
RIG: a MATLAB™ program for evaluation of mechanical data from experimental apparatuses

Matěj Peč
Departement of Geology
University of Basel
Switzerland

Manual

(version 1.3, August 2010)

Installation:

Unpack RIG.zip into any convenient directory. Add the RIG directory including subdirectories (“rig_analogue_chart”, “rig_digital_chart” and “files”) to your MATLAB path (file > set path > add folder with subfolders > RIG). After this you either have to restart MATLAB or update your toolbox path cache (file > preferences > general > update toolbox cahce)

Uninstalation:

Delete the RIG directory and remove the path from your MATLAB

Contact:

Any bug reports, suggestions etc. are welcome on:

matej.pec@unibas.ch or matej.pec@seznam.cz

www:

<http://sites.google.com/site/matejpec84/>

Copyleft:

© 2009–2010, Matěj Peč, University of Basel, Switzerland

Programs “getpoly.m” and “txtwrite.m” were written by Ondrej Lexa for polyLX toolbox (<http://petrol.natur.cuni.cz/~ondro/polylx:home>)

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or any later version.

This program is distributed in the hope that it will be useful, but **WITHOUT ANY WARRANTY**. See the GNU General Public License for more details.

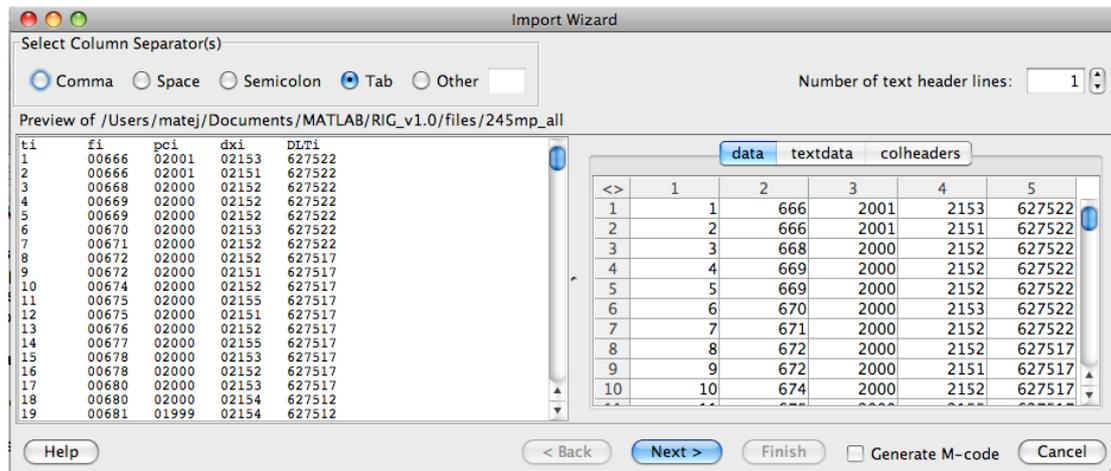
Acknowledgements:

Rüdiger Kilian, Ondrej Lexa, Renée Heilbronner and Petr Jeřábek provided many useful comments, which greatly improved this program. Anja Thust and Marianne Negrini are thanked for beta-testing and useful comments.

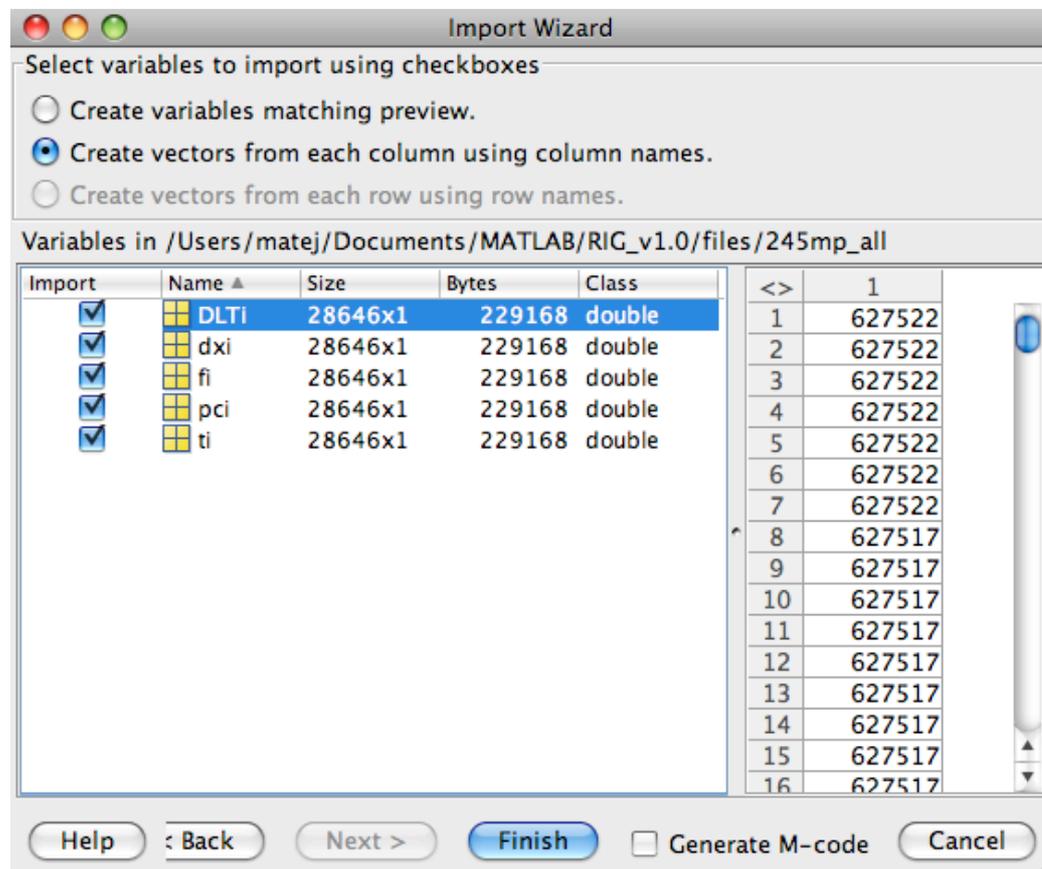
1. Quick guide	1
2. Detailed manual	3
2.1 Introduction	3
2.1.1 Motivation	3
2.2 rig_digital_chart	3
2.2.1 datacrop.m	5
2.2.2 rig.m	5
2.2.3 HP.m	5
2.2.4 conversions.m	8
2.2.5 conversionfactorsRIG1/2.m	8
2.2.6 rigC.m	8
2.2.7 rigS.m	11
2.2.8 rename.m	15
2.2.9 plots.m	16
3. Troubleshooting	16
4. References	17
5. Appendix	17
5.1 Variables used	17

1. Quick guide

Before you start, it is useful to close any open figures and to clear all variables on the workspace with the commands “*close all*” and “*clear all*” (write this into the command window). Then you have to import your data into MATLAB. The most convenient format to import is a text file with column headers corresponding to the variables in the column. Use the standard import function of MATLAB (file > import data...) a dialogue window like this will appear after you choose your file:



Select the appropriate separator and click “Next >”



select “Create vectors from each column using column names” and click “finish”. Now there are the desired variables on your workspace.

*Note: the variables **HAVE** to be called *ti* (time input), *fi* (force input) *pci* (confining pressure input) *dxi* (displacement input) and optionally *DLTi* (digital length transducer input – installed on rig2 in Tromsø). An example text file is included in the “files” folder for tests (245mp_all.txt).*

To run the program just type “rig” for evaluation of digital chart data or “rigA” for evaluation of “classical” chart data into the MATLAB command window and follow the instructions on the screen...that’s it!

Once the analysis is finished three text files will be created in the working directory (*name-of-your-experiment_log.txt* containing information about your input during the evaluation of data, calibrations used etc.; *name-of-your-experiment_fits.txt* containing the data about linear fits created during hit-point definition and three differently defined hit-point values [kN] (for details see fig. 4) and *name-of-your-experiment_coax.txt* or *_shear.txt* with the results for further processing using i.e spreadsheet editors.

If you want more help see section **3. Troubleshooting** or type “help rig” into the MATLAB command window or any other function name (see fig. 1 for names of used functions) – a simple help will be displayed.

2. Detailed manual

2.1 Introduction

We introduce an open platform MATLAB program for the evaluation of mechanical data from experiments. The program performs stress-strain, strain rate (and many more) calculations for samples deformed **a)** in **coaxial** compression **b)** in **shear** (induced by sliding of pre-cut blocks in coaxial compression). Further, it allows quantitative evaluation of the run-in and “elastic” curves. The program has been developed with special emphasis on the evaluation of data from solid medium Griggs deformation apparatuses but can be used for any other deformation apparatus working with the same sample geometry. Easily modifiable rig calibrations ensure that no deep knowledge of MATLAB is needed for the operation of this program. All calibrations are for the Tromsø rigs.

This program is based on the fortran programs rigPrep, rigC4 and rigS4 developed by Renée Heilbronner (<http://pages.unibas.ch/earth/micro/>) which were in turn developed from various precursors (RIGX etc.) written for the Griggs apparatus at Brown University and from another one (RIG) written for the Heard apparatus at ETH Zürich.

2.1.1 Motivation

During rock deformation experiments one usually records only the elapsed time, confining pressure ($\sigma_3=\sigma_2$), force (σ_1) and displacement of the σ_1 piston. These three input measurements, combined with the measured sample dimensions are used for calculating the final strain rate, stress-strain curves etc. It is clear that different numerical approaches can lead to significant differences in the final output data. Since usually experimental studies are not specific about the exact calculating procedure, we find it useful to introduce a program, which explicitly states how the raw data is evaluated. MATLAB was chosen because it is widely available on universities, allows easy plotting and the source code is easily readable.

The program is divided into two main sections: “rig_analogue_chart” and “rig_digital_chart”. The major difference is in the input data (chart units vs. volts) and conversion factors used. Otherwise is the calculating procedure identical. Rig_digital_chart includes some extra functions designed to deal with digital chart records and is described in detail here.

2.2 rig_digital_chart

The main function *rig.m* calls the following functions which will be described in the following sections: **a) *HP.m*** **b) *conversions.m*** **c) *conversionfactorsRIG1/2.m*** **d) *rigC.m*** **e) *rigS.m***. Additional scripts used are

f) *txtwrite.m* g) *getpoly.m* from the GNU polyLX toolbox (not described here).

The program is by default written for two rigs but can be easily expanded to any desired number. These functions have to be in the same folder, which must be in the MATLAB search path (see installation). Further scripts for more advanced MATLAB users are dealing with the raw and final data include h) *datacrop.m* i) *rename.m* j) *plots.m*. A flow chart summarizes the program structure in fig.1, a brief description of variables is in section 5.1.

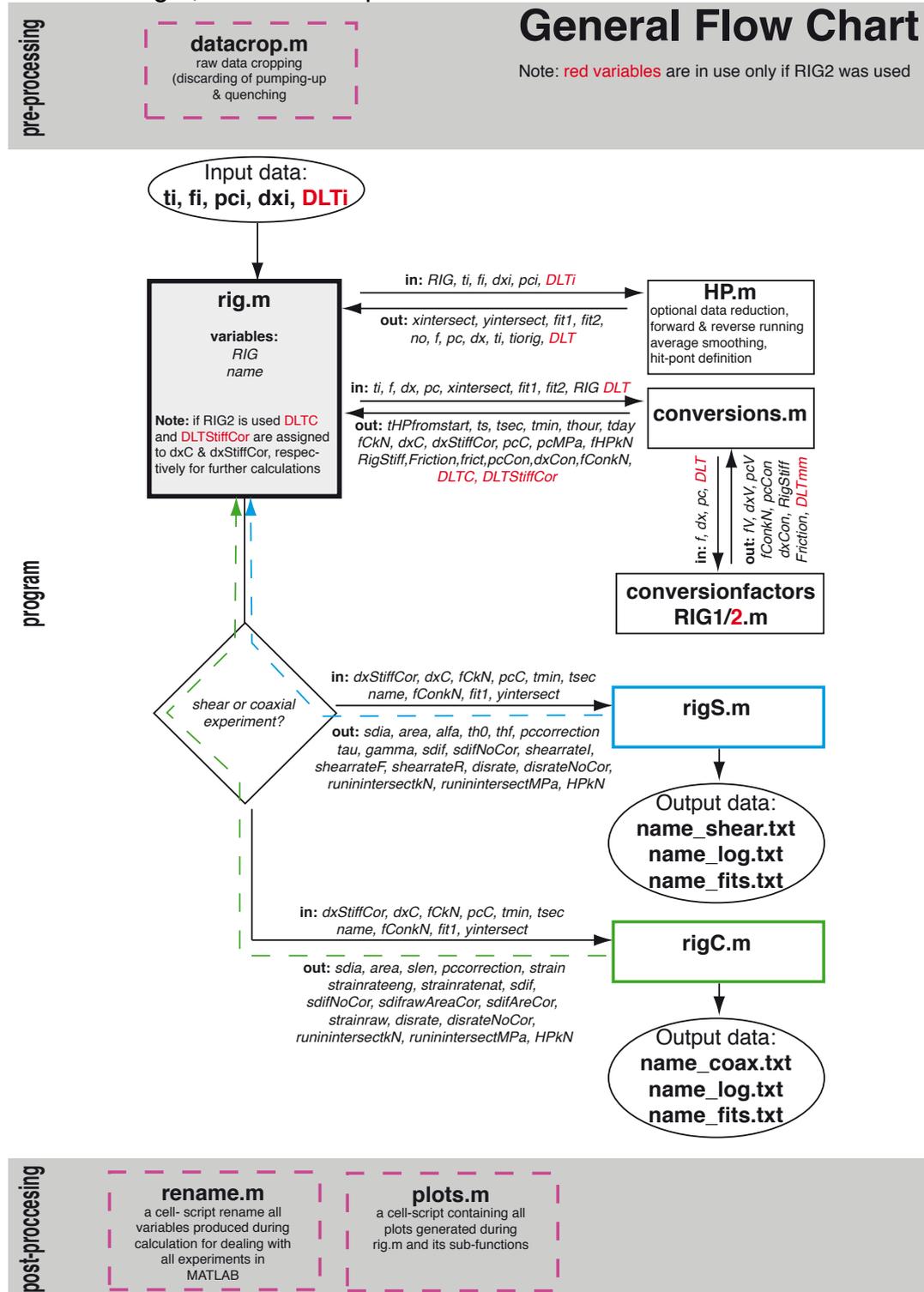


figure 1: RIG flow chart

2.2.1 **datacrop.m**

Script to crop data; used to cut away pumping up of the experiment and quenching. Plots f_i, p_{ci}, d_{xi} against t_i . Allows zooming to the area of interest and selecting of data via **getpoly.m**. Discards data outside of user-selected region. This script is not included in the **rig.m** and has to be called by typing “datacrop” into the MATLAB command window.

*Note: since the vectors have to have the same size for further evaluation it is necessary that the cut data start and stop on a common t_i , i.e. the selection has to be rectangular. Otherwise an error **??? Error using ==> plot Vectors must be the same lengths** will occur.*

2.2.2 **rig.m**

This script is used to call all the other functions, plot recalculated data (force [kN], displacement [mm] and confining pressure [MPa] vs. time [min]) and to write the output text files. Function **txtwrite.m** is used for writing the **_log.txt** and **_fits.txt** file. Standard c-language command **fprintf** is used to write the **_shear.txt** and **_coax.txt** files due to large amounts of data written into these files (**fprintf** is about one order of magnitude faster than **txtwrite.m**).

2.2.3 **HP.m**

This script is used to define the hit-point and basic data treatment. For large files (>100 000 records per variable) a data reduction option is offered to maintain computing speed. The data reduction takes every Xth point according to user input (i.e. entering “2” results in a file half the original size). Data smoothing by a forward and reverse running average is offered (**filtfilt.m**). This filter has the advantage that it doesn’t introduce any phase shift, effectively filters out digital input steps and high frequency noises (fig. 2). The minimum possible filter range is 3, too high filter range results in poor fits in areas with an abrupt change in slope (fig. 3)

Note: for smooth curves with a record interval of 1 sec anything between 100-200 usually works fine.

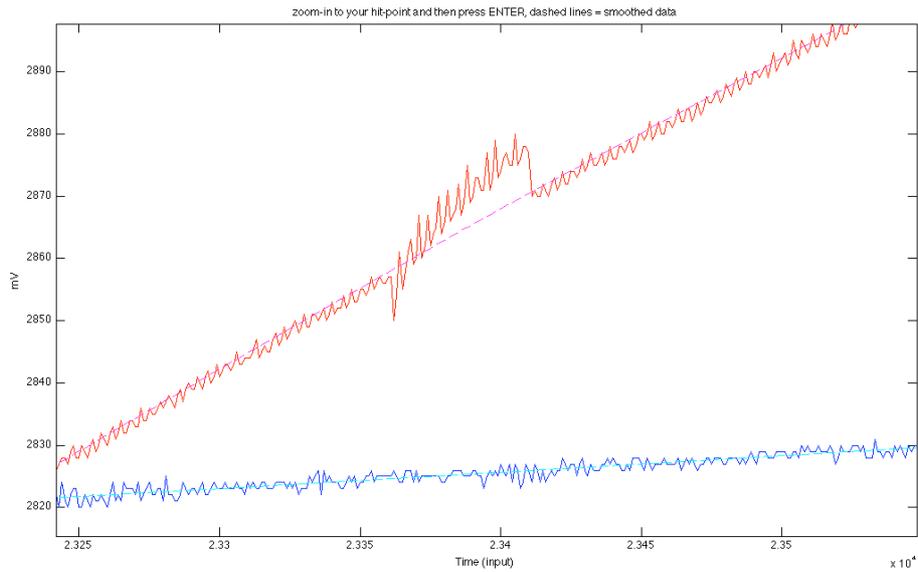


figure 2: raw data (solid lines), smoothed data (dashed lines)

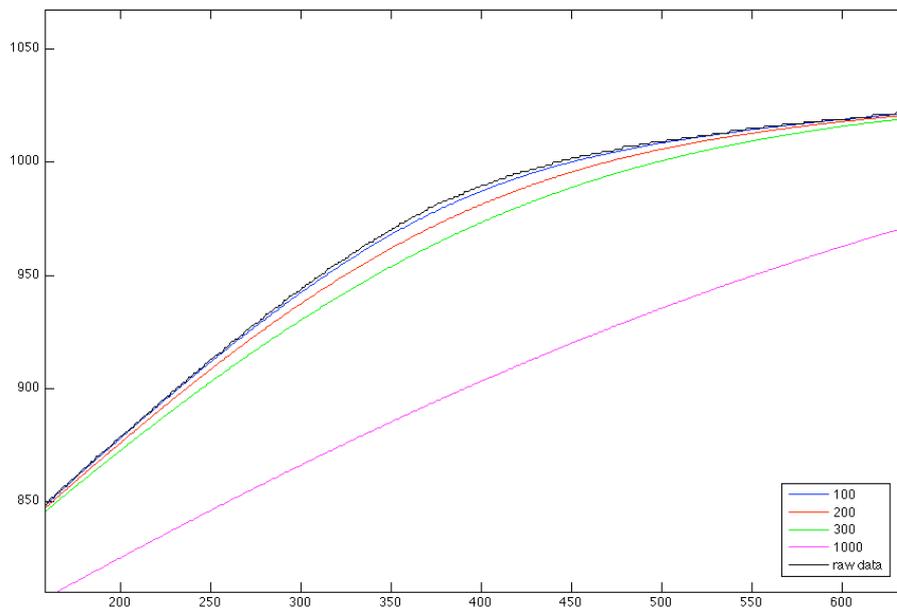


figure 3: effect of different filter range (purple = 1000, green = 300, red = 200, blue = 100, black are raw input data)

Afterwards a plot of the raw force [mV], displacement data [mV], confining pressure [mV] and its smoothed counterparts vs. time [record interval] is generated (fig. 4). Zooming into the individual curves allows verifying the quality of the data fit.

Note: by pressing "ctrl + c" you can abort the program if the smoothing is not good and start the program again to choose a more appropriate number of data points used for smoothing.

Once you are satisfied with the data fit you can zoom into the region of the hit-point so that the run-in as well as “elastic” part of the curve can be seen.

Note: Appropriate zooming (i.e. a long and narrow rectangle) also allows to vertically exaggerate the curves for better identification of the hit-point in weak samples.

After hitting “enter” the plot is updated and allows selecting of a region on the run-in curve and the “elastic” part of the force curve (using the **getpoly.m** script). These regions are fitted with a linear regression curve and their intersection is calculated (actual hit-point).

*Note: start with the **run-in curve** in order to get the right numbers in the `_fits.txt` text file – otherwise the slopes and y-intersects of the run-in and elastic curves will be switched.*

Output is the smoothed data, intersect of the two linear regression curves and the linear fit thru the run-in curve & “elastic” curve in the form $\text{fit}=[\mathbf{a},\mathbf{b}]$ where **a** is the slope of the run in curve and **b** its intersect with the y-axis ($y=\mathbf{a}x + \mathbf{b}$). In the end the script discards all data before the hit-point.

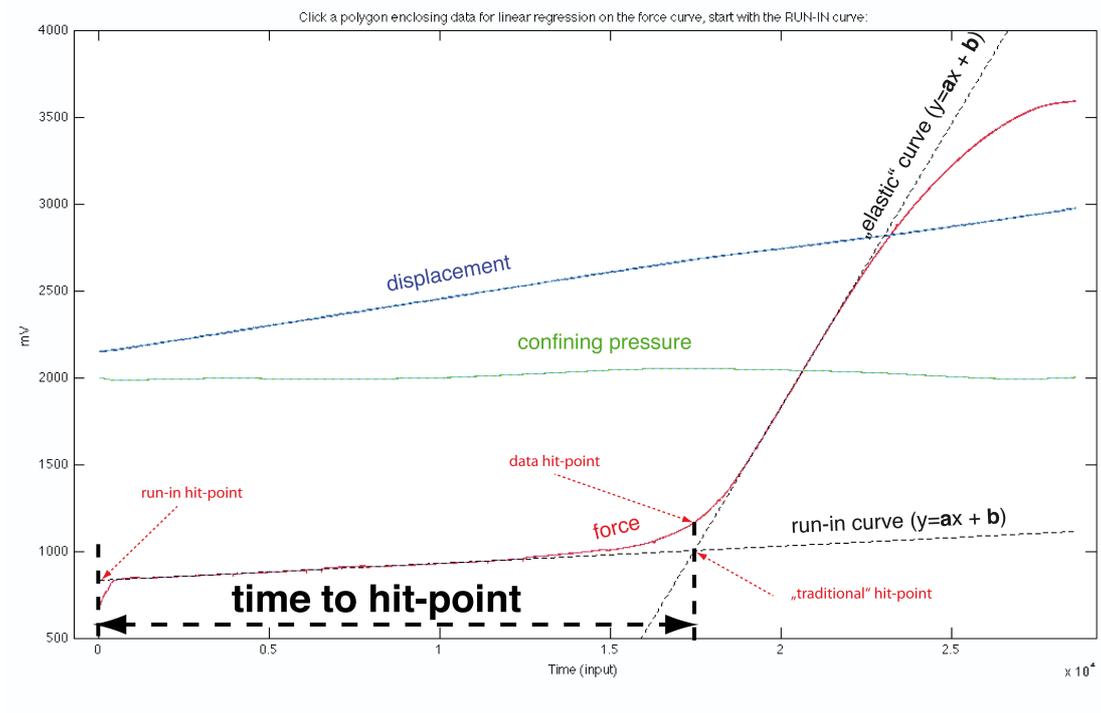


figure 4: raw and smoothed data, three different hit point definitions and definition of time to hit-point are shown (part of the *name-of-your-experiment_fits.txt* output)

2.2.4 conversions.m

This script recalculates the input time to time in seconds, minutes, hours and days and calculates the time at hit-point.

*Note: the time at hit point is calculated assuming that the analyzed file starts in the moment where the σ_1 motor has been started (fig. 4). (use **datacrop.m** to cut away the pumping-up of the experiment if it has been recorded into the same file as the actual deformation). The run-in hit-point is defined as the y-axis intersect of the run in curve, therefore user attention is needed to verify the quality of the fit (good for well behaving run-in curves, gets worse in weak samples and/or steep run-in curves)*

Further, it calls the function **conversionfactorsRIG1.m** (or **conversionfactorsRIG2.m**) where the appropriate conversion factors are stored and recalculates the raw data to [kN] for force, [MPa] for confining pressure and [mm] for displacement. Displacement corrected for rig stiffness and force [kN] corrected for friction is calculated as well (details of these corrections are described in 2.2.5).

It plots the uncorrected displacement [mm] vs. time [min] and the least squares fit forced thru 0 to check the linearity of motor speed (deviates from linear by strong samples)

2.2.5 conversionfactorsRIG1/2.m

A short function that stores the conversion factors ([V] to [kN],[MPa],[mm] for force, confining pressure and displacement, respectively) for individual rigs so that it can be easily modified.

It also contains the rig stiffness correction factor (*rigStiff*), i.e. how much “longer” does the rig get with applied load [mm/kN] and rig friction (*Friction*). The rig stiffness is typically obtained by loading some very stiff material (i.e. tungsten carbide rods) at conditions close to the experimental ones. One assumes that all the measured displacement is due to the compliance of the rig. The values for friction can be obtained from salt dummy runs that simulate the real experiment. In the case of a Griggs rig, one obtains a bulk correction factor including the confining media strength, jacket strength and any friction in the apparatus itself (Holyoke & Kronenberg, in press). Both of these corrections can be ignored (i.e set to 0).

2.2.6 rigC.m

This script evaluates the coaxial experiments. All corrections and assumptions are plotted so that the user can immediately see their effect. The script reads the measured sample length and diameter (fig. 5). Strain is

calculated by assuming that the sample retains a right cylinder shape during deformation by the following equation:

$$\varepsilon = \frac{1 - (l_0 - dx)}{l_0} \times 100 \quad [\%]$$

where dx is the displacement of σ_1 piston and l_0 is the original sample length.

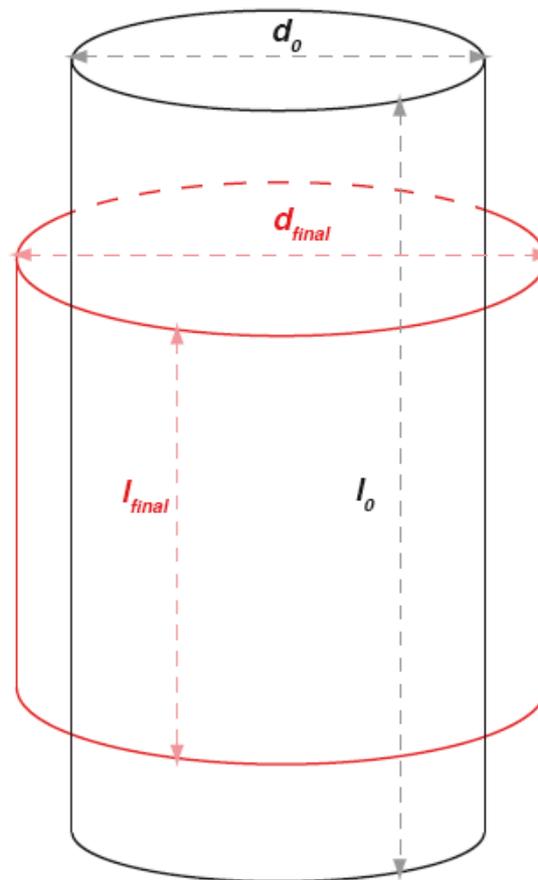


figure 5: sample in coaxial compression before (black) and after deformation (red) assuming no volume change

Displacement rate of the σ_1 piston is calculated from displacement (uncorrected as well as corrected for rig stiffness) as:

$$\dot{v} = \frac{\Delta dx}{\Delta t} \quad [mm * sec^{-1}]$$

Natural (finite) strain rate is then calculated as:

$$\dot{\varepsilon}_{nat.} = \frac{\dot{v}}{l_0} \quad [sec^{-1}]$$

and engineering (instantaneous) strain rate as:

$$\dot{\epsilon}_{eng.} = \frac{\dot{v}}{l_0 - (l_0 - dx)} \quad [sec^{-1}]$$

The applied stress on the sample is calculated assuming that the sample retains a constant volume (i.e. cross-sectional area expands at the same rate as the sample is being shortened) (fig. 5).

$$\sigma_1 = \frac{F \times \left(\frac{l_0 - dx}{l_0} \right)}{area} \quad [MPa]$$

Note: This correction can lead to significant changes in the trend of the data by high strain experiments (fig.6) and should be reviewed with caution by the user

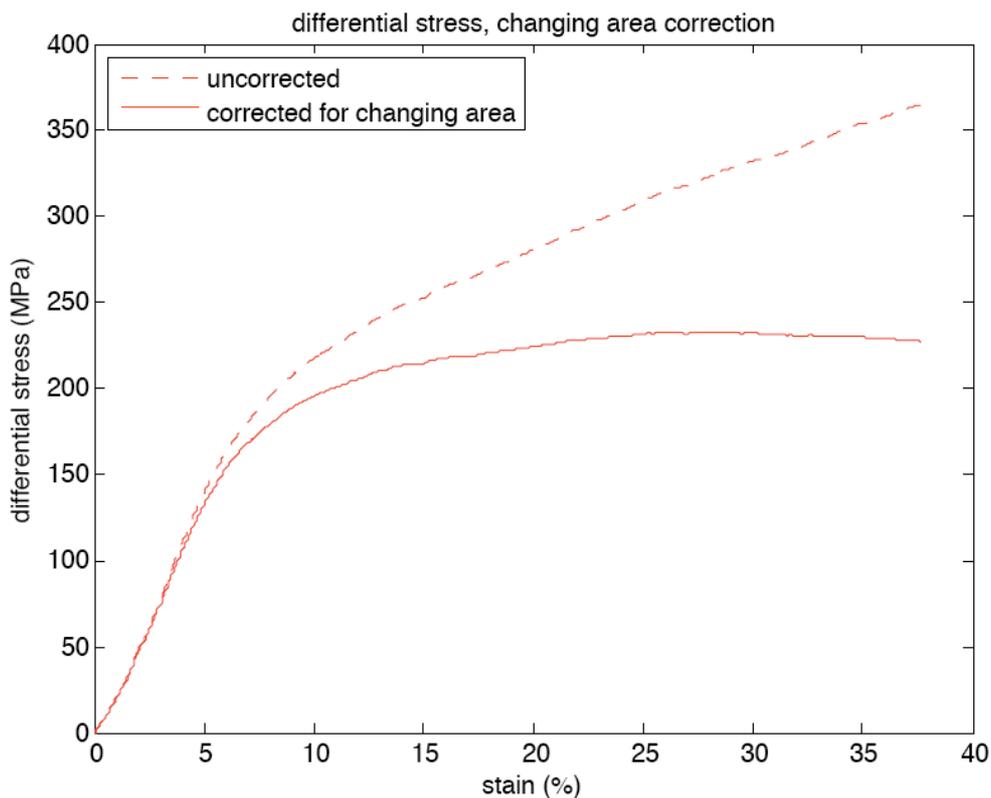


figure 6: strain hardening or steady state? The effect of constant volume assumption

Change in the confining pressure (σ_3) with time (x) occurs during Griggs rig experiments. Since the differential stress is calculated as $\sigma_1 - \sigma_3$ at the time of the hit-point one might prefer to calculate the differential stress taking in account the change of the confining pressure with time.

$$\Delta\sigma_{(x)} = \sigma_{1(x)} - \sigma_{3(x)} \quad [MPa]$$

The script plots raw stress (uncorrected for area change) and stress (corrected for area change) [MPa] vs. strain [%], displacement rate of the σ_1 piston (corrected as well as uncorrected for rig stiffness) [mm/sec] vs. time [min] and natural and engineering strain rates [sec⁻¹] vs. time [min]. If the user has chosen the confining pressure correction it also plots the differential stress (uncorrected and corrected for changing confining pressure) [MPa] against time [min].

2.2.7 rigS.m

This script evaluates shear experiments. All corrections and assumptions are plotted so that the user can immediately see their effect. The script reads the measured sample diameter; initial and final thickness of the shear zone [mm] and the angle of the pre-cut in the forcing blocks [°] (fig. 7 & 9). The initial thickness of the shear zone has to be measured i.e. on a hydrostatic experiment. We assume that no change in the area of the forcing blocks occurs (i.e d_0 stays constant during the deformation).

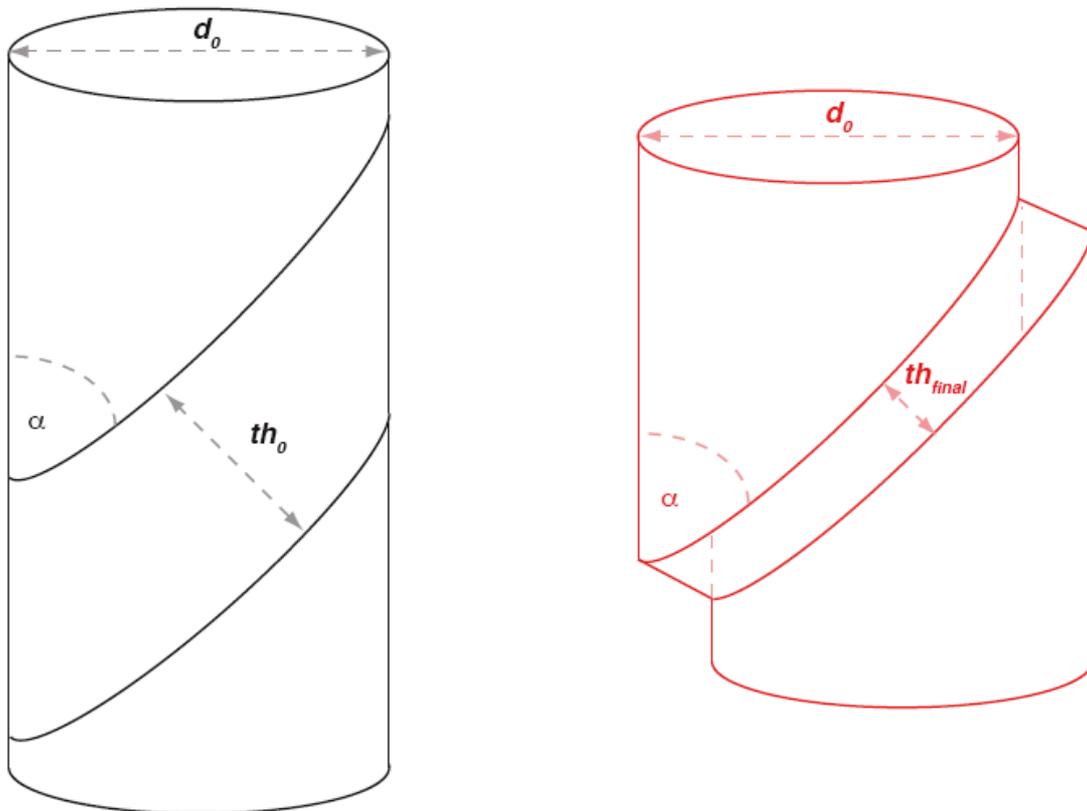


figure 7: Sample geometry for shear experiments before (black) and after (red) deformation.

Shear strain is calculated by resolving the vertical displacement (dx) into a pure-shear component (Δth) i.e. thinning of the shear zone and simple-shear component (dz_{simple}). We assume that the thinning of the shear zone occurs continuously during the whole run and that maximal piston overlap occurs in

the beginning of the experiment (fig. 9). First the proportion (i.e. how much vertical displacement does account for the compaction of the SZ) is calculated by the following equation:

$$proportion = \frac{dz_{1,tot}}{dx}$$

Then the vertical displacement component is calculated as:

$$\begin{aligned} dz_{1(x)} &= proportion \times dx_{(x)} && [mm] \\ dz_{2(x)} &= dx_{(x)} - dz_{1(x)} && [mm] \end{aligned}$$

The thickness of the shear zone at a given time (x) is calculated as (fig. 9):

$$th_{(x)} = th_0 - dz_{1(x)} \times \sin(\alpha) \quad [mm]$$

And the simple shear component is calculated as:

$$dz_{simple(x)} = \frac{dz_{2(x)}}{\cos(\alpha)} \quad [mm]$$

Shear strain (γ) at given time (x) is then calculated as:

$$\gamma_{(x)} = \frac{dz_{simple(x)}}{th_{(x)}}$$

The normal stress is calculated as (assuming no forcing block deformation):

$$\sigma = \frac{F}{area} \quad [MPa]$$

Where F is the force in $[N]$ and $area$ is in $[mm^2]$

Again a correction for changing confining pressure is offered:

$$\Delta\sigma_{(x)} = \sigma_{1(x)} - \sigma_{3(x)} \quad [MPa]$$

Assuming that the area of contact gets smaller with increasing slip, we calculate the changing area by the following equations:

$$contact_{(x)} = \frac{dz_{simple(x)}}{\frac{d_0}{\sin(\alpha)}}$$

Where $\frac{d_0}{\sin(\alpha)}$ is the length of the shear zone (fig. 9). The area overlap is calculated using auto correlation function of a circle. We use a cubic fit to one half of the auto correlation function profile (fig. 8). Note that the solution is identical also for an ellipse. For more information on ACF see i.e Heilbronner 2002.

$$acfArea = 1 - 1.2082 \times contact - 0.19134 \times contact^2 + 0.39461 \times contact^3$$

[mm²]

Note: this assumption has exactly the opposite effect as the constant volume assumption in coaxial experiments evaluation (i.e. force steady state turns into shear stress hardening because of the diminishing area on which the force acts).

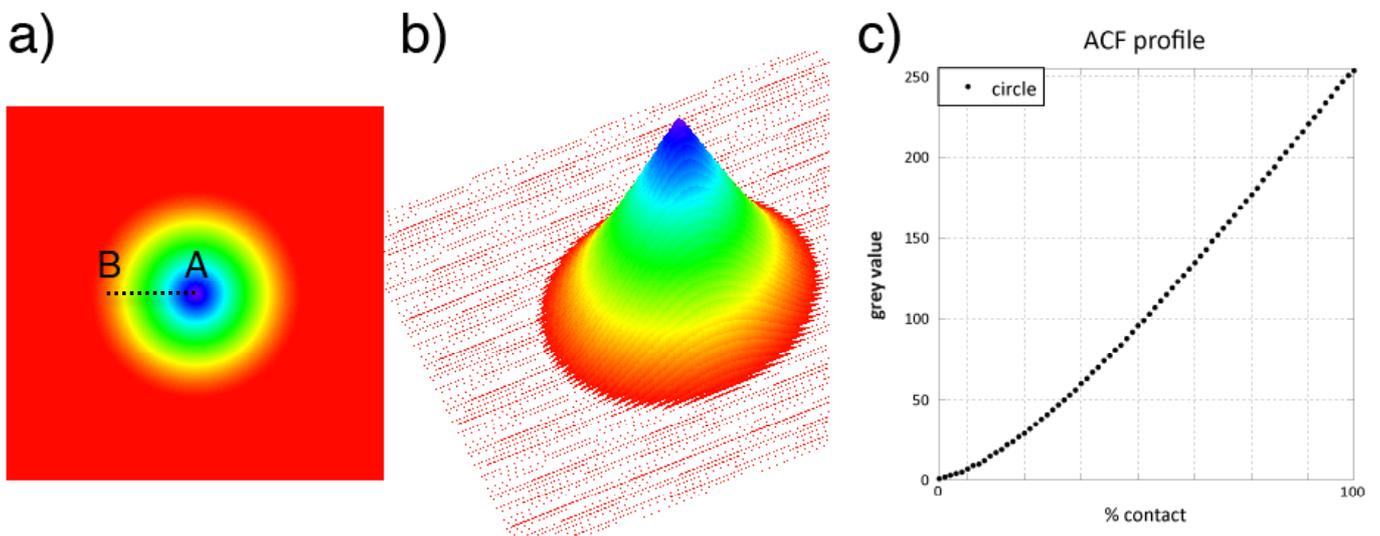


figure 8: **a)** auto correlation function of a circle, purple corresponds to 100% correlation (i.e. maximal piston overlap), red corresponds to no correlation (i.e. no piston overlap); **b)** surface plot of a); **c)** profile B-A of a), cubic fit of this curve is used to calculate *acfArea*

Shear stress (τ) is calculated for **45° pre-cut forcing blocks** as:

$$\tau = \frac{0.5 \times \Delta\sigma}{acfArea} \quad [MPa]$$

*Note: The calculated shear stress is **ONLY** valid for the case of 45° pre-cut forcing blocks – this will be fixed in some future release.*

Natural (finite) shear strain rate is then calculated as (Twiss & Moores 2007):

$$\dot{\gamma}_{nat} = \frac{0.5 \times \Delta dz_{simple}}{th_0 \Delta t} \quad [sec^{-1}]$$

Engineering (instantaneous) shear strain rate is calculated as (Twiss & Moores 2007):

$$\dot{\gamma}_{eng} = \frac{0.5 \times \Delta dz_{simple}}{th(x) \Delta t} \quad [sec^{-1}]$$

Shear strain rate compatible with the one calculated in rigS4.f (<http://pages.unibas.ch/earth/micro/>) is calculated as:

$$\dot{\gamma}_r = \frac{\Delta \gamma}{\Delta t} \quad [sec^{-1}]$$

Finally, the displacement rate of the σ_1 piston is calculated from dx (not corrected as well as corrected for rig stiffness) as:

$$\dot{v} = \frac{\Delta dx}{\Delta t} \quad [mm * sec^{-1}]$$

The program plots shear stress [MPa] vs. shear strain and all three strain rates [sec⁻¹] vs. t [min]. If the user has chosen the confining pressure correction it also plots the differential stress (uncorrected and corrected for changing confining pressure) [MPa] against time [min].

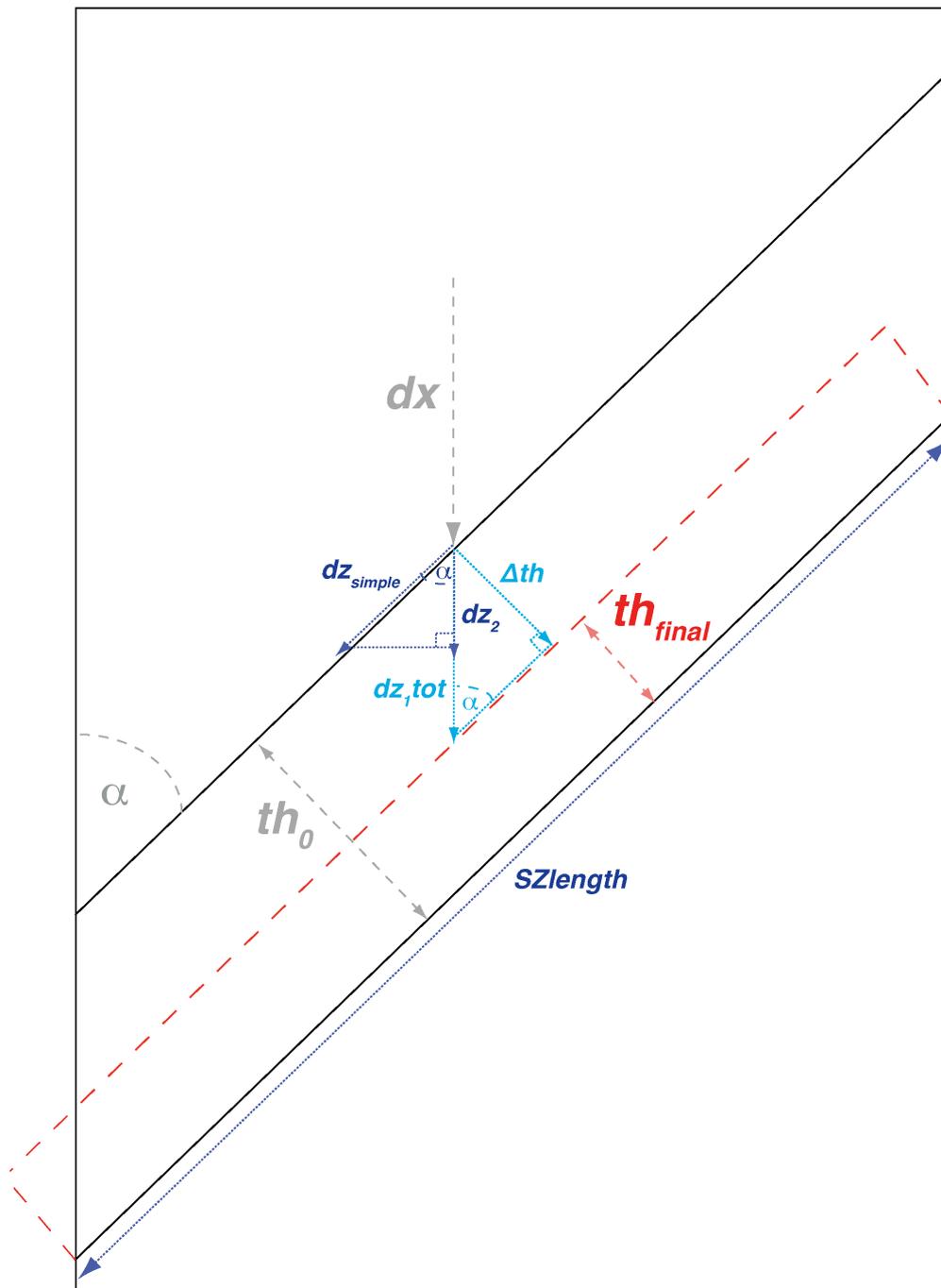


figure 9: Resolving of vertical displacement (dx) into pure shear (Δth) and simple shear (dz_{simple})

2.2.8 rename.m

A simple cell script that can be used to rename and clear all variables. It is separated into three cells, which can be executed by the “evaluate cell” button. The first one renames the DLT variables (use only if RIG2 was used), the second one is used to rename all variables used during the calculation of a shear experiment and the last one is used for renaming variables used during the calculation of coaxial experiments. Afterwards, you can save your workspace for further data treatment with MATLAB. To access this script just

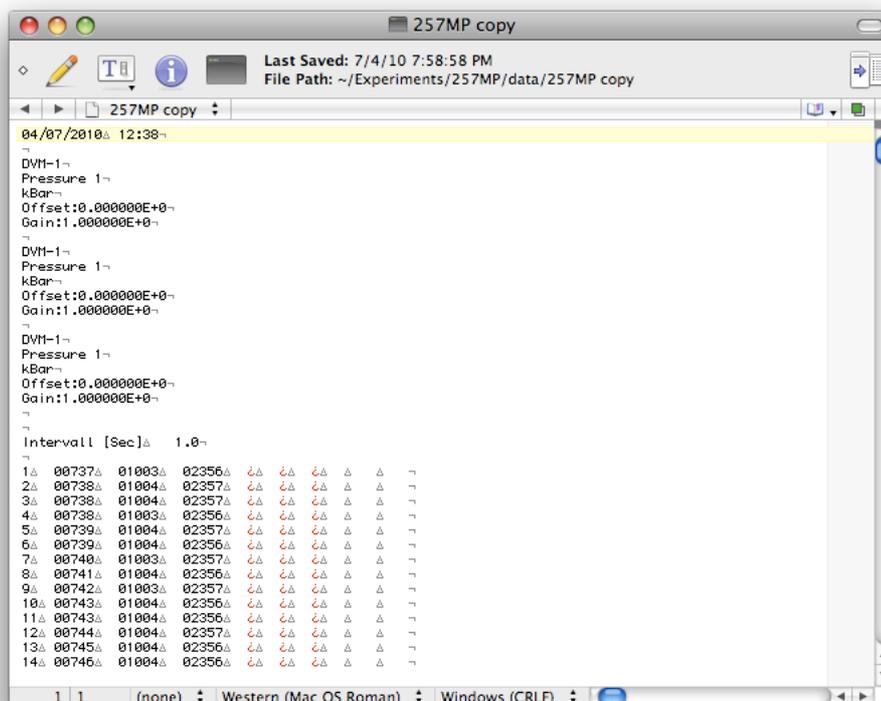
type “edit rename” into the command window. Then you can do “find & replace” and replace all “_255” with the number or name of your experiment.

2.2.9 plots.m

A cell script that contains all plot commands generated during *rig.m* program. Treat it the same way as *rename.m*

3.Troubleshooting

If you can't import a file open it in Textwrangler or any other text editor end select the option "show invisibles" usually there are some extra weird symbols after the 4th/5th column - just select one line and do "search and replace" to get rid of them. Make sure that every line is terminated by “enter” (and not “tab enter”)



If your file imported into MATLAB is not complete check the line at which the import stops in textwrangler (“apple + J” is “go to line”) if the data is not recorded due to buffer overflow etc. MATLAB stops importing at this point. Delete the lines without data and try once again.

HPkN – “traditional” hit-point value in kN (see fig. 4)

RIG – number/name of the rig used

RigStiff – correction factor for rig stiffness for the individual rig [*mm/kN*]

Alfa - angle of the shear zone with respect to σ_1 [$^\circ$]

Area - area of the sample [*mm²*]

disrateNoCor – displacement rate calculated from displacement not corrected for rig stiffness [*mm/sec*]

disrate - displacement rate calculated from displacement corrected for rig stiffness [*mm/sec*]

dxC – direct-current displacement transducer (DC-DT) data recalculated to mm from the hit point (i.e displacement at hit-point = 0).

dxCon – conversion factor to recalculate volts to mm for DC-DT

dxStiffCor – DC-DT data recalculated to mm and corrected for rig stiffness

dx – displacement in millivolts smoothed by the forward and reverse running average filter (used for all calculations instead of dxi)

extype – coaxial experiment (1) or shear experiment (2)

fCkN – force in kN from the hit-point (i.e force at hit-point = 0)

fConkN – conversion factor to recalculate volts to kN for the load cell

fHPkN – force at data hit-point in kN (see fig. 4)

f – force in millivolts smoothed by the forward and reverse running average filter (used for all calculations instead of fi)

fi – force input [*mV*]

fit1 – a and b parameters ($y=ax + b$) for the run-in curve

fit2 – a and b parameters ($y=ax + b$) for the “elastic” curve

friect – force corrected for friction (using uncorrected dx/DLT)

gamma – shear strain

name – name/number of the experiment

no – number of points used for the forward and reverse running average filter

pcC – change in confining pressure (in MPa) with respect to “traditional” hit-point (i.e confining pressure at hit-point = 0)

pcCon – conversion factor to recalculate volts to MPa for the confining pressure

pcMPa – confining pressure in MPa

pc – confining pressure in millivolts smoothed by the forward and reverse running average filter (used for all calculations instead of pci)

pccorrection – has the correction for changing confining pressure been used? (y/n)

pci – confining pressure input

runinintersectMPa – run-in hit-point in MPa (using initial area of the sample) (see fig. 4)

runinintersectkN – run-in hit-point in kN (see fig. 4)

sdia – sample diameter in mm

sdif – sdifNoCor corrected for changing confining pressure (if the user has chosen not to apply the changing confining pressure correction it will be identical to sdifNoCor) [*MPa*]

sdifAreaCor - differential stress corrected for changing area and changing confining pressure (if the user has chosen not to apply the changing confining pressure correction it will be identical to sdifrawAreaCor) [*MPa*]

sdifNoCor – differential stress assuming that the sample has retained the

original area throughout the deformation (i.e. no changing area correction)
[MPa].

sdifrawAreaCor – differential stress corrected for changing area [MPa]

slen – sample length [mm]

shearrateF – finite shear rate [sec^{-1}]

shearrateI – instantaneous shear rate [sec^{-1}]

shearrateR – shear rate compatible with rigS4.f [sec^{-1}]

strain – strain [%]

strainrateeng – engineering (instantaneous) strain rate [sec^{-1}]

srainratenat – natural (finite) strain rate [sec^{-1}]

strainraw – (sample length – displacement) / sample length

tHPfromstart – time to hit-point from the beginning of recording [sec]

tau – shear stress [MPa]

tday – time in days

th0 – original thickness of the shear zone [mm]

thf – final thickness of the shear zone [mm]

thour – time in hours

ti – time input

torig – saves the original time (including pumping if it was recorded in the same file) [sec]

ts – time record interval in seconds

tsec – time in seconds

xintersect – position of intersection of the run-in and elastic curves on the x-axis [ti]

yintersect - position of intersection of the run-in and elastic curves on the y-axis [mV]