cp.fth	Equation 3.32 in Gosse and Phillips (2001).
cp.feth	See Gosse and Phillips (2001).
cp.Phistaretha	Equation 3.26 in Gosse and Phillips (2001).
cp.Phistartha	Equation 3.38 in Gosse and Phillips (2001).
cp.Phistarethss	Equation 3.26 in Gosse and Phillips (2001).
cp.Phistarthss	Equation 3.38 in Gosse and Phillips (2001).
cp.DeltaPhistareth	Equation 3.29 in Gosse and Phillips (2001).
cp.DeltaPhistarth	Equation 3.41 in Gosse and Phillips (2001).
cp.DeltaPhistarstareth	Equation 3.30 in Gosse and Phillips (2001).
cp.FDeltaPhistarethss	Equation 3.28 in Gosse and Phillips (2001).
cp.FDeltaPhistareth	equivalent to cp.FDeltaPhistarethss.
cp.FDeltaPhistaretha	Equation 3.28 in Gosse and Phillips (2001).
cp.SFDeltaPhistaretha	Equation 3.39 in Gosse and Phillips (2001).
cp.SFDeltaPhistarethss	Equation 3.39 in Gosse and Phillips (2001).
cp.DeltaSFDeltaPhistareth	Equation 3.42 in Gosse and Phillips (2001).
cp.DeltaSFDeltaPhistaretha	Equivalent to negative cp.DeltaSFDeltaPhistareth.
cp.DeltaSFDeltaPhistarethss	Equivalent to cp.DeltaSFDeltaPhistareth.
cp.SFDeltaPhistartha	Equation 3.40 in Gosse and Phillips (2001).
cp.SFDeltaPhistarthss	Equation 3.40 in Gosse and Phillips (2001).

Continued on next page

Table 4 – continued from previous page

Parameter	Description
cp.FcompoundK	Compound factor for potassium.
cp.FcompoundCa	Compound factor for calcium.
cp.P_fasttotal	Combination of fast muon production from calcium and potassium.
cp.Prodmu	Combination of all muon production (fast and slow) for a specific sample.
cp.muon36	Second row contains muon production from Ca (fast and slow); third row contains muon production from K (fast and slow).
cp.negflux0	Surface negative muon flux.
cp.totalflux0	Surface total muon flux.
cp.Pmun0	Production rate of neutrons from muons at the surface.

Additional parameters are included in the code for production due to neutrons produced from radioactive decay of elements within the rock.

cp.Pnsf	See Fabryka-Martin (1988) for details.
cp.X	See Fabryka-Martin (1988) for details.
cp.Y	See Fabryka-Martin (1988) for details.
cp.Pnan	See Fabryka-Martin (1988) for details.
cp.Pethr	See Fabryka-Martin (1988) for details.
cp.Pthr	See Fabryka-Martin (1988) for details.
cp.N36r	Radiogenically produced chlorine-36 (see Fabryka-Martin (1988) for details).
cp.N36m	Measured chlorine-36 (taken directly from user input).
cp.N36c	Cosmogenically produced chlorine-36, calculated by subtracting the radiogenic component from the measured component.

2.5.5 `d2r.m`

This code is discussed in detail in Balco et al. (2008).

2.5.6 `desilets2006sp.m`

This code is discussed in detail in Balco et al. (2008). The original code was modified to use the geomagnetic history described in the main paper.

2.5.7 `dunai2001sp.m`

This code is discussed in detail in Balco et al. (2008). The original code was modified to use the geomagnetic history described in the main paper.

2.5.8 `ERA40ATM.m`

Syntax: `pressure = ERA40(site_lat,site_lon,site_elv)`

This file looks up mean sea level pressure and mean 1000 mb temperature from ERA-40 reanalysis and calculates site atmospheric pressures using these as inputs to the standard atmosphere equation. This file must load the data file ERA40.mat whenever called. Repeated calls to this function will be slow for this reason. To use this function for multiple samples, see `usingERA40atm.m`.

Input:

site_lat	Latitude (DD). Southern hemisphere is negative.
site_lon	Longitude (DD). The western hemisphere is negative, but it can also deal with 0-360 longitudes.
site_elv	Elevation (m).

Output: Site pressure in hPa.

2.5.9 `get_tdsf.m`

Syntax: `out = get_tdsf(sample,consts)`

This code calculates the scaling factors for all the nuclides, including both time-dependent and time-independent scaling models. The seven scaling

models used in CRONUScalc are the time-independent model by Lal/Stone, the time-dependent models by Dunai, Lifton, Desilets, time-dependent Lal/Stone, the Lifton/Sato/Dunai flux model, and the Lifton/Sato/Dunai nuclide-dependent model.

Input: The input argument 'sample' is a data structure. The 'sample' structure must have the following fields:

lat	Latitude, decimal degrees, S latitude is negative
long	Longitude, decimal degrees; code deals with both +180/-180 and 0/360 possibilities
pressure	Atmospheric pressure in mb (equivalent to hPa)
elevation	Elevation in meters

The constants are contained in the pmag_consts.mat file. The consts structure must have the following fields:

- consts.t_M
- consts.M
- consts.lon_Rc
- consts.lat_Rc
- consts.t_Rc
- consts.TTRc
- consts.IHRc
- consts.lat_pp_KCL
- consts.lon_pp_KCL
- consts.MM0_KCL
- consts.SInf
- consts.S.

Output:

out.tv	Vector of time values in years
out.Rc_De, out.Rc_Du, out.Rc_Li, out.Rc_Lm, out.Rc_Sa	Cutoff rigidities at the site (in GV) at the times in out.tv, corresponding to the scaling schemes.
out.SF_De, out.SF_Du, out. SF_Li, out. SF_Lm, out.SF_Sa, out.SF_Sa	Scaling factors for spallation at the site, at the times in out.tv, corresponding to the scaling schemes
out.SF_St	Scalar value – scaling factor for spallation at the site for the St scaling scheme.

Only the Lifton/Sato nuclide dependent scaling model differentiates between nuclides. However, the code requires a scaling factor for each one of the reactions. In all cases where there is no differentiation between reactions, all the reactions for a specific scaling model use the same scaling value.

2.5.10 `getcurrentsf.m`

`out=getcurrentsf(sf, t, scaling_model, nuclide)`

This code finds the scaling factor for the current time step given all the scaling factors (from `get_tdsf.m`) and the time.

Input:

sf	These are the scaling factors as returned by `get_tdsf.m`.
t	Time. Time is given based on the year 2010 with ages in the past indicated as negative numbers and ages occurring since the year 2010 indicated as positive values. For example, the year 2012 is written as t=2; the year 1000 BCE is indicated as t=-3010.
scaling_model	This 2-letter indicator for the scaling model is used to select the scaling model desired. The default is 'sa' for the LSD nuclide-dependent scaling, but other options are sf (LSD flux-only), li (Lifton), du (Dunai), de (Desilets), lm (Lal/Stone time-dependent scaling), st (Lal/Stone time-independent scaling). See the main paper for descriptions and references for all the scaling methods.
nuclide	This is currently only used to specify 'all', but may be used in the future to separate results by nuclide.

Output: The current scaling factors for each reaction are provided in a structure with the following fields. In the case of scaling models without reaction-dependent modeling, all the scaling factors are equal to out.Sel10 (the scaling factor for ^{10}Be).

out.Sel10	^{10}Be scaling factor. This is the only calculated parameter for nuclide-independent scaling methods.
out.SelSF	The LSD flux-only scaling factor. This is the nuclide-independent scaling.
out.Sel26	^{26}Al scaling factor. This is only calculated independently for nuclide-dependent scaling and is equivalent to Sel10 for all other scalings.
out.Sel14	^{14}C scaling factor. This is only calculated independently for nuclide-dependent scaling and is equivalent to Sel10 for all other scalings.
out.Sel3	^{3}He scaling factor. This is only calculated independently for nuclide-dependent scaling and is equivalent to Sel10 for all other scalings.
out.Sel36Ca	^{36}Cl scaling factor for spallation of Ca. This is only calculated independently for nuclide-dependent scaling and is equivalent to Sel10 for all other scalings.
out.Sel36K	^{36}Cl scaling factor for spallation of K. This is only calculated independently for nuclide-dependent scaling and is equivalent to Sel10 for all other scalings.
out.Sel36Ti	^{36}Cl scaling factor for spallation of Ti. This is only calculated independently for nuclide-dependent scaling and is equivalent to Sel10 for all other scalings.
out.Sel36Fe	^{36}Cl scaling factor for spallation of Fe. This is only calculated independently for nuclide-dependent scaling and is equivalent to Sel10 for all other scalings.
out.Sel21	This is currently a placeholder and set equal to Sel10 for all scaling methods.
out.SFth	^{36}Cl scaling factor for thermal energy production. This is only calculated independently for nuclide-dependent scaling and is equivalent to Sel10 for all other scalings.
out.SFeth	^{36}Cl scaling factor for epithermal energy production. This is only calculated independently for nuclide-dependent scaling and is equivalent to Sel10 for all other scalings.

2.5.11 `getparsXX.m`

```
[pp,sp36,sf36,cp36]=getpars36(sampledata36,maxdepth)
```

This code uses the nominal sample input (named `nominalXX`) to create all the necessary preliminary information for a particular sample. This essentially takes the regular input and calls the combination of `sampparsXX.m`, `physpars.m`, `compparsXX.m`, and `scalefacsXX.m` instead of having to call all of those files independently and in the correct order.

Input:

sampledataXX	This is the nominal sample data (as discussed in the first section of this document) for each nuclide.
maxdepth	This is the maximum depth for which the code needs to calculate values for production. For surface samples, this can be small (25-100 g/cm^2) but it will need to be increased for old samples, high erosion rates, or deep samples.

Output: Structures from `physpars.m` (pp), `sampparsXX.m` (sp), `compparsXX.m` (cp), and `scalefacsXX.m` (sf). See individual descriptions for each code for details.

2.5.12 `herrorbar.m`

This is included under the GNU public license from a contribution available at `http://uk.mathworks.com/matlabcentral/fileexchange/3963-herrorbar`. This allows for plotting of horizontal error bars.

2.5.13 `interpolate.m`

Syntax: `yout=interpolate(xin,yin,x)`

Given an ordered set of (xin(i),yin(i)) pairs, with xin(1) < xin(2) < ... < xin(n) and an ordered set of x values x(1) < x(2) < x(3) < ... < x(m), the code uses linear interopolation to produce yout(1), yout(2), ..., yout(m). The code returns NaN for any input x(i) where x(i) < xin(1) or x(i)> xin(n).

Input:

xin	An ordered series of x values for input.
yin	An ordered series of y values corresponding to the x values.
x	The x value to be used to find the interpolated y value.

Output: yout - The y-value corresponding the given x value in input.

2.5.14 `lifton2006sp.m`

This code is discussed in detail in Balco et al. (2008). The original code was modified to use the geomagnetic history described in the main paper.

2.5.15 `LiftonSatoSX.m`

Syntax: `out = LiftonSatoSX(h,Rc,SPhi,w,consts)`

This code implements the new Lifton/Sato scaling model (Lifton et al., 2014). This code calculates scaling factors for both the Lifton/Sato nuclide-dependent scaling model (Sa) and the Lifton/Sato flux scaling model (non-nuclide dependent) (Sf).

Input:

h	atmospheric pressure (hPa)
Rc	cutoff rigidity (GV)
SPhi	solar modulation potential (Phi, see source paper)
w	fractional water content of ground (nondimensional)

Output: The scaling factor for the flux-only scaling model and the scaling factors for the nuclide-dependent scaling factors for each reaction. The output is a structure with the following fields:

out.sp	Flux-based scaling factor for all nuclides
out.He	Scaling factor for helium-3 reaction
out.Be	Scaling factor for beryllium-10 spallation reaction
out.C	Scaling factor for carbon-14 spallation reaction
out.Al	Scaling factor for aluminum-26 spallation reaction
out.ClCa	Scaling factor for chlorine-36 calcium spallation reaction
out.ClK	Scaling factor for chlorine-36 potassium spallation reaction
out.ClTi	Scaling factor for chlorine-36 titanium spallation reaction
out.ClFe	Scaling factor for chlorine-36 iron spallation reaction
out.eth	Scaling factor for chlorine-36 epithermal energy reactions
out.th	Scaling factor for chlorine-36 thermal energy reactions

Details:

The code uses three additional functions (NeutronsLowE.m, NeutronsXS.m, and ProtonsXS.m) to calculate the fluxes of protons and neutrons at the sample location as well as a reference location (sea level and high latitude). The total fluxes of protons and neutrons are added together to produce a total flux at each location. The total flux is then divided by the reference flux to yield a scaling factor. This scaling factor is the Lifton/Sato flux method.

For the nuclide-specific scaling model, the flux that is applicable to each reaction is calculated independently. The applicable flux is calculated inside the `NeutronsXS.m`, `NeutronsLowE.m`, and `ProtonsXS.m` files using the reaction cross-section for a particular reaction.

2.5.16 `muonfluxsato.m`

Syntax: `out = muonfluxsato(z,h,Rc,s,consts,dflag)`

Calculates the production rate of ^{26}Al, ^{10}Beand other nuclides by muons as a function of depth below the surface, atmospheric pressure, and cutoff rigidity. The code is called by `comppars.m` and calculates muon fluxes that are used to compute production rates.

Input:

z	Depth below the surface (g/cm^2)
h	Site atmospheric pressure (hPa)
Rc	Cutoff rigidity (GV)

'consts' is a structure containing nuclide-specific constants, as follows:

consts.Natoms	atom number density of the target atom (atoms/g)
consts.k_neg	summary cross-section for negative muon capture (atoms/muon)
consts.sigma190	190 GeV x-section for fast muon production (cm^2)
consts.SPhiInf	solar modulation parameter

There is also a flag that indicates whether or not additional information will be displayed. The flag should be 'yes' in order to receive the complete set of muon fluxes.

Output:

out.phi_vert_slhl	The vertical flux of muons at sea level and high latitude.
out.R_vert_slhl	The vertical flux of negative muons at sea level and high latitude.
out.phi_vert_site	The vertical flux of muons at the site.
out.R_vert_site	The vertical flux of negative muons at the site.
out.phi	The integrated total flux of muons at the site.
out.R	The integrated flux of negative muons at the site.
out.SFmuslow	The scaling factors for slow muons at the site.
out.SFmufast	The scaling factors for fast muons at the site.
out.LambdaMu	Attenuation length for muons.
out.H	Atmospheric depth in g/cm^2.
out.Rz	Scaling factors for muons at given stopping depths.

2.5.17 `NCEPatm_2.m`

Please use the default pressure code is ERA40atm.m. If this code is required for some reason, it is discussed in detail in Balco et al. (2008).

2.5.18 `physpars.m`

Syntax: `pp=physpars()`

This code creates all the constants, such as decay constants and production rates, required for the rest of the code. There are no input or output arguments at this time. The scaling model is hard-coded in the second line of the function to allow for future streamlining of the code. The scaling model indicator is used in a 'switch case' instance to produce the correct production rates for each scaling model (all are included in every instance of physpars).

The parameters included in physpars are decay constants for all nuclides, compositional parameters for chlorine-36, production rates for all nuclides, fast muon parameters, and slow muon parameters. Production rates included with the distributed code are those calculated by CRONUS-Earth (Borchers et al., 2015).

2.5.19 `predNXX.m`

Syntax: `N36=predN36(pp,sp,sf,cp,age,scaling_model)`

This code predicts the concentration of the nuclide given the appropriate sample parameters and a particular age for the sample.

Input:

pp	Output from `physpars.m`
sp	Output from `samppars.m`
cp	Output from comppars.m
age	Desired calculation age in kyr before 2010.
scaling_model	Two letter scaling scheme indicator ('SA','SF','ST','DE','DU','LI','LM')

Output:

NXX	Concentration of the nuclide in units of atoms/gram.
N26	A second output is only included for predN1026 because it predicts both Be and Al concentrations (in that order).

How it works: The `predNXX.m` code uses `predNXXdepth.m` in order to calculate a predicted nuclide concentration for a given sample and a given age. The code divides the sample thickness into ndepths number of depths. ndepths is hard-coded to 10 in the code, but can be modified by the user. The concentration of the sample is predicted at each of these depths for the given age using the `predNXXdepth.m` function. The results from all depths

are averaged yielding the total production for the sample.

2.5.20 `predNXXdepth.m`

Syntax: `N36=predN36depth(pp,sp,sf,cp,age,depths,scaling_model)`

Input:

pp	Output from `physpars.m`
sp	Output from `samppars.m`
sf	Output from `scalefacsXX.m`
cp	Output from `comppars.m`
age	Desired calculation age in kyr before 2010.
depths	Depths for calculation (g/cm^2).
scaling_model	Two letter scaling scheme indicator ('SA','SF','ST','DE','DU','LI','LM')

This code predicts the concentration of the nuclide at a particular depth(s) for a particular age. Sample inputs include the outputs from physpars, samppars, scalefacs, and comppars. Other inputs include the age in kyr and a vector of sample depths in g/cm^2. In the code, the vector of sample depths is provided by predNXX.m.

How it works: The code calculates each depth to the appropriate depth at the start time using the following formula:

$$depth_{t0} = depth + erosionrate * age * 1000 \quad (2)$$

Where 1000 is a conversion factor for kyr to years.

At this point, the code adds up the total inventory for the sample starting at negative time (going into the past an amount of time equal to the input age). The concentration in the sample is equal to 0 initially. The time is incremented by the time step. At each time step, the new scaling factors are calculated and used to calculate the inventory for that time step using the prodZXX function. This concentration is added to the previous inventory. Simultaneously, the decay for the previously created nuclide is also calculated and subtracted from the inventory. The time step (deltat) is a parameter that is hard-coded but can be changed by the user. The default time step is 100 years.

The calculation of decay is performed using the following formula:

$$f = (1.0 - exp(-pp.lambda36Cl * deltat * 1000))/pp.lambda36Cl \quad (3)$$

$$N36 = N36 * exp(-pp.lambda36Cl * deltat * 1000) + pz * f \quad (4)$$

2.5.21 prodZXX.m

Syntax: `[output]=prodz1026(z,pp,sf,cp)`

Input:

z	depth of sample in g/cm^2
pp	Output from physpars.m
sf	Output from scalefacs.m
cp	Output from comppars.m

Output: The output is the production rate in terms of atoms/g target/year. The possible outputs are ProdsXX (production from spallation), ProdmuXX (production from muons), and ProdtotalXX (total production combining production from all pathways). A the end of each output, the percent production from pathways are also included, if assigned. These are not used in the calculations, but can be useful for other purposes. The results are different for each nuclide and are given in a vecor of outputs in the order shown below.

Nuclide	**Output vector order**
Al & Be	ProdtotalBe, ProdtotalAl, ProdsBe, ProdmuBe, ProdsAl, ProdmuAl
H	ProdtotalHe, ProdsHe (no muon production for He)
C	ProdtotalC, ProdsC, ProdmuC
Cl	Prodtotal, Prods, ProdsCa, ProdsK, ProdsTi, ProdsFe, Prodth, Prodeth, Prodmu, ProdmuCa, ProdmuK, Phith, Phieth, Kpercent, Capercent, Clpercent

If a depth is called that is deeper than the original calculations of production in comppars, then the code will produce an error: 'Prodz called for depth greater than maxdepth'. In this case, adjust the maxdepth parameter where `comppars.m` is called and rerun the code.

How it works: The production code is different for each nuclide, but is coded identically for the same types of production pathways. For example, all spallation reactions have the same type of coding. The code for spallation reaction is shown below. However, the production reactions for muons are discussed in those specific sections and the non-muon production reactions for chlorine are directly coded from Gosse & Phillips (2001).

Spallation reactions are calculated using the following formula:

$$ProdsBe = sf.currentsf.Sel10 * sf.ST * cp.PsBe * exp(-z/cp.Lambdafe) \quad (5)$$

2.5.22 ProtonsX.m

Syntax: `[pflux] = ProtonsX(h,Rc,s,consts)`

Input:

h	Atmospheric pressure (hPa)
Rc	Cutoff rigidity of the site
s	Solar modulatioin parameter
consts	physpars.m results

Output: Proton flux

Calculates the total proton flux for the given input parameters. The code directly programs in the Sato et al. (2008) Neutron Spectrum Analytical Function Approximation (PARMA). It does not account for any reaction cross-sections.

2.5.23 ProtonsXS.m

Syntax: `[output]= ProtonsXS(h,Rc,s,consts)`

The code directly programs in the Sato et al. (2008) Neutron Spectrum Analytical Function Approximation (PARMA). Code calculates the proton flux for each reaction individually and accounts for reaction cross-sections.

Output: The output is the total proton flux (pflux) and the output for each reaction accounting for the cross-sections (PXXp). The output is a vector with the parameters in the following order: pflux, P3p, P10p, P14p, P26p, P36Cap, P36Kp, P36Tip, P36Fep.

2.5.24 Ptoelev.m

Syntax: `elev=Ptoelev(lat,long,P)` This code simply takes a latitude in decimal degrees (lat), longitude in decimal degrees (long), and pressure in hPa (P) and calculates the elevation for that sample based on ERA40.

2.5.25 r2d.m

Syntax: `degrees = r2d(radians)`

Converts radians to degrees.

2.5.26 rawattenuationlength.m

Syntax: `l=attenuationlength(pressure,rigiditycutoff)`

Interpolates an attenuation length for a given atmospheric depth (in g/cm^2) and rigidity cutoff. The attenuation length returned is the "flat horizon" effective attenuation length based on the information presented in Sato et al. (2008). Further correction may be required to account for topography.

2.5.27 Rv0.m

Syntax: `[out]= Rv0(z)`

This function returns the stopping rate of vertically traveling muons as a function of depth z at sea level and high latitude. This code is discussed in detail in Balco et al. (2008).

2.5.28 `sampparsXX.m`

Syntax: `sp=samppars36(sampledata)`

This code takes the nominalXX as input and names all the parameters to the correct variables required by the code.

Input: The only input to this function is the sample data, as defined in the "inputs" section.

Output: The output produced is a structure (referred to as 'sp'). The outputs are described individually here. I have labeled them as sp36 if they correspond only to chlorine-36. Otherwise, they are listed as sp because they apply to all the nuclides.

sp.concentration	Directly copied from user input; nuclide concentration (atoms/gram). For Al/Be, there are two concentrations (one Al, one Be).
sp.inheritance	Directly copied from user input; inheritance in terms of total atoms of nuclide (atoms). For Al/Be, there are two inheritances (one Al, one Be).
sp.epsilon	Directly copied from user input; erosion rate (mm/kyr)
sp36.qavg	Directly copied from user input; water content (volume percent)
sp.rb	Directly copied from user input; bulk density (g/cm^3)
sp36.oxidenormfactor	Used to normalize the composition of the sample. For chlorine-36, the sample composition is provided as a dry sample composition. In the code, however, the total sample composition is necessary. The volume percent water is provided in sp36.qavg and this is used to normalize the composition to 100% (water percent plus oxide percents = 100%).
sp36.originaloxideinput	Directly copied from user input; this variable saves the original information so it can be viewed by the user later.
sp.latitude	Directly copied from user input; latitude of the sample (decimal degrees)
sp.longitude	Directly copied from user input; longitude of the sample (decimal degrees)
sp.ST	Directly copied from user input; topographic shielding factor.
sp.Lambdafe	Directly copied from user input; effective attenuation length for fast neutrons.
sp36.ci	composition of the sample in terms of weight percent oxide in the order listed in the table in Fabyrka-Martin (1988).
sp36.tci	Directly copied from user input; Target element composition in this order: K, Ca, Ti, Fe, Cl. K, Ca, Ti, and Fe are all in weight percent oxide (not normalized) and Cl is in ppm.
sp.elevation	Directly copied from user input; elevation of the sample in meters above sea level.
sp.P	Directly copied from user input; Pressure at sample location in hPa.
sp.depthtotop	Directly copied from user input; Depth to the top of the sample (g/cm^2)
sp.tfinal	Final age of the sample in kyr before 2010. This is calculated using the user input sample collection date.

2.5.29 `scalefacs.m`

Syntax: `sf=scalefacs36(sp36,scaling_model)`

This code creates variables required to call `get_tdsf.m` and then calls that function.

Input: Only input required is the original sample input for the appropriate nuclide as discussed in the input section above and the two-letter indicator for scaling model.

How it works: The code copies in some of the variables from previous parts of the code so that they are accessible (topographic shielding, pressure, elevation). The code then loads the geomagnetic constants file, `pmag_consts.mat`. Finally the code prepares the variables so that they are properly named to call `get_tdsf.m`.

Output: sf.tdsf is the result and includes the scaling factors for all scaling models and all nuclides. See discussion of get_tdsf for details.

2.5.30 `spiketoconc.m and spiketoconcmc.m`

Syntax: `[conc, concunc, ccl, cclunc, covar]= spiketoconc(spikeconc, spikeconcunc, spikemass, spikemassunc, A0, A0unc, samplemass, samplemassunc, RS, RSunc, SS, SSunc)`

This code is specific to chlorine-36 and calculates the concentration of chlorine-36 and total chlorine in a sample processed with a ^{35}Cl-enriched carrier. The code also produces appropriate uncertainties and the covariance between Cl concentration and ^{36}Cl concentration. The uncertainty calculation is linearized (using derivatives) in the original `spiketoconc.m`, but this was not appropriate for low-Cl samples (lower than approximately 10ppm) and blanks, so a Monte Carlo version was produced for low-Cl samples (`spiketoconcmc.m`). The inputs/outputs for these two functions are identical. These functions were originally designed for AMS results from the Purdue PRIME Lab, but may be used with other AMS results if they can be converted to these particular input formats.

Input:

spikeconc	Spike concentration [mg ^{35}Cl/gram of solution]
spikeconcunc	Uncertainty in spike concentration [mg ^{35}Cl/gram of solution]
spikemass	Mass of spike added to sample [grams of solution]
spikemassunc	Uncertainty in mass of spike [grams of solution]
A0	Isotope ratio of the spike [atoms ^{35}Cl/total atoms Cl]
A0unc	Uncertainty in A0 [atoms ^{35}Cl/total atoms Cl]
samplemass	Mass of sample [grams]
samplemassunc	Uncertainty in mass of sample [grams]
RS	Ratio of ^{36}Cl to total chlorine in a spiked sample [$\times 10^{-15}$, unitless]; From AMS analysis. Note that this should be entered as 154 for a sample with a ratio of 1.54E-13.
RSunc	Uncertainty in RS ratio [$\times 10^{-15}$, unitless]; From AMS analysis.
SS	Ratio of ^{35}Cl to ^{37}Cl in a spiked sample [unitless]; From AMS analysis.
SSunc	Uncertainty in the SS parameter [unitless]; From AMS analysis.

Output:

conc	Concentration of ^{36}Cl in the unspiked sample [atoms ^{36}Cl/g]
concunc	Uncertainty in ^{36}Cl concentration [atoms ^{36}Cl/g]
ccl	Concentration of total chlorine in the sample [ppm]
cclunc	Uncertainty in the chlorine concentration [ppm]
cov	Covariance between the uncertainties of ^{36}Cl concentration and total chlorine concentration in the sample due to the spiking procedure and calculations.

How it works: The function calls `spiketorm.m` to compute the ratio of chlorine-36 to total chloride in the unspiked rock as well as the total chlorine concentration. The results are used to calculate the total chlorine-36

concentration. The code then uses derivatives to calculate each uncertainty parameter as well as the covariance (spiketoconc.m) or uses Monte Carlo with 10,000 samples to calculate the uncertainty and covariance (spiketoconcmc.m).

2.5.31 spiketorm.m

Syntax: `[Rm,ccl]=spiketorm(spikeconc, spikemass, A0, samplemass, RS, SS)`

The code takes the spiking inputs and calculates the unspiked ratio of chlorine-36/total chloride and the total chlorine concentration in the unspiked sample. This code does not calculate uncertainties on the output parameters. These nominal values are calculated according to the equations presented in Desilets et al. (2006).

Input:

spikeconc	Spike concentration [mg ^{35}Cl/gram of solution]
spikemass	Mass of spike added to sample [grams of solution]
A0	Isotope ratio of the spike [atoms ^{35}Cl/atoms Cl]
samplemass	Mass of sample [grams]
RS	Ratio of ^{36}Cl to total chlorine in a spiked sample [unitless]; From AMS analysis.
SS	Ratio of ^{35}Cl to ^{37}Cl in a spiked sample [unitless]; From AMS analysis.

Output: Rm - Ratio of ^{36}Cl to total chlorine in the unspiked sample [unitless] ccl - Concentration of total chlorine in the sample [ppm]

2.5.32 stdatm.m

Syntax: `out = stdatm(z)`

This function converts elevation to atmospheric pressure according to the "standard atmosphere" (cf. CRC Handbook of Chem and Phys). See Balco (2008) for more details.

2.5.33 stone2000.m

This code is discussed in detail in Balco et al. (2008).

2.5.34 stone2000Rcsp.m

This code is discussed in detail in Balco et al. (2008).

2.5.35 usingattenuationlengthnormal.m

Use this to calculate attenuation length for all samples of unknown age (most samples). The rigidity cutoff is the long-term value.

Syntax: `out=usingattenuationlengthnormal(atteninput)`

Input:

Atteninput is group of samples (one sample per row) that contain the columns for latitude, longitude, elevation, and pressure.

Output:

Output is out.attenlength (the attenuation length for the sample).

2.5.36 usingattenuationlength.m

Use this function to calculate attenuation length for samples of known age (calibration samples). The rigidity cutoff is averaged only over the exposure period for the sample.

Syntax: `out=usingattenuationlength(atteninput)`

Input:

Atteninput is group of samples (one sample per row) that contain the columns for latitude, longitude, elevation, pressure, and independent age (ka).

Output:

Output is a struct with the following fields:

out.time	These are the time steps in the geomagnetic history
out.RcEstAge	This is the rigidity cutoff (Rc) averaged over the exposure age of the sample
out.RcEst	This is the Rc averaged over the entire available history (for comparison, not used).
out.endage	This is the last time used by the code for the geomagnetic history.
out.Lambdanotopo	This is the attenuation length for each sample.

2.5.37 using...m

There are several files that begin with 'using' and these all serve the same purpose - to make it simple to perform the function for all the samples in a

particular data set. For example, `ERA40atm.m` can only calculate the pressure for a single sample at a time and `usingERA40atm.m` loops through all the samples in a dataset and outputs pressures for all the samples. All the 'using' codes function in a similar fashion (`usingspiketoconc.m`, `usingpredN1026.m`, etc.).

2.5.38 usingspiketoconc.m

Syntax: `out=usingspiketoconc(spiking)`

This code makes it easy to run the spiketoconc.m code for multiple samples simultaneously. The code sets up the variables and calls the spiketoconc.m code repeatedly.

Input:

spiking - table with all the appropriate information for spiketoconc is in the columns and each row is a sample. This function just produces outputs for all the different samples simultaneously

Output:

Output is a struct with the same pieces from the spiketoconc.m code (conc, concunc, ccl, cclunc, covar) as well as a new variable, total. Total has all the previous outputs in a single variable in the order they were given so that they can be easily copied into a spreadsheet. Each row is a different sample and each column is one of the variables above. For example, total has the chlorine-36 concentration for each sample in the first column of the variable.

2.6 profilecalc

The profilecalc folder has code that ages depth profiles. Use the XXscript.m file as a template. Change the data file to load and the inheritance, age, and erosion rate bounds the calculator will use. This will be covered in more detail as the depth profile calculator is brought online.

2.6.1 ageprofilecalcXX

Syntax: `[posterior_er,posterior_age,posterior_inher, MAP, mu_bayes, chi2grid, lhsurf, jposterior] = ageprofilecalc1026(comp, concsig, depths, erates, ages, inhers, prior, maxerosion)`

This code calculates the age profiles for samples in a depth profile. The code includes versions for Al/Be and Cl.

Input:

comp	profile composistion
concsig	vector of concentration uncertanties
depths	vector of depths
erates	vector of erosion rates
age	vector of ages
inhers	vector of inhers
prior	prior distribution
maxerosion	maximum erosion allowed

Output:

posterior_er	posterior erosion rate distribution
posterior_age	posterior age distribution
posterior_inher	posterior inheritance distribution
MAP	maximum a posterior solution
mu_bayes	bayesian mean solution

How it works: The code defines the prior distributions, maximum depth for the calculation, and obtains the necessary sample parameters. Using the boundaries on ages, inheritance, and erosion rate, a three-dimensional grid is created. The joint posterior distribution is calculated for each point. The maximum probability point is found and reported as the maximum a priori solution (MAP). Finally, the code produces marginalized joint posteriors for each parameter. The marginalized joint posteriors allow for the visualization of the data in two dimensions.

2.6.2 bayesianCI

Syntax: `[BCI, P_x] = bayesianCI(x,p_x,alpha)`

The code calculates the upper and lower bounds of the Bayesian confidence interval.

2.6.3 bayesianCI2

Syntax: `[BCI mid] = bayesianCI2(x,y,p_x,alpha)`

Computed bayesian credible intervals for both x and y, at the (1-alpha) confiedence level. The optional output 'mid', is the height of the contour at which the (1-alpha) BCI should be drawn.

2.6.4 computechi2XX

Syntax: `chi2grid = computechi21026(erates, ages, inhers, comp, depths, measuredconc, sigma)`

The code computes a 3-dimensional array of Chi^2 values corresponding to the ages, erosion rates, inheritances, given for the profile defined by comp, depths, measuredconc, and sigma.

The code includes versions for Al/Be and Cl.

2.6.5 ex_1026

This code requires no inputs.

Profile calculator script for Greenland Be-10 #3 profile from Goehring (2010).

2.6.6 greenland_IC06-3.mat

Data for the Greenland Be-10 profile example (see ex_1026).

2.6.7 makeplotsXX

Syntax: `[erate_vs_age, age_vs_inher, erate_vs_inher] = makeplots-1026 (erates, ages, inhers, posterior_er, posterior_age, posterior_inher, nominal, uncerts, depths, map, jposterior,scaling_model)`

This code creates seven different plots including the posterior distributions for all three main parameters (erosion rate, age, and inheritance), best-fit profile, and the joint posterior surface for each combination of two parameters. This code has versions for both Al/Be and Cl.

Inputs:

erates	Erosion rate range for the profile.
ages	Age range for the profile.
inhers	Inheritance range for the profile.
posterior_er	The posterior for the erosion rates for the profile.
posterior_age	The posterior for the ages for the profile.
posterior_inher	The posterior for the inheritance for the profile.
nominal	Original sample inputs
uncerts	Original sample uncertainties
depths	
map	Best-fit solution (erosion rate, age, inheritance)
jposterior	
scaling_model	Two-letter indicator for the scaling model used for the profile.

Outputs:

- erate_vs_age
- age_vs_inher
- erate_vs_inher

2.6.8 plotprofXX

Syntax: `plotprof1026(comp,uncerts,depths,er,age,inher,scaling_model)`
This code plots the model against the original data.

2.6.9 plotstyle

Syntax: `plotstyle(hXLabel,hYLabel,hLegend)`
This code is used to set the style for plots.

2.6.10 prior.m

Syntax: `pr = prior (erates, ages, inhers, er_type, age_type, inher_type)`
The code calculates the 3-dimensional prior distribution.

2.6.11 profilecalcXX

Syntax: `[jposterior, posterior_er, posterior_age, posterior_inher, MAP, mu_bayes, BCI, chi2grid, lhsurf] = profilecalc1026 (filename, erates, ages, inhers, er_prior, age_prior, inher_prior, alpha, maxerosion)`
The code is a profile calculator for Be10/Al26.

2.6.12 PV06.mat

This file is the dataset needed for the PV06 depth profile.

2.6.13 PV06script.m

This code calculates the best-fit results for the depth profile PV06 (Panamint Valley). Contact Fred Phillips (phillips@nmt.edu) or Shasta Marrero (shastamarrero@gmail.com) for more details on this dataset. This script does not require any inputs.

2.6.14 wheremax.m

Syntax: `[indexofmax,themax]=wheremax(A)`

This code finds the index of maximum of a 3-dimensional matrix (A) and returns that maximum value. This is used in the profile calculator.

2.7 Surfacecalc

This folder contains necessary files to perform surface sample aging. Example sample files for each nuclide are also included in the Excelformatting folder.

2.7.1 YYXXage.m

Syntax: `[output,times,plotprodal,derivs]=al26age(sampledata,sampleuncertainties, scaling_model)`

This is the main aging routine that creates both age and one standard deviation uncertainty. If you are interested in looking at the derivatives of age with respect to each of the input parameters, you can call the code with four outputs. The outputs 'times' and 'plotprodXX' are used for plotting results in the interface and provide information about the production rate through time.

2.7.2 YYXXageraw.m

Syntax: `age=cl36ageraw(sampledata,pp,sf)`

This is a simplified version of the aaXXage.m code and it does all exposure age calculations except uncertainty calculations.

2.7.3 computeageXX

Syntax: `age=computeage10(pp,sp,sf,cp)`

This code uses the nuclide concentration to estimate the age of a sample. This is an estimate and not an exact calculation. This code is primarily used in order to estimate an age so that computation time might be reduced.

2.7.4 usingBe10age.m

This code is designed to facilitate the use of `be10age.m` (or similar files). The original code was designed to handle only one sample, but this version handles all samples in the input variable and records the resulting ages,

uncertainties, and other useful information in a variable called 'total' so that it can be easily copied to a spreadsheet.

2.7.5 be10erate.m

These functions perform erosion rate calculations (with uncertainty) for each nuclide. These have not been implemented for stable nuclides yet.

Syntax: `[erate,uncert,eratemm,uncertmm]=be10erate(nominal,uncerts,scaling_model)`

Input:

nominal	Regular input for the nuclide as discussed at the beginning of this document.
uncerts	Uncertainties for the nuclide (as discussed above).
scaling_model	Two-letter indicator for scaling model ('DE','DU','LI','LM','SA','SF','ST').

Output:

erate	Erosion rate in units of g/cm^2.
uncert	Erosion rate uncertainty in units of g/cm^2.
eratemm	Erosion rate in units of mm/kyr.
uncertmm	Erosion rate uncertainty in units of mm/kyr.

How it works: The function calculates the parameters needed using physpars (pp), samppars (sp), scalefacs (sf), and comppars (cp). The function calls be10erateraw.m, which calculates the erosion rate. The uncertainty is then computed using derivatives. The results are returned in two different units.

2.7.6 be10erateraw.m

This function performs the erosion rate calculation, but not uncertainty calculations.

Syntax: `erate=al26erateraw(pp,sp,sf,cp,scaling_model,minrate)`

The inputs are all created by `al26erate.m`. Minrate (optional input) can be used to set a lower bound on the erosion rate search.

How it works:

Using the maxage listed in this file (calculated using >6 half-lives), the erosion rate is calculated using a midpoint search to match the predicted concentration to the concentration measured in the sample. There is a very large maximum erosion rate set in the file to catch typographical and unit errors in entry. The only output is the erosion rate in units of g/cm^2.

2.8 Pf0calib Folder

This folder contains the files necessary to calibrate the Pf0 pathway for chlorine-36 based on the procedure outlined in Marrero (2012). Due to the limited utility of many of these files and the specific nature of the example, the files are only briefly discussed here. The files are also included for the calibration of the value of Pf0 from the Copper Canyon profile as discussed in Marrero (2012). The data files required for this exercise are also included. The Copper Canyon dataset is not ideal and the value obtained in the calibration is not in agreement with other values of Pf0 obtained from the CRONUS-Earth dataset. See the dissertation for details of the site and a discussion of the results.

In order to calibrate the Copper Canyon profile, first calibrate the 10Be profile using `calccbecore.m`. This produces an attenuation length and an erosion rate. Modify the appropriate file (`calccpf0.m`) to use these values and then run the Copper Canyon samples. The resulting value is the $P_f(0)$ parameter and the goodness of fit statistics (chi-squared, p-value). The values for $P_f(0)$ can be plotted using the `plotpf0.m` file (or its variants, depending on the type of plotting desired).

2.9 `calccbecore.m`

This code calibrates the erosion rate and attenuation length at the Copper Canyon site using the beryllium-10 dataset.

Inputs: There are no inputs to this code. The 10Be core data file is hard-coded into the file, but can be changed if necessary.

Outputs: Two parameters are calibrated, the attenuation length and the erosion rate. These are the only outputs.

The code finds the best-fit value for both the erosion rate and the attenuation length using the beryllium-10 data for Copper Canyon.

2.9.1 CCMM10.mat

This file has the beryllium-10 data for the Copper Canyon depth profile. This data is discussed in detail in Marrero (2012).

2.9.2 CCMM36QTZ.mat

This file has the chlorine-36 data for the Copper Canyon depth profile. This data is discussed in detail in Marrero (2012).

2.9.3 calccpf0qtz.m

This script uses the LM method to calibrate the erosion rate and Pf(0) from the Copper Canyon 36-Cl data.

There are no inputs to this code; The dataset is hard-coded. The outputs include many different parameters, but specifically sigmapstar (contains the calibrated production rate and erosion rate).

2.9.4 ccfun.m and corefun10.m

These codes calculate the derivatives. ccfun.m is used for the chlorine-36 profile and corefun10.m is used for beryllium-10.

2.9.5 ccjac.m and corejac10.m

These codes calculate the residuals for a set of samples. ccjac.m is used for the chlorine-36 profile and corejac10.m is used for beryllium-10.

2.10 `plotpf0.m`

This code plots the data against the values predicted by specified values of Pf(0). The dataset used in this code is `CCMM36QTZ.mat`. The code produces several plots using different symbols.

2.11 Interface Folder

The interface folder is a collection of MATLAB files that take batches of samples from the online calculator and processes them for use in CRONUScalc. The interface code then takes the results from the CRONUScalc program and formats it for communication to the user. The interface code is provided in the online repository as it is closely coupled to the calculator code, but is not described in detail here as it is related to the online interface and does not need to be used to run calculations directly. Files used for calculations are described below.

2.11.1 topofactor.m

Syntax: `out=topo(input,DD,dip_angle,options...)`

This code calculates the topographical shielding of a sample in a similar manner to that used in CHLOE (Phillips & Plummer, 1996). The newer version of this code, topo.m updated by Lifton, smooths the function and produces slightly more accurate results.

Input:

input	An n by 2 array of Theta & Horizon values. e.g. [1 1;2 2;3 3;1 0]. If no Theta and Horizon measurements (e.g., dipping surface only), just put one row of [0,0]. Theta measurements are compass bearings (0-360) and horizon values are the corresponding measurements of the horizon at those bearings (degrees).
DD	Direction of dip. Note that this is not strike.
dip_angle	Dip angle (degrees)
options	Optional values for constants Ssnow, Lf_t, and Lf_a. If given, they will override the default values. Example usage: topo(input,40,25,'Ssnow',1,'Lf_a',180)

Output: A struct with the following fields:

.ST_calc	Shielding factor incorporating all effects
.Slambda	Relative production change due to effective attenuation length
.Lf_calc	Effective fast neutron attenuation length
.Stopog	Calculated topographic/surface geometry shielding factor

References

Balco, G., Stone, J. O., Lifton, N., & Dunai, T. J. (2008). A complete and easily accessibly means of calculating surface exposure ages or erosion rates from ^{10}Be and ^{26}Al measurements. *Quaternary Geochronology*, *3*, 174–195.

Blard, P.-H., Balco, G., Burnard, P., Farley, K., Fenton, C., Friedrich, R., Jull, A., Niedermann, S., Pik, R., Schaefer, J. M., Scott, E., Shuster, D., Stuart, F., Tibari, B., Winckler, G., & Zimmermann, L. (2014). An inter-laboratory comparison of cosmogenic ^{3}He and ^{4}He in the CRONUS-P pyroxene standard. *Quaternary Geochronology*, *CRONUS-Earth Special Volume*, doi:10.1016/j.quageo.2014.08.004.

Borchers, B., Marrero, S. M., Balco, G., Caffee, M., Goehring, B., Gosse, J., Lifton, N., Nishiizumi, K., Phillips, F. M., Schaefer, J., & Stone, J. O. (2015). Geological calibration of spallation production rates in

the CRONUS-Earth project. *Quaternary Geochronology*, *CRONUS-Earth special volume*, doi:10.1016/j.quageo.2015.01.009.

Desilets, D., Zreda, M. G., Almasi, P. F., & Elmore, D. (2006). Determination of cosmogenic ^{36}Cl in rocks by isotope dilution: innovations, validation and error propagation. *Chemical Geology*, *233*, 185–195.

Gosse, J. C., & Phillips, F. M. (2001). Terrestrial in situ cosmogenic nuclides: theory and application. *Quaternary Science Reviews*, *20*, 1475–1560.

Jull, A., Scott, E., & Bierman, P. (2013). The CRONUS-Earth intercomparison for cosmogenic isotope analysis. *Quaternary Geochronology*, *CRONUS-Earth Special Volume*, doi:10.1016/j.quageo.2013.09.003.

Jull, A., Scott, E., & Marrero, S. (2011). The CRONUS-Earth intercomparison for cosmogenic isotope analysis. In *The Twelfth International Conference on Accelerator Mass Spectrometry* (p. 242).

Lifton, N., Sato, T., & Dunai, T. J. (2014). Scaling in situ cosmogenic nuclide production rates using analytical approximations to atmospheric cosmic-ray fluxes. *Earth and Planetary Science Letters*, *386*, 149–160.

Marrero, S. M. (2012). *Calibration of Cosmogenic Chlorine-36*. Doctorate New Mexico Tech, Dept of Earth & Environmental Science.

Nishiizumi, K., Imamura, M., Caffee, M. W., Southon, J. R., Finkel, R. C., & McAninch, J. (2007). Absolute calibration of 10Be AMS standards. *Nuclear Instruments and Methods in Physics Research Section B - Beam Interactions with Materials and Atoms*, *258*, 403–413.

Phillips, F. M., & Plummer, M. A. (1996). CHLOE: a program for interpreting in-situ cosmogenic nuclide data for surface exposure dating and erosion studies. *Radiocarbon (Abstr. 7th Int. Conf. Accelerator Mass Spectrometry)*, *38*, 98–99.

Sato, T., Yasuda, H., Niita, K., Endo, A., & Sihver, L. (2008). Development of PARMA: PHITS-based analytical radiation model in the atmosphere. *Radiation Research*, *170*, 244–259.

Schaefer, J. M., Winckler, G., Blard, P.-H., Balco, G., Shuster, D., Friedrich, R., & Schluechter, C. (2014). Performance of CRONUS-P: a pyroxene reference material for helium isotope analysis. *Quaternary Geochronology*, *CRONUS-Earth Special Volume*, doi:10.1016/j.quageo.2013.09.003.

Vermeesch, P., Balco, G., Blard, P.-H., Dunai, T. J., Kober, F., Niedermann, S., Shuster, D., Strasky, S., Stuart, F., Wieler, R., & Zimmerman, L. (2012). Interlaboratory comparison of cosmogenic 21Ne in quartz. *Quaternary Geochronology, http://dx.doi.org/10.1016/j.quageo.2012.11.009*, *CRONUS-Earth Special Volume*.