Niget: NanoIndentation General Evaluation Tool

Anna Charvátová Campbell, Radek Šlesinger, Petr Grolich
Czech Metrology Institute

November 19, 2018

Contents

1 About

Nanoindentation instruments often include data processing software, however, this may not be sufficient. The desired evaluation method may not implemented, the implementations may not be suitable for the given physical problem, the description of the implementation may be insufficient or more information about uncertainties is sought for.
This toolbox aims to fill this gap (at least partly). It is open-source, so the implementation

can be checked by the user. It can also be easily expanded to include also other evaluation methods. Uncertainties are provided both in the standard framework of uncertainty propagation and also using Monte Carlo simulations.

Visit Niget home page at http://nanometrologie.cz/niget.

2 System requirements, compiling, and installation

Niget sources as well as binaries for 32-bit Windows systems can be downloaded from its home page at http://nanometrologie.cz/niget. The software is written mainly in C language using GTK+ (version 2) toolkit (http://www.gtk.org) and libraries from Gwyddion data analysis software (http://gwyddion.net). Some tools (Oliver-Pharr ODR, Hertz ODR, Two slopes, and Stiffness) use orthogonal distance regression (ODR) which includes Fortran code from ODRPACK95 project, available at http://www.netlib.org/toms/869.zip.

2.1 Linux

There are no distribution packages available, and users are supposed to compile Niget from source.

Requirements

1. C compiler (preferably GNU gcc or Intel icc)
2. (optional) Fortran compiler (preferably GNU gfortran or Intel ifort)
3. CMake
4. GNU Make or compatible
5. pkg-config
6. GTK2 (and its dependences), including development libraries
7. (preferable) Gwyddion development libraries (see http://gwyddion.net/download.php for distribution-specific instructions; FFTW3 and GtkGLExt development libraries may be also required as dependences)

If paths to Gwyddion libraries and includes are not found by CMake or provided by user, a recent version of Gwyddion is automatically downloaded and built. Please that this does not include any additional tools or libraries which might be required by Gwyddion; these must be installed manually according to the installation instructions of Gwyddion.

Compiling with CMake In the Niget source directory, proceed as follows:

1. `mkdir build` (out-of-tree builds are preferred with CMake)
2. `cd build`
3. `cmake ..` (CMake looks for compilers and libraries, and configures the build)
4. `make`

This compiles Niget using default configuration. If CMake finds a suitable Fortran compiler, ODRPACK95 will be compiled and ODR-based tools enabled. Optional configuration parameters can be set by adding `-D OPTION=VALUE` to the `cmake` command (`cmake -D OPTION1=VALUE1 ... -D OPTIONn=VALUEn ..`). Presently available options are:

1. `DEBUG` – `ON` (default) / `OFF`: make debug build
2. `VERBOSE` – `ON` / `OFF` (default): increase verbosity of some tools for debugging purposes

Specific compilers can be provided using `CC` and `FC` environment variables, e.g. `CC=icc FC=ifort cmake ..` to use Intel compilers. After running CMake, the options above are stored into CMake's cache in the build directory, and need not be specified with further CMake runs (or have to be specified explicitly if a change is desired). Note: until the code is sufficiently tested, `DEBUG` is `ON` by default, and release build must be triggered manually.

After the software compiles successfully, the `niget` binary, created in the build directory, can be run.

2.2 Windows

Niget Windows 32-bit binaries are distributed in a single zip-file, which contains all the required libraries. This is the preferred way for Windows users to start using Niget.

2.2.1 MinGW suite

Compiling using CMake and MinGW suite in MSYS2 environment has been tested and is currently used to provide the Windows builds of Niget. Unfortunately, no straightforward procedure is available at the moment.

3 Main Window

Niget's graphical user interface (GUI) can be started by running the `niget` (`niget.exe`) executable. Using two optional parameters *filename* and *fileformat* (currently available options: *niget*, *unht*, *hysitron*), a file of a given type is opened on application start by running `niget -f fileformat filename`.

The main window shows all currently available methods as different tabs and a row of general function buttons. The tabs are inactive at the start of the program and get activated when a file is opened.

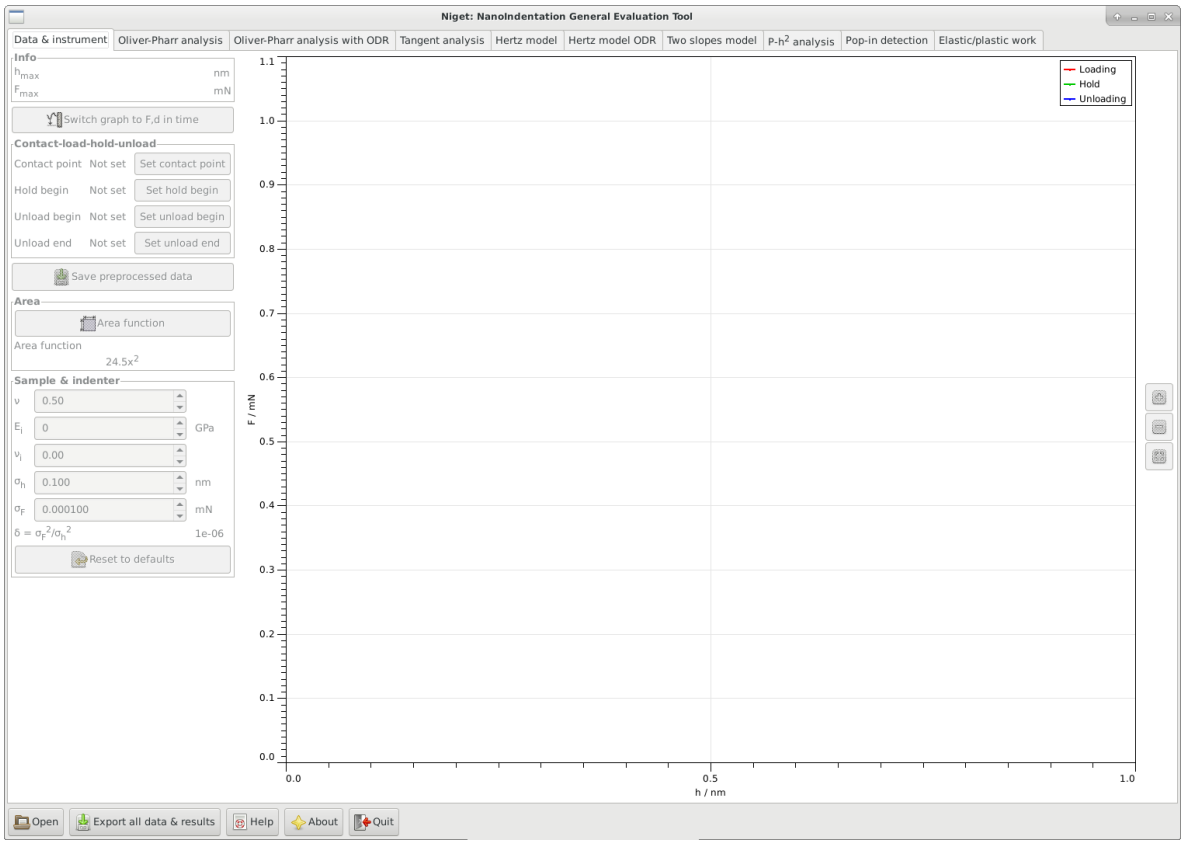

Figure 1: Main window

The buttons are:

- *Open:* Load a file in a chosen format. Currently, only files of one of the predefined plain-text formats. can be opened. Each file should contain only one loading-unloading curve. Files with more than one indentation should load as well, but the behaviour may be "surprising". Currently, only the decimal point is supported as the decimal mark. Lines that do not correspond to the given format, because they are, e.g., empty or hold comments, are skipped.
  WARNING: The units MUST agree with the given format or you will get nonsense numerical results!

  - `Time (s)  Depth (nm)  Load (mN)`: first three columns correspond to time (which

is skipped), depth in nm and load in mN.

- `Time (s)  Depth (nm)  Load (uN)`: first three columns correspond to time (which is skipped), depth in nm and load in μN.
- `Depth (nm)  Load (uN)`: first two columns correspond to depth in nm and load in μN.
- `Depth (nm)  Load (mN)`: first two columns correspond to depth in nm and load in mN.
- `Load (mN)  Depth (nm)`: first two columns correspond to load in mN and depth in nm.
- `Load (uN)  Depth (nm)`: first two columns correspond to load in μN and depth in nm.
- `Niget`: native format of Niget.

The file cannot be loaded if it's not in the given format or if it contains `inf` or `nan` values. The data are supposed to be sufficiently comprehensive in order to define the contact point as well the loading and unloading part.

- *Export all data & results:* Save the processed indentation data and results from all methods. The user chooses a basename to which a suffix is appended for each method. A file is created even if there are no meaningful results for a method.
- *Help:* Open HTML documentation in an associated browser.
- *About:* Display additional information about the software.
- *Quit:* Exit the program.

Note that when the program exits (either by pressing *Quit* or closing the main window), no results are saved.
The program keeps its own few settings for user comfort. These are saved in `niget_settings.cfg` in the user configuration directory.

4 Data & area

In this tab the user can set the contact point as well as other important points of the indentation process, mechanical properties of the tip and sample, indenter noise and define the area function. After successfully loading a file, the software creates the force-distance diagram from the data, and attempts to automatically detect the loading, hold and unloading parts of the data.

4.1 Window

The window consists of the following:

- *Info* displays the maximum depth and force during the indentation determined as in section 4.2.

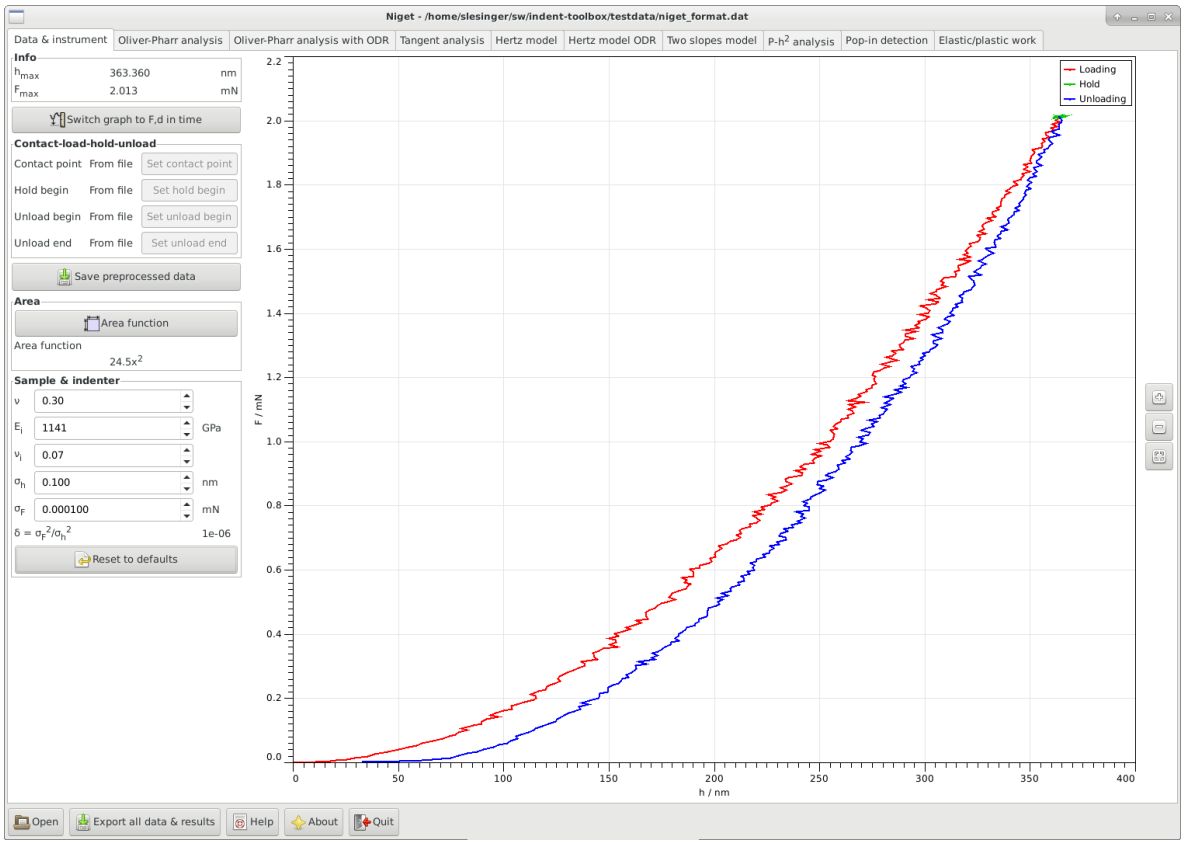

Figure 2: Force-distance diagram after automatic split

- *Switch graph to ...* switches between two display modes: force as a function of depth (F-d) and force and depth as functions of the (pseudo)time (F,d-t). The index of the data point is used instead of the real time, which is not read from the file. In the second case, both curves are displayed dimensionless, and scaled just to provide good visual resolution.

- *Contact-load-hold-unload* Allows to manually split the data into the loading, hold and unloading parts:

  - Contact point: determines the beginning of the loading curve. The depth and force at this point h_c and F_c are subtracted from the loading-unloading curves.
  - Hold begin
  - Unload begin
  - Unload end: define the end of the unloading part. By default, the data are truncated at zero depth.

  Whether the point was set automatically or manually is shown. Note, that if the division of the data into the different parts is changed, all analysis results are deleted!

- *Save preprocessed data* saves the data including the selected split of the data in the native format Niget.

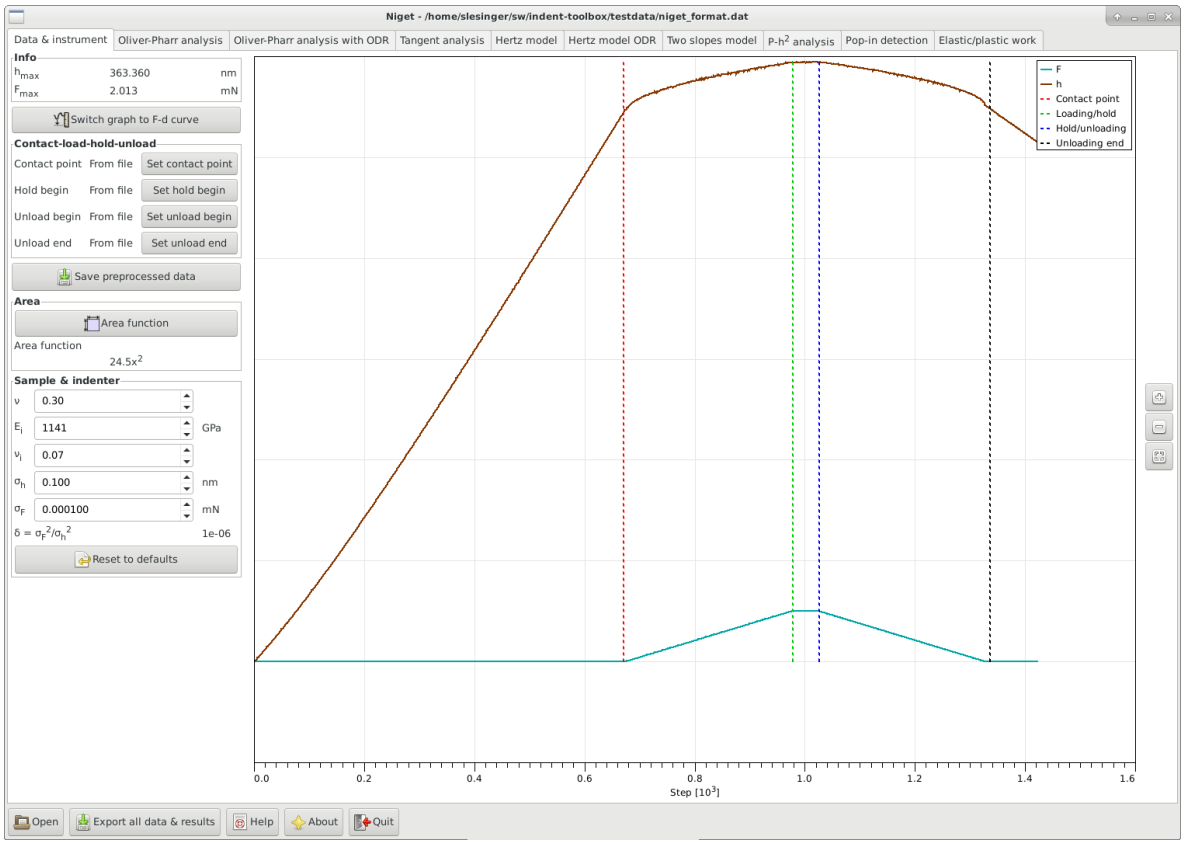

Figure 3: Manual definition of points in force-distance in time view

- *Area*
  - Area function button: opens a separate dialog for the definition of the area function, see 4.3
  - displays the area function used currently.
- *Sample & indenter* The parameters here are Poisson's value of the sample, Young's modulus of the indenter, Poisson's value of the indenter and the noise of the displacement and load sensors. The default values are 0.3, 1141 GPa, and 0.07. The first is a reasonable estimate of the often unknown Poisson's value for many materials, the other two values are the literature values for diamond which is a common indenter tip material. The noise of the sensors is used for the fitting procedures and for the uncertainty analysis. These values are saved in settings and can be reset to their default values.
- *Graph* displays the indentation curve. Stepwise zooming/unzooming can be performed by selecting a range with the mouse and pressing the *Zoom/ Unzoom* buttons. The graph is restored to its original size by the *Restore* button. The zooming procedure is independent in the two regimes (F-d vs. F,d-t).

4.2 Maximum depth and force

The maximum force $F_{\max}$ is the maximum force value found in the unloading data. The maximum depth $h_{\max}$ is the corresponding depth, NOT the maximum depth value!

4.3 Area function

The area function can be given either in form of a polynomial of the form

$$A(h) = c_2 h^2 + c_1 h + c_{1/2} h^{1/2} + c_{1/4} h^{1/4} + c_{1/8} h^{1/8} + c_{1/16} h^{1/16}, \tag{1}$$

or a raw data file can be loaded, which will be (linearly) interpolated in subsequent calculations. The format of the area data file MUST be two columns: first column depth in nm, second column area in nm^2.
The area function is shown for user information. A warning is issued if a raw data file is used and any of the methods extrapolates the area. The coefficients of a polynomial area function are saved in the settings file.

For specific formats, the coefficients can also be loaded directly from a file. Currently, this should work for an `.ara` file (as exported from a Hysitron instrument) or for an `.ind` file (as exported from a UNHT Anton Paar instrument). This is under testing and may not work correctly for other versions.

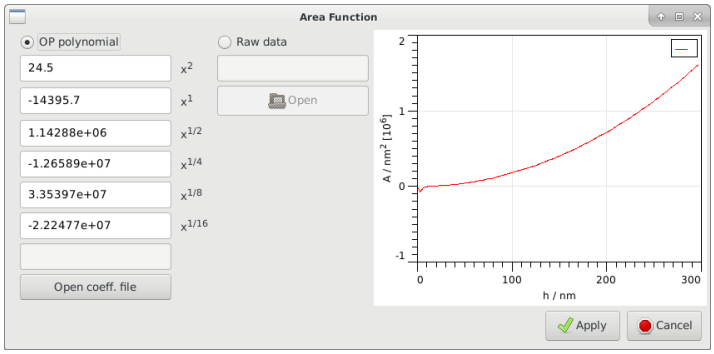

Figure 4: Area function dialog

5 Oliver Pharr

For a definition of the Oliver Pharr method see [1].

5.1 Window

The window consists of several blocks:

- *Info* displays the maximum depth and force during the indentation
- *Parameters* shows the selected range in nm and in % of the maximum force, and the correction β.

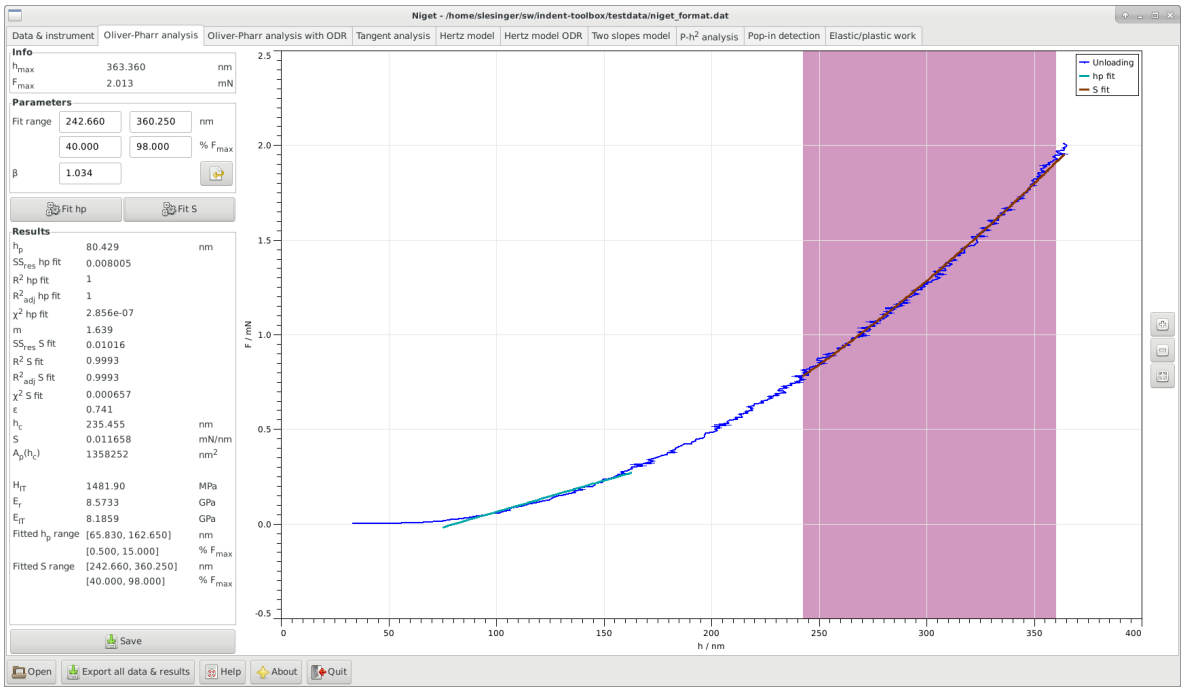

Figure 5: Oliver-Pharr analysis

- The fitting range can be selected either using the mouse or typing in the range entries. The range can be defined either in nm or in percent of the maximum force. It is often recommended to use the range 0.5–15 % F_{max} for the hp fit and 40–98 % F_{max} for the S fit, see section 5.2.
- The parameter β accounts for any deviations from the axisymmetric case and is used in the calculation of the reduced modulus in equation (10). Currently, the default value is the value for three-sided pyramides $\beta = 1.034$. The value supplied by the user is saved in the settings and can be reset to its default value.

- *Fit* buttons for the two fits, see section 5.2 for details of the calculation.
- *Results* displays all results in the following order: the residual depth h_p, the power m of the power law function, the parameter ε, the contact depth h_c, the slope S, the contact area $A_p(h_c)$, the indentation hardness H_{IT}, the contact modulus E_r, the indentation modulus E_{IT} and the ranges used for the fitting procedures. The variables are described in detail in section 5.2.
- *Save* save parameters and results to given file.
- *Graph* display the unloading curve and the fitted curves. Stepwise zooming/unzooming can be performed by selecting a range with the mouse and pressing the *Zoom/ Unzoom* buttons. The graph is restored to its original size by the *Restore* button.

5.2 Procedure

The standard calculation consists of three steps

1. The residual depth must be determined as the intersection of the unloading curve and the x-axis. This is implemented by fitting a straight line using a Deming fit with $\delta = 1$, see section A.2. For a brief description of the Deming fit see section A.2

$$F = a_{h_p} h_p + b_{h_p} \tag{2}$$

using total least squares. This fit is called the h_p fit. The residual depth is calculated as

$$h_p = -b_{h_p}/a_{h_p}. \tag{3}$$

The range of the data must be chosen adequately, the range 0.5–15 % F_{max} is often a reasonable value.

2. The main part of the unloading curve is fitted by a power law function

$$F = \alpha(h - h_p)^m. \tag{4}$$

This is converted to a total least squares fit in the variables using a Deming fit with $\delta = 1$, see section A.2

$$\log F = \log \alpha + m \log(h - h_p). \tag{5}$$

The range should not contain the lower part of the unloading range, a range of 40–98 % F_{max} is recommended as a first guess.

3. The auxiliary parameter ε is calculated from the power m

$$\varepsilon = m \left(1 - \frac{2(m-1)\Gamma\left(\frac{m}{2(m-1)}\right)}{\sqrt{\pi}\Gamma\left(\frac{1}{2(m-1)}\right)} \right), \tag{6}$$

Γ is the Gamma-function.
The contact depth is calculated as

$$h_c = h_{max} - \epsilon \frac{F_{max}}{S} \tag{7}$$

and the slope at the maximum depth as

$$S = m \frac{F_{max}}{h_{max} - h_p}. \tag{8}$$

4. The contact depth is used to evaluate the contact area $A(h_c)$. This can be used to find the indentation hardness

$$H_{IT} = \frac{F_{max}}{A(h_c)}. \tag{9}$$

and together with the slope to find the contact modulus

$$E_r = \sqrt{\pi} \frac{S}{2\beta\sqrt{A(h_c)}}. \tag{10}$$

For a comparison with Young's modulus found in literature it is useful to calculate the indentation modulus E_{IT}

$$E_{IT} = \frac{1 - \nu^2}{1/E_r - (1 - \nu_i^2)/E_i}. \tag{11}$$

Here ν is the Poisson's value of the sample and ν_i and E_i are the Poisson's value and the modulus of the indenter.

6 Oliver Pharr with ODR

For a definition of the Oliver Pharr method see [1].
This method is shown unless the software was compiled without Fortran support.

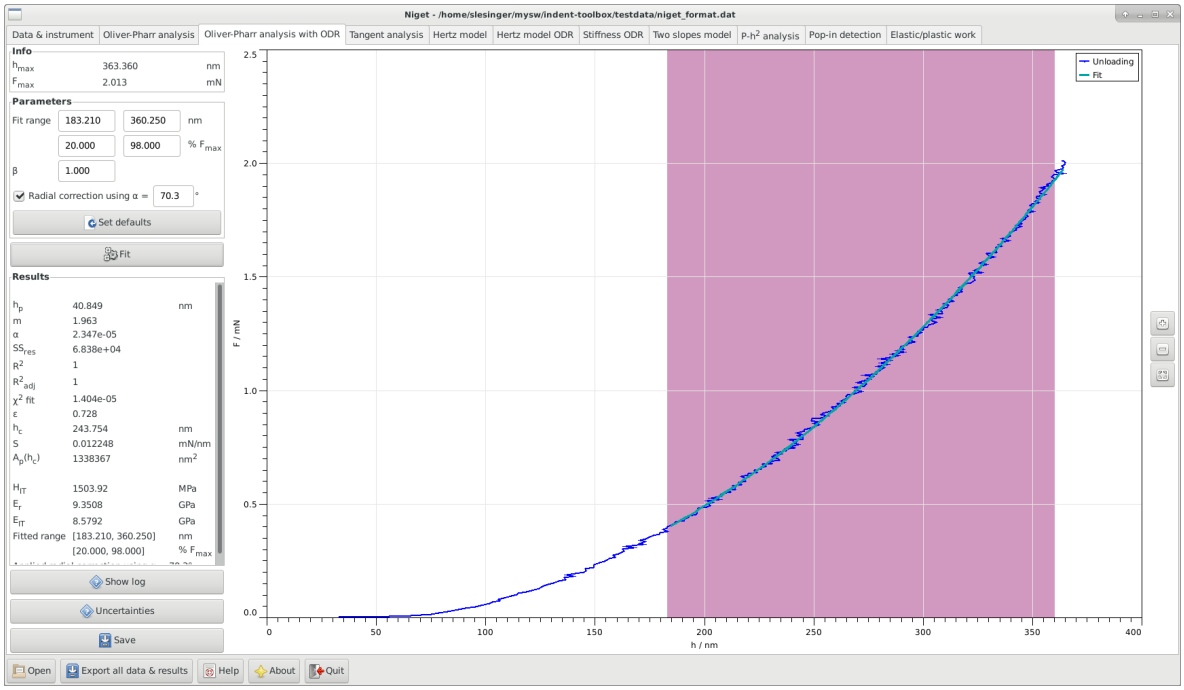

Figure 6: Oliver-Pharr analysis using orthogonal data regression

6.1 Window

The window consists of several blocks:

- *Info* displays the maximum depth and force during the indentation
- *Parameters* shows the selected range in nm and in % of the maximum force, the correction β, and the radial correction.
  - The fitting range can be selected either using the mouse or typing in the range entries. The range can be defined either in nm or in percent of the maximum force. It is often recommended to use the range 40–98 % F_{max} for the fit, see section 6.2.
  - The parameter β accounts for any deviations from the axisymmetric case and is used in the calculation of the reduced modulus in equation (10). Currently, the default value is no correction $\beta = 1.0$. The value supplied by the user is saved in the settings and can be reset to its default value.
  - Optional radial correction of calculated hardness and modulus values according to ISO 14577-1:2015. The angle α denotes the cone-equivalent angle between the tip side and axis. For a Berkovich tip, $\alpha = 70.3°$.

– *Set defaults* button resets the fitting range, the beta value, and radial correction settings according to recommendations provided in ISO 14577-1.

- *Fit* button, see section 6.2 for details of the calculation.
- *Results* displays all results in the following order: the residual depth h_p, the power m of the power law function, the parameter ε, the contact depth h_c, the slope S, the contact area $A_\mathrm{p}(h_\mathrm{c})$, the indentation hardness H_{IT}, the contact modulus E_r, the indentation modulus E_{IT} and the ranges used for the fitting procedure. The variables are described in detail in section 6.2. If the fittings procedure failed a warning is shown.
- *Uncertainties* show the uncertainty analysis window, see section 15.2.1.
- *Show log* Show the report about the fitting procedure in a separate window. The reports are saved to files *fit.log.op.err* and *fit.log.op.rpt*.
- *Save* save parameters and results to given file.
- *Graph* display the unloading curve and the fitted curves. Stepwise zooming/unzooming can be performed by selecting a range with the mouse and pressing the *Zoom/ Unzoom* buttons. The graph is restored to its original size by the *Restore* button.

6.2 Procedure

This is a slight modification of the standard Oliver Pharr method described in section 5.2 using a better fitting procedure.

1. Fit the upper part of the unloading curve with a power law function

$$F = \alpha(h - h_\mathrm{p})^m.$$

   using orthogonal least squares as implemented in the package ODRPACK95 [2]. The range should be approx. 40–98 % $\mathrm{F_{max}}$. All three parameters are fitted.

2. Same as steps 3–4 in 5.2.

3. If the radial correction is on, the hardness and reduced modulus are found by iterating the following relations

$$H^n = H^0 \left(1 + K\frac{H^{n-1}}{E_{IT}^{n-1}}\right)^2 \tag{12}$$

$$E_r^n = E_r^0 \left(1 + K\frac{H^{n-1}}{E_{IT}^{n-1}}\right) \tag{13}$$

   where H^0 and E_r^0 are the initial estimates obtained from step 4 in section 5.2. The iteration limit is set to a relative difference $0.5 \cdot 10^{-4}$ Young's modulus is updated in each step according to (11) and K is the radial correction factor computed from Poisson's ratio and the angle between tip sides and axis α

$$K = \frac{1 - 2\nu}{2(1 + \nu)} \sin\alpha. \tag{14}$$

The area function is corrected with the final values of hardness and modulus as

$$A = A^0 \left(1 + K \frac{H}{E_{IT}}\right)^2 . \tag{15}$$

7 Tangent method

The method is based on the linear model as described in [3]. It is recommended only for use with highly plastic materials where the depth of elastic recovery is less than 10 % of max.

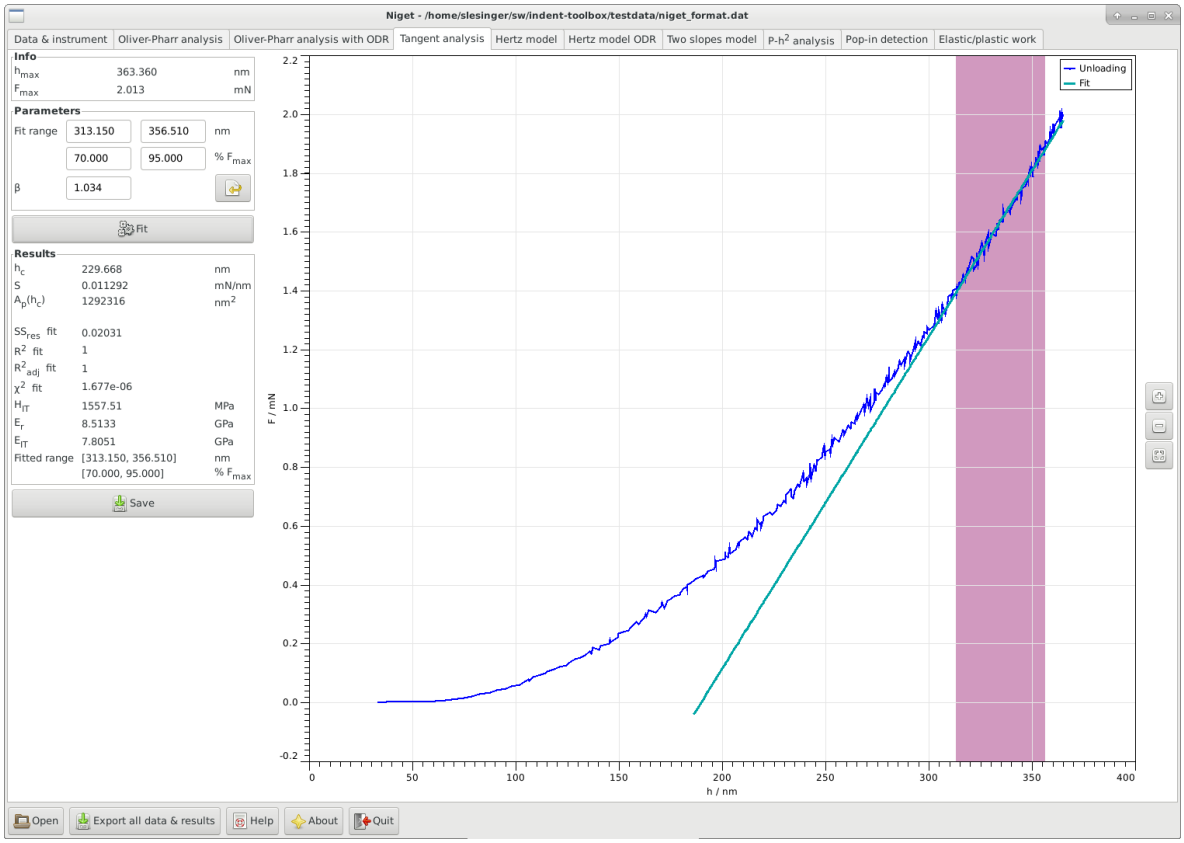

Figure 7: Tangent method analysis

7.1 Window

The window consists of several blocks:

- *Info* displays the maximum depth and force during the indentation
- *Parameters* shows the selected range in nm and in % of the maximum force, and the correction β.
  - The fitting range can be selected either using the mouse or typing in the range entries. The range can be defined either in nm or in percent of the maximum force. It is often recommended to use the range 70–95 % F_{max} for the fit, see section 7.2.

- The parameter β accounts for any deviations from the axisymmetric case and is used in the calculation of the reduced modulus in equation (10). Currently, the default value is the value for three-sided pyramides $\beta = 1.034$. The value supplied by the user is saved in the settings and can be reset to its default value.

- *Fit* button, see section 7.2 for details of the calculation.

- *Results* displays all results in the following order: the contact depth h_c, the slope S, the contact area $A_p(h_c)$, the indentation hardness H_{IT}, the contact modulus E_r, the indentation modulus E_{IT} and the ranges used for the fitting procedure. The variables are described in detail in section 7.2.

- *Save* save parameters and results to given file.

- *Graph* display the unloading curve and the fitted curves. Stepwise zooming/unzooming can be performed by selecting a range with the mouse and pressing the *Zoom/ Unzoom* buttons. The graph is restored to its original size by the *Restore* button.

7.2 Procedure

The tangent method uses a different approach to determine the slope of the unloading curve at its maximum depth than the Oliver Pharr method.

1. Fit the uppermost part of the unloading curve with a straight line using a Deming fit with $\delta = \sigma_F^2/\sigma_h^2$, see section A.2,
$$F = Sh + q,$$
using total least squares. The range should be approx. 70–95 % F_{max}.

2. Set $\varepsilon = 0.75$ and calculate the contact depth as
$$h_c = h_{max} - \varepsilon F_{max}/S. \tag{16}$$

3. Same as step 4 in 5.2.

8 Hertz method

The Hertz method is the application of the Hertzian model of elastic contact to the initial stage of the loading curve [4]. It can be used either to estimate the modulus if the tip radius is known or vice versa, depending what information is available.

8.1 Window

The window consists of several blocks:

- *Info* displays the maximum depth and force during the indentation

- *Parameters* shows the selected range in nm.

  - The input variable can be chosen to be either the tip radius, the reduced modulus or the indentation modulus and its value should be set accordingly.

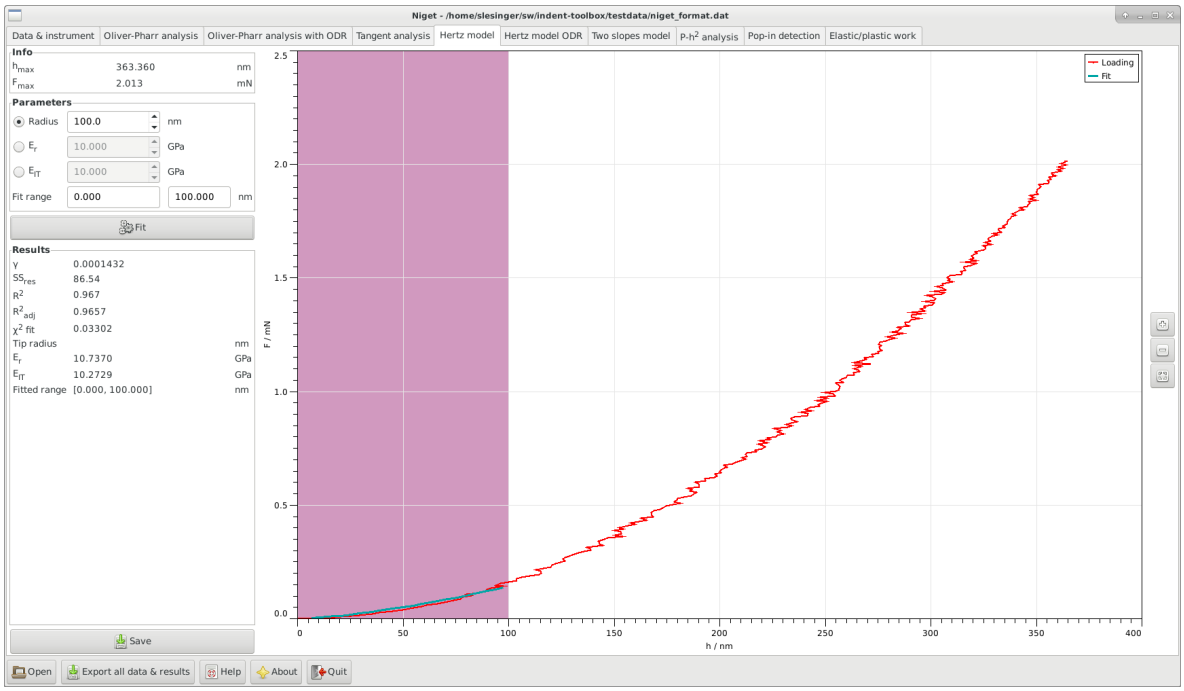

Figure 8: Hertzian model analysis

- The fitting range can be selected either using the mouse or typing in the range entries. The range must be chosen so that the behavior remains elastic and the fit adequate.

- *Fit* button, see section 8.2 for details of the calculation.
- *Results* displays the results and the ranges used for the fitting procedure. The variables are described in detail in section 8.2.
- *Save* save parameters and results to given file.
- *Graph* display the unloading curve and the fitted curves. Stepwise zooming/unzooming can be performed by selecting a range with the mouse and pressing the *Zoom*/ *Unzoom* buttons. The graph is restored to its original size by the *Restore* button.

8.2 Procedure

1. The Hertzian model predicts for the contact between sphere and halfspace the following shape of the loading curve

$$F = ah^{3/2}, \tag{17}$$

with

$$a = \frac{4}{3} E_{\mathrm{r}} \sqrt{R}. \tag{18}$$

This is fitted using ordinary least squares with an additional intercept possible, see section A.3.

2. If the tip radius is given, the contact modulus is calculated as

$$E_\mathrm{r} = \frac{3}{4}\frac{a}{\sqrt{R}}. \tag{19}$$

For a comparison with Young's modulus found in literature the indentation modulus E_{IT} (11) is useful

3. If the reduced modulus is given, the tip radius is calculated as

$$R = \left(\frac{3}{4}\frac{a}{E_\mathrm{r}}\right)^2 \tag{20}$$

4. If the indentation reduced modulus is given, the material parameters ν, ν_i and E_i are used to convert it to the reduced modulus

$$E_\mathrm{r} = \left(\frac{1-\nu_\mathrm{i}^2}{E_\mathrm{i}} + \frac{1-\nu^2}{E_\mathrm{IT}}\right)^{-1}. \tag{21}$$

from which the tip radius can be calculated as in the previous step.

9 Hertz method with ODR

This is a modification of the Hertz method explained in section 8 and uses orthogonal distance regression. The loading curve is fitted with a more general power function but the relationship between the proportionality coefficient and the radius and modulus is assumed to stay the same. If the resulting power differs from the theoretical value 1.5, this is an indicator that the Hertzian model is not adequate. The results for radius or modulus in this case do not make any physical sense.
This method is shown unless the software was compiled without Fortran support.

9.1 Window

The window consists of several blocks:

- *Info* displays the maximum depth and force during the indentation
- *Parameters* shows the selected range in nm.
  - The input variable can be chosen to be either the tip radius, the reduced modulus or the indentation modulus and its value should be set accordingly.
  - The fitting range can be selected either using the mouse or typing in the range entries. The range must be chosen so that the behavior remains elastic and the fit adequate.
  - Check box, whether or not the exponent should be fitted or fixed at the value 1.5, see 9.2.
- *Fit* button, see section 9.2 for details of the calculation.

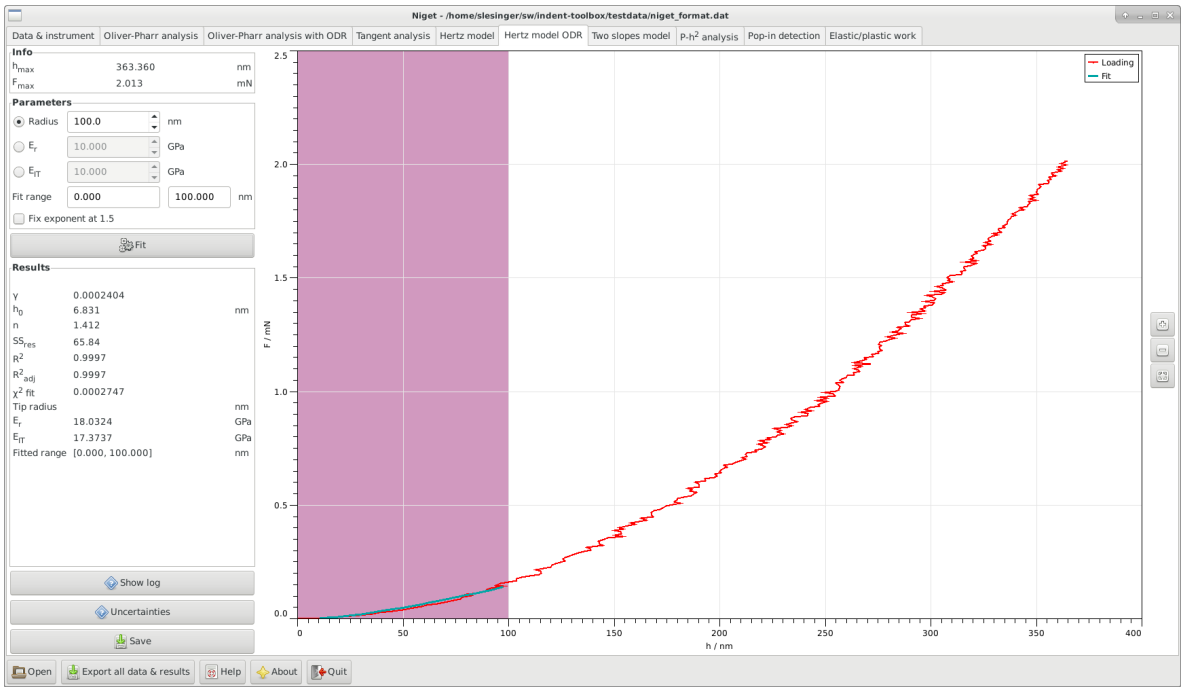

Figure 9: Hertzian model with ODR analysis

- *Results* displays the results and the ranges used for the fitting procedure. The variables are described in detail in section 9.2.
- *Show log* Show the report about the fitting procedure in a separate window. The reports are saved to files *fit.log.hz.err* and *fit.log.hz.rpt*.
- *Uncertainties* show the uncertainty analysis window, see section 15.2.2.
- *Save* save parameters and results to given file.
- *Graph* display the unloading curve and the fitted curves. Stepwise zooming/unzooming can be performed by selecting a range with the mouse and pressing the *Zoom/ Unzoom* buttons. The graph is restored to its original size by the *Restore* button.

9.2 Procedure

1. The Hertzian model (17) is fitted with a more general function

$$F = \gamma(h - h_0)^n, \tag{22}$$

using orthogonal least squares implemented in ODRPACK. The power n can be fixed at the theoretical value 1.5.

2. Further steps are the same as steps 2–4 in section 8.2 with γ used instead of a.

10 Stiffness method

This module allows the determination of the stiffness of an object by fitting both loading and unloading curve with a straight line. This method is shown unless the software was compiled without Fortran support.

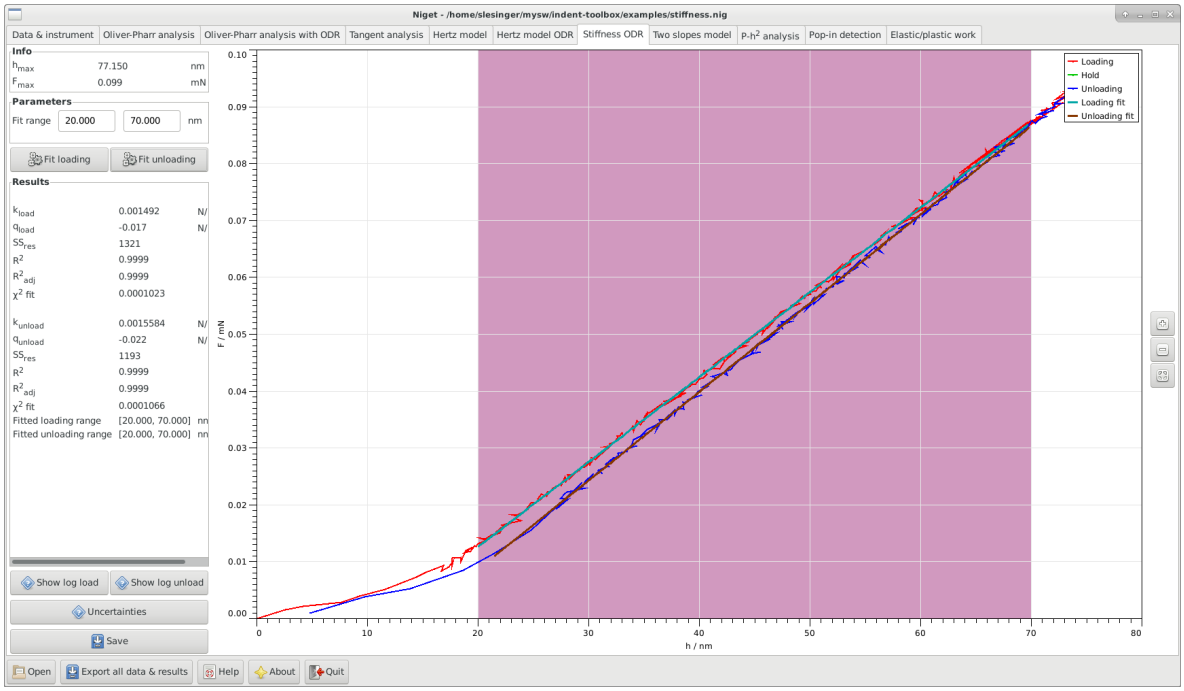

Figure 10: Stiffness analysis using orthogonal data regression

10.1 Window

The window consists of several blocks:

- *Info* displays the maximum depth and force during the indentation
- *Parameters* shows the selected range in nm and in % of the maximum force, and the correction β.
  - The fitting range can be selected either using the mouse or typing in the range entries.
- *Fit* buttons for indepedent fitting of the loading and unloading curves, see section 10.2 for details of the calculation.
- *Results* displays all results in the following order: the slope of the loading curve k_{load}, the offset of the loading curve q_{load}, the residual sum of squares of the corresponding fit and the goodness of fit parameters, the slope of the unloading curve k_{unload}, the offset of the loading curve q_{unload}, the residual sum of squares of the corresponding fit and the

goodness of fit parameters and the ranges used for the fitting procedures. Warnings are displayed if the fittings procedures failed.

- *Uncertainties* show the uncertainty analysis window, see section 15.2.3.
- *Show log load* Show the report about the fitting procedure of the loading curve in a separate window. The reports are saved to files *fit.log.stiff.load.err* and *fit.log.stiff.load.rpt*.
- *Show log unload* Show the report about the fitting procedure of the unloading curve in a separate window. The reports are saved to files *fit.log.stiff.unload.err* and *fit.log.stiff.unload.rpt*.
- *Save* save parameters and results to given file.
- *Graph* display the unloading curve and the fitted curves. Stepwise zooming/unzooming can be performed by selecting a range with the mouse and pressing the *Zoom/ Unzoom* buttons. The graph is restored to its original size by the *Restore* button.

10.2 Procedure

The stiffness analysis simply fits both loading and unloading curve with straight lines.

1. Fit the loading curve with a straight line

$$F = k_{\text{load}} h + q_{\text{load}}$$

   using orthogonal least squares as implemented in the package ODRPACK95 [2].

2. Fit the unloading curve with a straight line

$$F = k_{\text{unload}} h + q_{\text{unload}}$$

   using orthogonal least squares as implemented in the package ODRPACK95 [2].

11 Two slopes method

For a description of the two slopes method see [5].
This method is shown unless the software was compiled without Fortran support.

11.1 Window

The window consists of several blocks:

- *Info* displays the maximum depth and force during the indentation
- *Parameters* shows the selected range in nm and in % of the maximum force, and the correction β.
  - The fitting range can be selected either using the mouse or typing in the range entries. The range can be defined either in nm or in percent of the maximum force. It is often recommended to use the range 80–98 % F_{max} for the fit, see section 11.2.

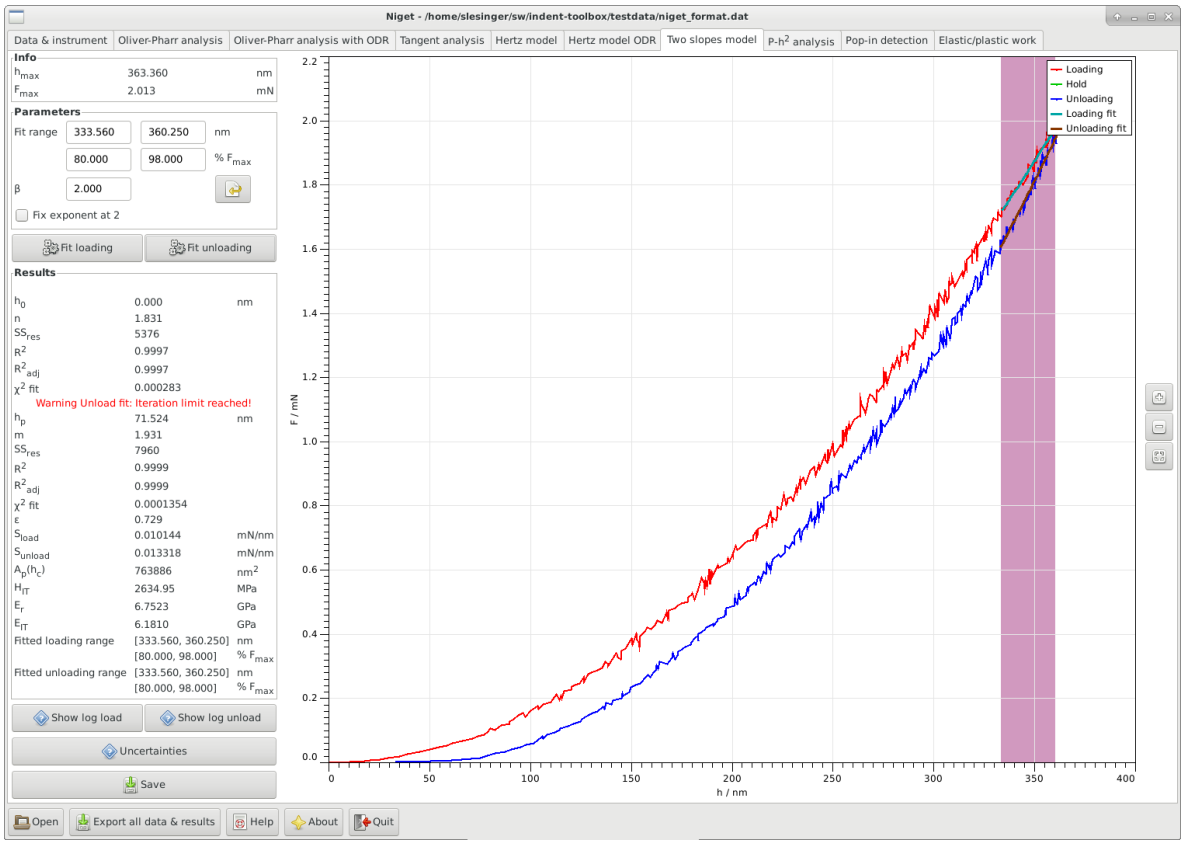

Figure 11: Two slopes analysis using orthogonal data regression

- The parameter β accounts for any deviations from the axisymmetric case and is used in the calculation of the reduced modulus in equation (10). Currently, the default value is no correction $\beta = 1.0$. The value supplied by the user is saved in the settings and can be reset to its default value.
- Check box, whether or not the exponent of the loading curve should be fitted or fixed at the theoretical value 2, see 11.2.

- *Fit* buttons for indepedent fitting of the loading and unloading curves, see section 11.2 for details of the calculation.
- *Results* displays all results in the following order: the auxiliary depth parameter h_0, the power n of the power law loading function, the residual depth h_p, the power m of the power law unloading function, the parameter ε, the loading slope S_load, the unloadingslope S_unload, the contact area $A_\mathrm{p}(h_\mathrm{c})$, the indentation hardness H_{IT}, the contact modulus E_r, the indentation modulus E_{IT} and the ranges used for the fitting procedures. The variables are described in detail in section 11.2. Warnings are displayed if the fittings procedures failed.
- *Uncertainties* show the uncertainty analysis window, see section 15.2.4.
- *Show log load* Show the report about the fitting procedure of the loading curve in a separate window. The reports are saved to files *fit.log.slopes.load.err* and *fit.log.slopes.load.rpt*.

- *Show log unload* Show the report about the fitting procedure of the unloading curve in a separate window. The reports are saved to files *fit.log.slopes.unload.err* and *fit.log.slopes.unload.rpt*.
- *Save* save parameters and results to given file.
- *Graph* display the unloading curve and the fitted curves. Stepwise zooming/unzooming can be performed by selecting a range with the mouse and pressing the *Zoom/ Unzoom* buttons. The graph is restored to its original size by the *Restore* button.

11.2 Procedure

The two slopes method combines the standard Oliver Pharr method and the quadratic loading curve to avoid the need of the contact area.

1. Fit the upper part of the loading curve with a power law function

$$F = \gamma(h - h_0)^n.$$

   using orthogonal least squares as implemented in the package ODRPACK95 [2]. The range should be approx. 85–98 % F_{max}. The exponent n can be kept fixed at its theoretical value 2.

2. Fit the upper part of the unloading curve with a power law function

$$F = \alpha(h - h_p)^m.$$

   using orthogonal least squares as implemented in the package ODRPACK95 [2]. The range should be approx. 85–98 % F_{max}. All three parameters are fitted.

3. The auxiliary parameter ε is calculated from the power m as in (6)

4. The slopes at the maximum depth are calculated as

$$S_{load} = n\frac{F_{max}}{h_{max} - h_0} \tag{23}$$

$$S_{unload} = m\frac{F_{max}}{h_{max} - h_p} \tag{24}$$

5. The contact area, indentation hardness and contact modulus are calculated as

$$A_p(h_c) = C_0 F_{max}^2 \left(\frac{2S_{unload} - \beta\varepsilon S_{load}}{S_{unload} S_{load}}\right)^2 \tag{25}$$

$$H_{IT} = \frac{1}{C_0 F_{max}} \left(\frac{S_{unload} S_{load}}{2S_{unload} - \beta\varepsilon S_{load}}\right)^2 \tag{26}$$

$$E_r = \frac{1}{2\beta F_{max}} \sqrt{\frac{\pi}{C_0}} \frac{S_{unload}^2 S_{load}}{2S_{unload} - \beta\varepsilon S_{load}} \tag{27}$$

   with C_0 the coefficient of the h^2 term in the area calibration function and β a geometric correction term. Currently, we use $\beta = 1$.

6. The indentation modulus can be calculated from (11)

12 F-h^2 analysis

The analysis of F vs h^2 follows [6, 7, 8].

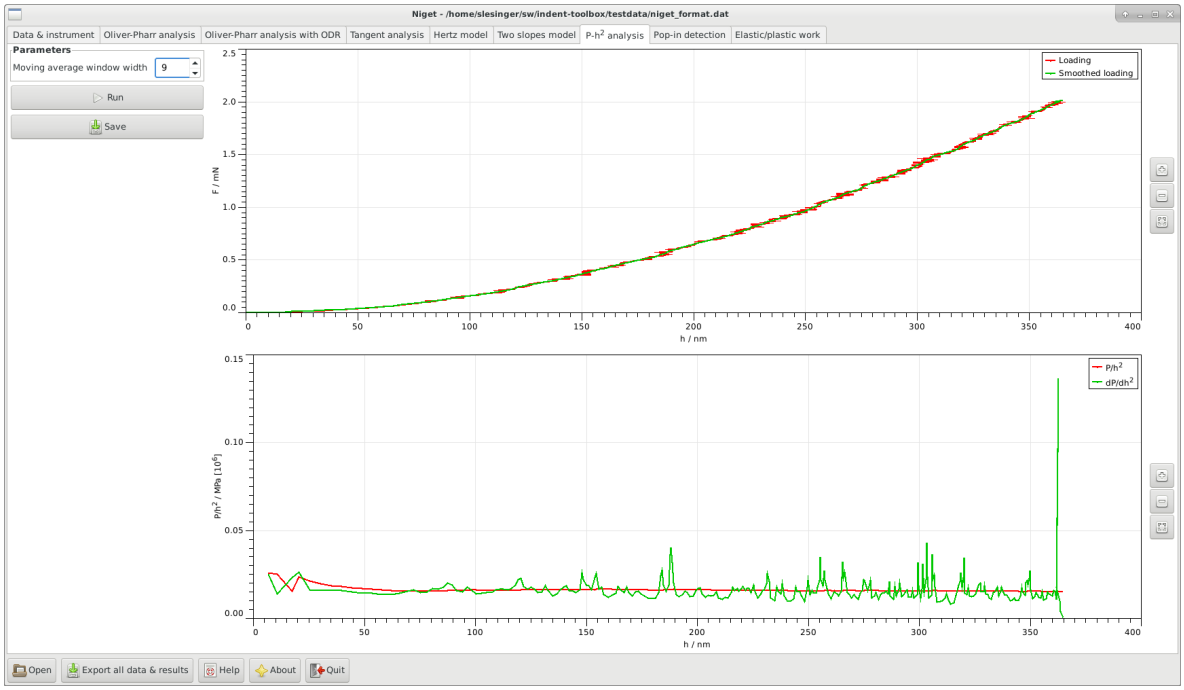

Figure 12: F vs h^2 analysis

12.1 Window

The window consists of several blocks:

- *Parameters* allows the user to set the width of the moving average window. The value 1 corresponds to no smoothing. This value is saved in settings.
- *Run* perform calculation and display curve, see section 12.2.
- *Save* save parameters and results to given file.
- *Graph*

  Top: display the loading curve and the smoothed curve.

  Bottom: display the F/h^2 and the $\mathrm{d}F/\mathrm{d}h^2$ curves.

  Stepwise zooming/unzooming can be performed by selecting a range with the mouse and pressing the *Zoom/ Unzoom* buttons. The graph is restored to its original size by the *Restore* button. Zooming in the two graphs is independent.

12.2 Procedure

1. We use a moving average with a fixed width and constant weight. This means we substitute a value with its average with s values to the left and to the right, $w = 2s + 1$

$$\hat{x}_i = \frac{1}{w} \sum_{j=-s}^{s} x_{i+j}.$$

   The value $w = 1$ corresponds to the original data. There is only one moving average type defined for both depth and load. Increasing the value of w noise becomes less influential but important small effects can get lost as well. Therefore, the value should not be too large, below 11 is recommended.

2. The ratio F/h^2 is calculated for for each data pair $(\hat{h}, \hat{F})$ of the (smoothed) loading curve and plotted as a function of the (smoothed) depth $\hat{h}$.

3. The derivative $\mathrm{d}F/\mathrm{d}h^2$ is calculated for for each data pair $(\hat{h}, \hat{F})$ of the (smoothed) loading curve and plotted as a function of the (smoothed) depth $\hat{h}$. The derivative is done numerically as the ratio of the derivatives of F and h with respect to the (time)step or index i

$$\frac{\mathrm{d}F}{\mathrm{d}h^2} = \frac{\mathrm{d}F}{\mathrm{d}i} \left(\frac{\mathrm{d}h^2}{\mathrm{d}i} \right)^{-1}.$$

   The numerical derivatives are calculated using the three-point formula for equally spaced data.

13 Pop-in detection

The pop-in effect is a reaction of the crystalline structure to load. Defects of the crystalline structure occur during loading, these then reveal themselves as discontinuous jumps in the depth at constant load. The critical load values are characteristic for the given orientation of the given material and can be compared to theoretical predictions based on the knowledge of the crystal lattice.

13.1 Window

The window consists of several blocks:

- *Info* displays the maximum depth and force during the indentation
- *Parameters* allows the user to set the following parameters and the selected range in nm.
  - *Moving average window width* the width of the moving average window. The value 1 corresponds to no smoothing.
  - *Derivative threshold for pop-in detection* the minimum derivative to identify the point as a pop-in.
  - *Derivative threshold for pop-in width* determines how far to extend the pop-in once it was identified.

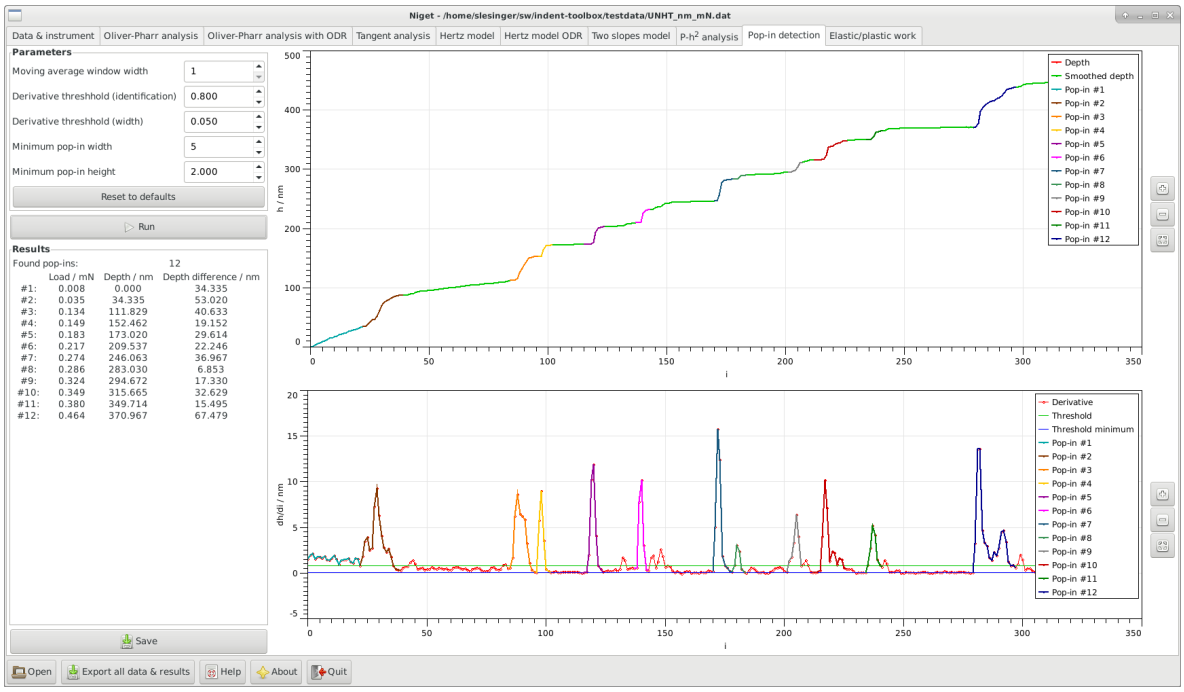

Figure 13: Pop-in detection

- *Minimum pop-in width* minimum number of datapoints needed for a pop-in.
- *Minimum pop-in height* (in nm) minimum jump in height needed for a pop-in.

These values are saved in settings. Default values are provided, but most likely the user will have to find proper values for each curve. For a detailed description see 13.2

- *Run* perform calculation and display curve, see section 13.2. *Results* displays the found pop-in events, the load and depth at which they occurred and the jump in the depth associated with it, see section 13.2.
- *Save* save parameters and results to given file.
- *Graph*

  Top: display the loading curve and the smoothed curve.

  Bottom: display the derivative of the depth with respect to the index (pseudo-time) together with the two derivative thresholds.

  Identified pop-ins are shown in color.
  Stepwise zooming/unzooming can be performed by selecting a range with the mouse and pressing the *Zoom/ Unzoom* buttons. The graph is restored to its original size by the *Restore* button. Zooming in the two graphs is independent.

13.2 Procedure

There is no standardized procedure how to define pop-in events. We use here a brute force direct method.

1. We use a moving average with a fixed width and constant weight. This means we substitute a value with its average with s values to the left and to the right, $w = 2s + 1$

$$\hat{x}_i = \frac{1}{w} \sum_{j=-s}^{s} x_{i+j}.$$

The value $w = 1$ corresponds to the original data. Increasing the value of w noise becomes less influential but important small effects can get lost as well. Therefore, the value should not be too large, below 11 is recommended.

2. Calculate the derivative of the loading curve with respect to the index (pseudo-time). This is the numerical derivative (in this case threee point derivative with equal steps)

$$dx_i = \frac{1}{2}(x_{i+1} - x_{i-1}),$$

where h is the step in the data.

3. Find the indices i for which x_i is larger then the *Derivative threshold for pop-in detection*

4. Group the indices found in the previous step into consecutive groups.

5. Enlarge each group of indices to the left and to the right to include all values larger than *Derivative threshold for pop-in width.*

6. For each group, check that the difference between the leftmost and righmost index is at least *Minimum pop-in width.*

7. For each group, check that the difference in depth between the leftmost and righmost point is at least *Minimum pop-in height.*

8. For each group, find the average load for the range between leftmost and rightmost, the depth at the leftmost point (beginning of the pop-in), the difference in height between the leftmost and rightmost point and the indices.

14 Elastic/plastic work

The elastic and plastic work can be calculated.

14.1 Window

The window consists of several blocks:

- *Parameters* allows the user to set the width of the moving average window. The value 1 corresponds to no smoothing. This value is saved in settings.
- *Run* perform calculation and display curve, see section 12.2.
- *Save* save parameters and results to given file.
- *Graph* display the (smoothed) indentation curve. Stepwise zooming/unzooming can be performed by selecting a range with the mouse and pressing the *Zoom/ Unzoom* buttons. The graph is restored to its original size by the *Restore* button.

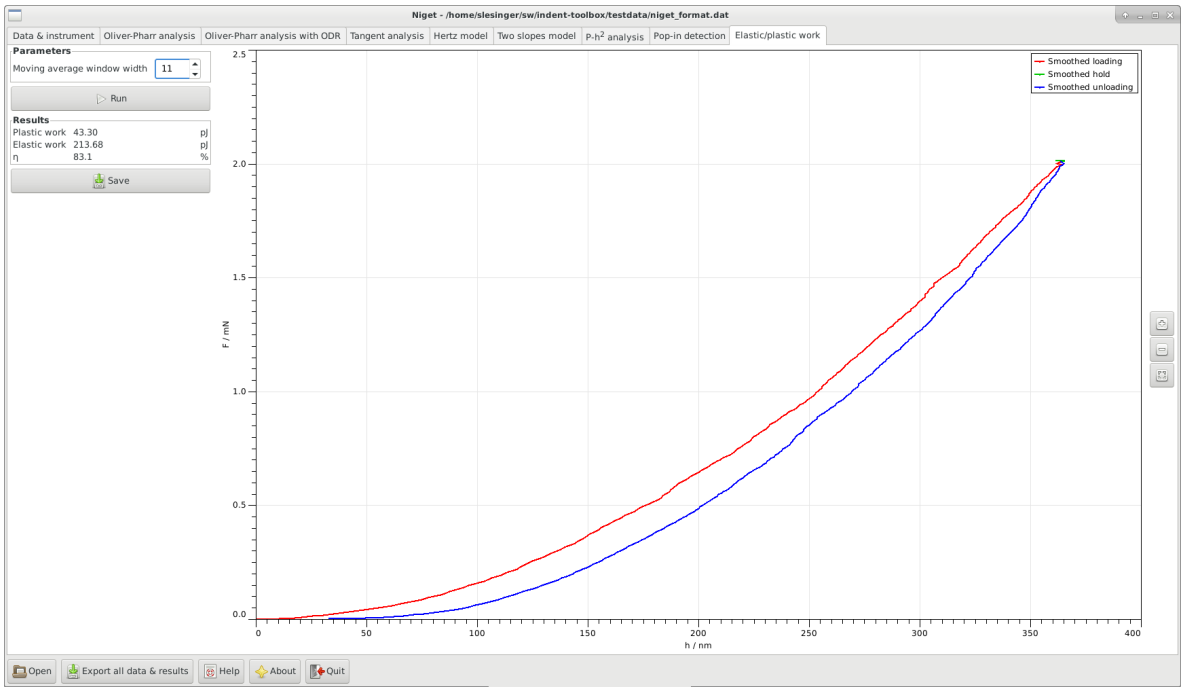

Figure 14: Elastic and plastic work calculation

14.2 Procedure

1. We use a moving average with a fixed width and constant weight. This means we substitute a value with its average with s values to the left and to the right, $w = 2s + 1$

$$\hat{x}_i = \frac{1}{w} \sum_{j=-s}^{s} x_{i+j}.$$

The value $w = 1$ corresponds to the original data. There is only one moving average type defined for both depth and load and for all three parts of the curve (load, unload, hold). Increasing the value of w noise becomes less influential but important small effects can get lost as well.

2. We use the simple trapezoidal rule for the integration of each curve

$$I = \sum_{i=2}^{n} \frac{1}{2}(y_i - y_{i-1})(x_i - x_{i-1}) \tag{28}$$

3. The elastic work is the area under the unloading curve, the plastic work is the area enclosed by the whole indentation curve. The energy ratio is the ratio of the elastic work and the total work

$$\begin{aligned} W_\mathrm{e} &= I_\mathrm{unload} \\ W_\mathrm{p} &= I_\mathrm{load} + I_\mathrm{hold} - I_\mathrm{unload} \\ \eta_\mathrm{IT} &= \frac{W_\mathrm{e}}{W_\mathrm{e} + W_\mathrm{p}} \end{aligned} \tag{29}$$

15 Uncertainties

Uncertainties can be calculated for the Oliver Pharr method (either fitting procedure), the tangent method and the Hertz method. The uncertainty sources are the noise in the depth and load, the uncertainty in the tip radius (Hertz method) and the uncertainties in the material parameters (Poisson's ratio, Young's modulus and Poisson's ratio of indenter). These are treated by the Gaussian propagation of uncertainties. The Monte Carlo method is used to treat the noise of depth load. In all cases we use a normal distribution and assume zero correlation (also between different depth values, i.e. $\rho(h_i, h_j) = 0$). The uncertainty of the contact point is demonstrated separately by explicitly performing the evaluation of data with different contact points and comparing them. The uncertainty of the choice of the fitting interval is not implemented yet.

15.1 Window

In all cases, pressing the *Uncertainties* button opens a separate window with the following blocks

- *Uncertainties in input values* the user can calculate the uncertainties and the individual contributions to the contact depth and area, indentation hardness and modulus and contact modulus. For the Hertz method only the uncertainties of the contact and indentation modulus are available. These contributions are calculated using the propagation of uncertainties [9]. Details are in sections 15.2.1, 15.2.2, 15.2.4.
- *Uncertainties due to choice of contact point* shows results, that would be obtained, if the contact point had been chosen differently. In many cases it is non-trivial how to choose the contact point, so this can be a significant contribution to the overall uncertainty.
- *Save* Save the resuls of the uncertainty analysis, including the main results of the corresponding main calculation, as the uncertainty analysis refers only to this calculation.
- *Monte Carlo calculation of uncertainties* set the number of iterations and launch the calculation of the uncertainties using the Monte Carlo method [10, 11]. In [10, 11] a minimum value of 10 000 is recommended, however, results obtained with smaller values can be used with proper care as a first guess. For ODR, one should start with approx. 100, and then gradually increase, as this is significantly more time consuming. The procedure is described in section 15.4.

Only one *Uncertainty* window may be open for each method. Values of uncertainties are saved in (and loaded from) the settings file for future use.

15.2 Gaussian propagation of uncertainties

The standard treatment to uncertainties is described in [9]. Here we use only the most important results.
Let two quantities Y_1 and Y_2 be estimated by y_1 and y_2 and depend on a set of uncorrelated variables $X_1, X_2, \ldots, X_N$. Let $u^2(x_a)$ be the estimated variance of the estimate x_a of X_a. Then the estimated variance associated with y_i is given by

$$u^2(y_i) = \sum_{a=1}^{N} \left(\frac{\partial Y_i}{\partial x_a} \right)^2 u^2(x_a) \tag{30}$$

and the estimated covariance associated with y_1 and y_2 is given by

$$u(y_i, y_j) = \sum_{a=1}^{N} \frac{\partial Y_i}{\partial x_a} \frac{\partial Y_j}{\partial x_a} u^2(x_a). \tag{31}$$

Let Z be estimated by z and depend on the correlated quantities $Y_i, i = 1, \ldots, M$. Let $u^2(y_i)$ be the estimated variance of the estimate y_i of Y_i and $u(y_i, y_j)$ the estimate of the covariance associated with y_i and y_j for $i \neq j$. Then the estimated variance associated with z is

$$u(z)^2 = \sum_{i=1}^{M} \left(\frac{\partial Z}{\partial y_i} \right)^2 u^2(y_i) + \sum_{i=1}^{M} \sum_{j=1, j\neq i}^{M} \frac{\partial Z}{\partial y_i} \frac{\partial Z}{\partial y_j} u(y_i, y_j) \tag{32}$$

The covariance of variables Z_s depending on the independent random variables X_i through intermediate variablse Y_a can be computed as.

$$\begin{aligned}
\operatorname{cov}(Z_s, Z_t) &= \sum_{a=1}^{n} \frac{\partial Z_s}{\partial x_a} \frac{\partial Z_t}{\partial x_a} u(x_a)^2 && (33)\\
&= \sum_{a=1}^{N} \sum_{i=1}^{M} \frac{\partial Z_s}{\partial y_i} \frac{\partial Y_i}{\partial x_a} \sum_{j=1}^{M} \frac{\partial Z_t}{\partial y_j} \frac{\partial Y_j}{\partial x_a} u(x_a)^2 && (34)\\
&= \sum_{i=1}^{M} \sum_{j=1}^{M} \frac{\partial Z_s}{\partial y_i} \frac{\partial Z_t}{\partial y_j} \operatorname{cov}(y_i, y_j) && (35)
\end{aligned}$$

In our case the independent variables X_a are the depth values h_i, load values F_i. We assume that they are all have a normal distribution function with the same variance, i.e.

$$\begin{aligned}
u(h_i) &= u(h), i = 1, \ldots, n\\
u(F_i) &= u(F), i = 1, \ldots, n
\end{aligned}$$

Equation (30) can then be written as the sum of two terms: the sum of the contributions from the depth data and from the force data

$$\begin{aligned}
u(y)^2 &= \sum_{i=1}^{n} \left(\frac{\partial Y}{\partial h_i} \right)^2 u^2(h_i) + \sum_{i=1}^{n} \left(\frac{\partial Y}{\partial h_i} \right)^2 u^2(h_i)\\
&= u(y; h)^2 + u(y; F)^2\\
u(y_a, y_b) &= \sum_{i=1}^{n} \frac{\partial Y_a}{\partial h_i} \frac{\partial Y_b}{\partial h_i} u^2(h_i) + \sum_{i=1}^{n} \frac{\partial Y_a}{\partial F_i} \frac{\partial Y_b}{\partial F_i} u^2(F_i)\\
&= \operatorname{cov}(y_a, y_b; h) + \operatorname{cov}(y_a, y_b; F).
\end{aligned}$$

For the indentation modulus we additionally need the tip radius R, Poisson's ratio ν, Young's modulus of the indenter E_i and Poisson's value of the indenter ν_i. We assume that they are independent and have a normal distribution.

15.2.1 Uncertainty propagation for Oliver Pharr method with ODR

The package ODRPACK does not allow to separate the contributions from depth and load uncertainty. The results which correspond to a fit using only the uncertainties in the depth, resp. load are shown instead of the contributions and the result from a fit using both uncertainties is shown instead of the combined uncertainty.

1. Obtain the uncertainties of m and h_p, as well as their covariance from ODRPACK, see A.1.

2. Calculate the uncertainty of ε from equation (6).
   For ε we find

$$u(\varepsilon;h) = u(m;h)\left[\frac{\varepsilon}{m} - \frac{\varepsilon - m}{m-1} - \frac{\varepsilon - m}{2(m-1)^2}(\psi_0(z) - \psi_0(w))\right], \tag{36}$$

   with $z = \frac{m}{2(m-1)}$, and $w = \frac{1}{2(m-1)}$ and similarly for $u(\varepsilon;F)$. The derivative $\mathrm{d}\varepsilon/\mathrm{d}m$ is needed in further calculations

$$\frac{\partial\varepsilon}{\partial m} = \frac{\varepsilon}{m} + \frac{\varepsilon - m}{m-1} - \frac{\varepsilon - m}{2(m-1)^2}(\psi_0(z) - \psi_0(w)) \tag{37}$$

3. For S we get

$$\begin{aligned} u(S;h)^2 &= \left(\frac{S}{m}\right)^2 u(m;h)^2 + \left(\frac{S}{h_\mathrm{max} - h_\mathrm{p}}\right)^2 u(h)^2 + \\ &\quad + \left(\frac{S}{h_\mathrm{max} - h_\mathrm{p}}\right)^2 u(h_\mathrm{p};h)^2 + \frac{2S^2}{m(h_\mathrm{max} - h_\mathrm{p})}\mathrm{cov}(m, h_\mathrm{p}; h) \\ u(S;F)^2 &= \left(\frac{S}{m}\right)^2 u(m;F)^2 + \left(\frac{S}{F_\mathrm{max}}\right)^2 u(F)^2 + \\ &\quad + \left(\frac{S}{h_\mathrm{max} - h_\mathrm{p}}\right)^2 u(h_\mathrm{p};F)^2 + \frac{2S^2}{m(h_\mathrm{max} - h_\mathrm{p})}\mathrm{cov}(m, h_\mathrm{p}; F) \end{aligned} \tag{38}$$

   For h_c we obtain

$$\begin{aligned} u(h_\mathrm{c};h)^2 &= (h_\mathrm{max} - h_\mathrm{p})^2\left(\frac{\varepsilon}{m^2} - \frac{1}{m}\frac{\partial\varepsilon}{\partial m}\right)^2 u(m;h)^2 + \left(1 - \frac{\varepsilon}{m}\right)^2 u(h_\mathrm{max};h)^2 + \\ &\quad + \left(\frac{\varepsilon}{m}\right)^2 u(h_\mathrm{p};h)^2 + 2\,(h_\mathrm{max} - h_\mathrm{p})\left(\frac{\varepsilon}{m^2} - \frac{1}{m}\frac{\partial\varepsilon}{\partial m}\right)\frac{\varepsilon}{m}\mathrm{cov}(m, h_\mathrm{p}; h) \end{aligned} \tag{39}$$

$$\begin{aligned} u(h_\mathrm{c};F)^2 &= (h_\mathrm{max} - h_\mathrm{p})^2\left(\frac{\varepsilon}{m^2} - \frac{1}{m}\frac{\partial\varepsilon}{\partial m}\right)^2 u(m;F)^2 + \left(\frac{\varepsilon}{m}\right)^2 u(h_\mathrm{p};F)^2 + \\ &\quad + 2\,(h_\mathrm{max} - h_\mathrm{p})\left(\frac{\varepsilon}{m^2} - \frac{1}{m}\frac{\partial\varepsilon}{\partial m}\right)\frac{\varepsilon}{m}\mathrm{cov}(m, h_\mathrm{p}; F) \end{aligned} \tag{40}$$

4. Calculate the uncertainties of $A_\mathrm{p}(h_\mathrm{c})$, H_IT and E_r from equations (1) to (10). The uncertainty of $A_\mathrm{p}(h_\mathrm{c})$ for a polynom (1) is simple

$$u(A_\mathrm{p}(h_\mathrm{c})) = \frac{\partial A_\mathrm{p}(h_\mathrm{c})}{\partial h_\mathrm{c}}u(h_\mathrm{c}) = \sum_k k c_k h_\mathrm{c}^{k-1} u(h_\mathrm{c}) \tag{41}$$

Uncertainties in the coefficients of the polynom are not taken into account.

5. The two contributions to the hardness are

$$u(H_{\mathrm{IT}};h) = \frac{H_{\mathrm{IT}}}{A_{\mathrm{p}}(h_{\mathrm{c}})}\frac{\partial A_{\mathrm{p}}}{\partial h_{\mathrm{c}}}u(h_{\mathrm{c}};h) \tag{42}$$

$$u(H_{\mathrm{IT}};F)^2 = \left(\frac{H_{\mathrm{IT}}}{F_{\max}}\right)^2 u(F)^2 + \left(\frac{H_{\mathrm{IT}}}{A_{\mathrm{p}}(h_{\mathrm{c}})}\right)^2 \left(\frac{\partial A_{\mathrm{p}}}{\partial h_{\mathrm{c}}}\right)^2 u(h_{\mathrm{c}};F)^2 \tag{43}$$

The two contributions to the contact modulus are

$$\begin{aligned} u(E_{\mathrm{r}};h)^2 = {} & \left(\frac{E_{\mathrm{r}}}{S}\right)^2 u(S;h)^2 + \left(\frac{E_{\mathrm{r}}}{2A_{\mathrm{p}}(h_{\mathrm{c}})}\right)^2 u(A_{\mathrm{p}}(h_{\mathrm{c}});h)^2 + \\ & -\frac{E_{\mathrm{r}}}{S}\frac{E_{\mathrm{r}}}{A_{\mathrm{p}}(h_{\mathrm{c}})}\frac{\partial A_{\mathrm{p}}}{\partial h_{\mathrm{c}}}\mathrm{cov}(S,h_{\mathrm{c}};h) \end{aligned} \tag{44}$$

$$\begin{aligned} u(E_{\mathrm{r}};F)^2 = {} & \left(\frac{E_{\mathrm{r}}}{S}\right)^2 u(S;F)^2 + \left(\frac{E_{\mathrm{r}}}{2A_{\mathrm{p}}(h_{\mathrm{c}})}\right)^2 u(A_{\mathrm{p}}(h_{\mathrm{c}});F)^2 + \\ & -\frac{E_{\mathrm{r}}}{S}\frac{E_{\mathrm{r}}}{A_{\mathrm{p}}(h_{\mathrm{c}})}\frac{\partial A_{\mathrm{p}}}{\partial h_{\mathrm{c}}}\mathrm{cov}(S,h_{\mathrm{c}};F) \end{aligned} \tag{45}$$

where the covariances $\mathrm{cov}(S,h_{\mathrm{c}};h)$ and $\mathrm{cov}(S,h_{\mathrm{c}};F)$ are

$$\begin{aligned} \mathrm{cov}(S,h_{\mathrm{c}};h) = {} & -\frac{S(1-\varepsilon)}{h_{\max}-h_{\mathrm{p}}}u(h)^2 - \frac{S\,(h_{\max}-h_{\mathrm{p}})}{m}\frac{\partial\varepsilon}{\partial m}u(m;h)^2 \\ & +\frac{S\varepsilon}{h_{\max}-h_{\mathrm{p}}}u(h_{\mathrm{p}};h)^2 + \\ & +S\left(\frac{\varepsilon}{m}-\frac{\partial\varepsilon}{\partial m}\right)\mathrm{cov}(h_{\mathrm{p}},m;h) \end{aligned} \tag{46}$$

$$\begin{aligned} \mathrm{cov}(S,h_{\mathrm{c}};F) = {} & -\frac{S}{m}\,(h_{\max}-h_{\mathrm{p}})\,\frac{\partial\varepsilon}{\partial m}u(m;F)^2 + \frac{S\varepsilon}{h_{\max}-h_{\mathrm{p}}}u(h_{\mathrm{p}},F)^2 \\ & +S\left(\frac{\varepsilon}{m}-\frac{\partial\varepsilon}{\partial m}\right)\mathrm{cov}(h_{\mathrm{p}},m;F) \end{aligned} \tag{47}$$

6. Combine the uncertainties originating in depth and load $u(E_{\mathrm{r}};h)$ and $u(E_{\mathrm{r}};F)$ with the uncertainties of the material parameters ν, ν_{i} and E_{i}. The contributions are

$$u(E_{\mathrm{IT}};\nu) = \frac{2\nu}{1-\nu^2}E_{\mathrm{IT}}u(\nu) \tag{48}$$

$$u(E_{\mathrm{IT}};\nu_{\mathrm{i}}) = \frac{2\nu_{\mathrm{i}}}{1-\nu^2}\frac{E_{\mathrm{IT}}^2}{E_{\mathrm{i}}}u(\nu_{\mathrm{i}}) \tag{49}$$

$$u(E_{\mathrm{IT}};E_{\mathrm{i}}) = \frac{(1-\nu_{\mathrm{i}}^2)}{(1-\nu)^2}\frac{E_{\mathrm{IT}}^2}{E_{\mathrm{i}}^2}u(E_{\mathrm{i}}) \tag{50}$$

$$u(E_{\mathrm{IT}};E_{\mathrm{r}}) = \frac{E_{\mathrm{IT}}^2}{E_{\mathrm{r}}^2(1-\nu^2)}\sqrt{u(E_{\mathrm{r}};h)^2+u(E_{\mathrm{r}};F)^2} \tag{51}$$

15.2.2 Uncertainty propagation for Hertz method with ODR

1. Calculate the uncertainty of the proportionality factor γ in (22), see section A.1
2. If the tip radius was given, the contributions to the uncertainty of the contact modulus are

$$\begin{aligned} u(E_\mathrm{r};h) &= \frac{E_\mathrm{r}}{\gamma}u(\gamma;h) \\ u(E_\mathrm{r};F) &= \frac{E_\mathrm{r}}{\gamma}u(\gamma;F) \\ u(E_\mathrm{r};R) &= \frac{E_\mathrm{r}}{2R}u(R) \end{aligned}$$

3. Calculate the uncertainty of E_IT as in step 6 in 15.2.1.
4. If the contact modulus was given the contributions to the uncertainty of the radius are

$$\begin{aligned} u(R;h) &= 2\frac{R}{\gamma}u(\gamma;h) \\ u(R;F) &= 2\frac{R}{\gamma}u(\gamma;F) \\ u(R;E_\mathrm{r}) &= 2\frac{R}{E_\mathrm{r}}u(E_\mathrm{r}) \end{aligned}$$

5. If the indentation modulus was given the uncertainties can be expressed in terms of the contact modulus

$$E_\mathrm{r} = \left[\frac{1-\nu^2}{E_\mathrm{IT}} + \frac{1-\nu_\mathrm{i}^2}{E_\mathrm{i}}\right]^{-1}$$

as

$$\begin{aligned} u(R;h) &= 2\frac{R}{\gamma}u(\gamma;h) \\ u(R;F) &= 2\frac{R}{\gamma}u(\gamma;F) \\ u(R;\nu) &= 4R\frac{\nu E_\mathrm{r}}{E_\mathrm{IT}}u(\nu) \\ u(R;\nu_\mathrm{i}) &= 4R\frac{\nu_\mathrm{i} E_\mathrm{r}}{E_\mathrm{i}}u(\nu_\mathrm{i}) \\ u(R;E_\mathrm{i}) &= 2\frac{E_\mathrm{r}}{E_\mathrm{i}^2}(1-\nu_\mathrm{i}^2)u(E_\mathrm{i}) \\ u(R;E_\mathrm{IT}) &= 2\frac{E_\mathrm{r}}{E_\mathrm{IT}^2}(1-\nu^2)u(E_\mathrm{IT}) \end{aligned}$$

15.2.3 Uncertainty propagation for stiffness analysis

The package ODRPACK does not allow to separate the contributions from depth and load uncertainty. The results which correspond to a fit using only the uncertainties in the depth, resp. load are shown instead of the contributions and the result from a fit using both uncertainties is shown instead of the combined uncertainty. The uncertainties of k_load, q_load, k_unload and q_unload are obtained directly from ODRPACK, see A.1

15.2.4 Uncertainty propagation for two slopes method with ODR

The package ODRPACK does not allow to separate the contributions from depth and load uncertainty. The results which correspond to a fit using only the uncertainties in the depth, resp. load are shown instead of the contributions and the result from a fit using both uncertainties is shown instead of the combined uncertainty. In the following we assume that neither fit includes the $(h_{\max}, F_{\max})$ point and that the fits are independent.

1. Obtain the uncertainties of n, h_0, m, h_p, and their covariance from ODRPACK, see A.1.

2. Same as step 2 in section 15.2.1.

3. The unloading slope S_unload is the same as in step 3 in section 15.2.1. The uncertainty of the loading slope is analogous to (38)

$$\begin{aligned} u(S_\mathrm{load};h)^2 &= \left(\frac{S_\mathrm{load}}{n}\right)^2 u(n;h)^2 + \left(\frac{S_\mathrm{load}}{h_{\max}-h_0}\right)^2 u(h)^2 + \\ &\quad + \left(\frac{S_\mathrm{load}}{h_{\max}-h_0}\right)^2 u(h_0;h)^2 + \frac{2S_\mathrm{load}^2}{n(h_{\max}-h_0)}\mathrm{cov}(n,h_0;h) \\ u(S_\mathrm{load};F)^2 &= \left(\frac{S_\mathrm{load}}{n}\right)^2 u(n;F)^2 + \left(\frac{S_\mathrm{load}}{F_{\max}}\right)^2 u(F)^2 + \\ &\quad + \left(\frac{S_\mathrm{load}}{h_{\max}-h_0}\right)^2 u(h_0;F)^2 + \frac{2S_\mathrm{load}^2}{n(h_{\max}-h_0)}\mathrm{cov}(n,h_0;F) \end{aligned} \tag{52}$$

   Furthermore we will need the covariance of S_unload and ε

$$\begin{aligned} \mathrm{cov}(S_\mathrm{unload},\varepsilon;h) &= \frac{\partial\varepsilon}{\partial m}\mathrm{cov}(S_\mathrm{unload},m;h) \\ &= \frac{\partial\varepsilon}{\partial m}\left[m\frac{F_{\max}}{(h_{\max}-h_\mathrm{p})^2}\mathrm{cov}(h_\mathrm{p},m;h) + \frac{F_{\max}}{h_{\max}-h_\mathrm{p}}u(m;h)^2\right] \\ \mathrm{cov}(S_\mathrm{unload},\varepsilon;F) &= \frac{\partial\varepsilon}{\partial m}\mathrm{cov}(S_\mathrm{unload},m;F) \\ &= \frac{\partial\varepsilon}{\partial m}\left[m\frac{F_{\max}}{(h_{\max}-h_\mathrm{p})^2}\mathrm{cov}(h_\mathrm{p},m;F) + \frac{F_{\max}}{h_{\max}-h_\mathrm{p}}u(m;F)^2\right] \end{aligned} \tag{53}$$

4. It is practical to calculate the uncertainties of the auxiliary term K given as

$$K = \frac{2S_\mathrm{unload} - \beta\varepsilon S_\mathrm{load}}{S_\mathrm{unload}S_\mathrm{load}} = \frac{2}{S_\mathrm{load}} - \frac{\beta\varepsilon}{S_\mathrm{unload}} \tag{54}$$

These are

$$\begin{aligned} u(K;h)^2 &= \left(\frac{-2}{S_{\text{load}}^2}\right)^2 u(S_{\text{load}};h)^2 + \left(\frac{\beta\varepsilon}{S_{\text{unload}}^2}\right)^2 u(S_{\text{unload}};h)^2 + \left(\frac{\beta}{S_{\text{unload}}}\right)^2 u(\varepsilon;h)^2 \\ &\quad + \frac{\beta\varepsilon}{S_{\text{unload}}^2}\text{cov}(S_{\text{unload}},\varepsilon;h) \\ u(K;F)^2 &= \left(\frac{-2}{S_{\text{load}}^2}\right)^2 u(S_{\text{load}};F)^2 + \left(\frac{\beta\varepsilon}{S_{\text{unload}}^2}\right)^2 u(S_{\text{unload}};F)^2 + \left(\frac{\beta}{S_{\text{unload}}}\right)^2 u(\varepsilon;F)^2 \\ &\quad + \frac{\beta\varepsilon}{S_{\text{unload}}^2}\text{cov}(S_{\text{unload}},\varepsilon;F) \end{aligned} \tag{55}$$

We will also need the covariance

$$\begin{aligned} \text{cov}(S_{\text{unload}},K;h) &= \frac{\beta\varepsilon}{S_{\text{unload}}^2}u(S_{\text{unload}};h)^2 + \frac{-\beta}{S_{\text{unload}}}\text{cov}(S_{\text{unload}},\varepsilon;h) \\ \text{cov}(S_{\text{unload}},K;F) &= \frac{\beta\varepsilon}{S_{\text{unload}}^2}u(S_{\text{unload}};F)^2 + \frac{-\beta}{S_{\text{unload}}}\text{cov}(S_{\text{unload}},\varepsilon;F) \end{aligned} \tag{56}$$

5. In terms of K the uncertainties of the contact area, hardness and contact modulus can be expressed as

$$\begin{aligned} u(A;h) &= A\frac{u(K;h)}{K} \\ u(A;F)^2 &= \left(2A\frac{u(K;h)}{K}\right)^2 + \left(2A\frac{u(F_{\text{max}})}{F_{\text{max}}}\right)^2 \end{aligned} \tag{57}$$

$$\begin{aligned} u(H_{\text{IT}};h) &= 2A\frac{u(K;h)}{K} \\ u(H_{\text{IT}};F)^2 &= \left(2A\frac{u(K;F)}{K}\right)^2 + \left(A\frac{u(F_{\text{max}})}{F_{\text{max}}}\right)^2 \end{aligned} \tag{58}$$

and

$$\begin{aligned} u(E_{\text{r}};h)^2 &= \left(E_{\text{r}}\frac{u(S_{\text{unload}};h)}{S_{\text{unload}}}\right)^2 + \left(E_{\text{r}}\frac{u(K;h)}{K}\right)^2 - E_{\text{r}}^2\frac{\text{cov}(S_{\text{unload}},K;h)}{KS_{\text{unload}}} \\ u(E_{\text{r}};F)^2 &= \left(E_{\text{r}}\frac{u(S_{\text{unload}};F)}{S_{\text{unload}}}\right)^2 + \left(E_{\text{r}}\frac{u(K;F)}{K}\right)^2 + \left(E_{\text{r}}\frac{u(F)}{F_{\text{max}}}\right)^2 \\ &\quad - E_{\text{r}}^2\frac{\text{cov}(S_{\text{unload}},K;F)}{KS_{\text{unload}}} \end{aligned} \tag{59}$$

6. Same as step 6 in section 15.2.1.

15.3 Uncertainty due to choice of contact point

It may not be always clear where exactly the contact between the indenter and the sample occurs. This induces an uncertainty of type B which must be estimated. In order to facilitate this, we explicitly show how the results change when the contact point was chosen to be more to the left or to the right by a certain amount of points. Zero contact point shift means the original contact point was used. A positive (negative) contact point N shift means the Nth neighboring point to the right (left) was used instead. This corresponds to a shift by $(\Delta h, \Delta F)$ in the unloading data; data are added or removed from the loading data as well as shifted by $(\Delta h, \Delta F)$. The ranges for the interval of the fitting procedures are transformed. If they were chosen in the length regime, either by mouse or by input in the entries, they are shifted by δh. Percentages of the maximum forces are not transformed, since the maximum force is already shifted.

15.4 Monte Carlo calculation of uncertainties

The calculation is run by clicking the *Monte Carlo* button in the Uncertainty window. After finishing a window containg the results is shown. The calculation itself is described in 15.4.2. A more detailed description of the use of the Monte Carlo method for the evaluation of uncertainties is in [10, 11].

15.4.1 Window

- *Input of Monte Carlo simulation* shows the input values with which the calculation was run.
- *Results of Monte Carlo simulation* shows the mean and standard deviation calculated from the resulting PDFs. A histogram can be shown by clicking on the histogram button for each variable.
- *Save* save the results of this Monte Carlo simulation. This includes the mean, standard deviation, histogram and full data for each output variable.

Only one *Monte Carlo calculation* window may be open for each method. It is closed when the Uncertainty window is closed and results are discarded.

15.4.2 Monte Carlo calculation of uncertainties: Description of method

A more detailed description of the use of the Monte Carlo method for the evaluation of uncertainties is in [10, 11]. The procedure is based on the propagation of probability distribution functions (PDF) of the input variables and obtaining the PDF of the output variable. The input variables are varied according to a given PDF and the output variables are calculated in each case. For a large enough number of trials we obtain a PDF of the output variables, which can be further analyzed. Here we calculate only the mean and the standard deviation and construct a simple histogram. This method is very well suited for complicated measurement models, especially non-linear models or non-Gaussian PDFs of the input variables. In simple cases, it gives the same results as the Gaussian law of propagation. Two Monte Carlo method has two significant drawbacks compared to the Gaussian law of propagation: Firstly, it can become very time consuming, especially when several input variables are present. Secondly,

it is impossible to separate the uncertainty contributions from different input variables. Different calculations must be made, thus increasing even more the time needed. Therefore, it is currently used only for the uncertainty in the depth and the load. The uncertainties in the material parameters and the tip radius (Hertz model) can be described sufficiently by the Gaussian law. The model uses independent, normal PDFs with constant variance for all depth and load values. Only values contained in the fitting range of the main calculation are considered, i.e., the fitting range is not determined for every individual calculation. This leads to a significant speed up. For well-behaved data, this should not cause any significant errors.

References

[1] W. C. Oliver and G. M. Pharr. An improved technique for determining hardness and elastic-modulus using load and displacement sensing indentation experiments. *J. Mater. Res.*, 7:1564 – 1583, 1992.

[2] Jason W. Zwolak, Paul T. Boggs, and Layne T. Watson. Algorithm 869: ODRPACK95: A weighted orthogonal distance regression code with bound constraints. *ACM Trans. Math. Softw.*, 33(4), August 2007.

[3] M.F. Doerner and W.D. Nix. A method for interpreting the data from depth-sensing indentation instruments. *Journal of Materials Research*, 1(4):601–609, 008 1986.

[4] Heinrich Hertz. Ueber die Berührung fester elastischer Körper. *Journal für die reine und angewandte Mathematik*, 1882(92):156–171, 1882.

[5] W. C. Oliver. Alternative technique for analyzing instrumented indentation data. *Journal of Materials Research*, 16(11):3202–3206, 2001.

[6] M. R. McGurk and T. F. Page. Using the P-δ^2 analysis to deconvolute the nanoindentation response of hard-coated systems. *Journal of Materials Research*, 14(6):2283–2295, 006 1999.

[7] J. Malzbender and G. de With. The use of the indentation loading curve to detect fracture of coatings. *Surface and Coatings Technology*, 137(1):72 – 76, 2001.

[8] J. Malzbender and G. de With. The use of the loading curve to assess soft coatings. *Surface and Coatings Technology*, 127(2–3):265 – 272, 2000.

[9] ISO/IEC. Uncertainty of measurement – part 3: Guide to the expression of uncertainty in measurement (GUM:1995). ISO/IEC 98–3:2008, International Organization for Standardization, Geneva, Switzerland, 2008.

[10] ISO/IEC. Propagation of distributions using a Monte Carlo method. ISO/IEC 98–3:2008/Suppl 1:2008, International Organization for Standardization, Geneva, Switzerland, 2008.

[11] ISO/IEC. Extension to any number of output quantities. ISO/IEC 98–3:2008/Suppl 2:2011, International Organization for Standardization, Geneva, Switzerland, 2011.

[12] Paul T. Boggs, Janet R. Donaldson, Richaard h. Byrd, and Robert B. Schnabel. Algorithm 676: Odrpack: Software for weighted orthogonal distance regression. *ACM Trans. Math. Softw.*, 15(4):348–364, December 1989.

[13] P T Boggs and J E Rogers. The computation and use of the asymptotic covariance matrix for measurement error models. Internal Report 89-4102, National Institute of Standards and Technology, Gaithersburg, MD, 1990.

A Data fitting

We use types of data fitting: the Deming fit for straight lines, the least squares fit of the 3/2 power and orthogonal distance regression [12] for power law functions.

A.1 Orthogonal distance regression

Orthogonal distance regression, also called generalized least squares regression, errors-in-variables models or measurement error models, attempts to tries to find the best fit taking into account errors in both x- and y- values. Assuming the relationship

$$y^* = f(x^*; \beta) \tag{60}$$

where β are parameters and x^* and y^* are the "true" values, without error, this leads to a minimization of the sum

$$\min_{\beta,\delta} \sum_{i=1}^{n} \left[(y_i - f(x_i + \delta; \beta))^2 + \delta_i^2 \right] \tag{61}$$

which can be interpreted as the sum of orthogonal distances from the data points (x_i, y_i) to the curve $y = f(x, \beta)$. It can be rewritten as

$$\min_{\beta,\delta,\varepsilon} \sum_{i=1}^{n} \left[\varepsilon_i^2 + \delta_i^2 \right] \tag{62}$$

subject to

$$y_i + \varepsilon_i = f(x_i + \delta_i; \beta). \tag{63}$$

This can be generalized to accomodate different weights for the datapoints and to higher dimensions

$$\min_{\beta,\delta,\varepsilon} \sum_{i=1}^{n} \left[\varepsilon_i^T w_\varepsilon^2 \varepsilon_i + \delta_i^T w_\delta^2 \delta_i \right],$$

where ε and δ are m and n dimensional vectors and w_ε and w_δ are symmetric, positive diagonal matrices. Usually the inverse uncertainties of the data points are chosen as weights. We use the implementation ODRPACK [12].

There are different estimates of the covariance matrix of the fitted parameters β. Most of them are based on the linearization method which assumes that the nonlinear function can be adequately approximated at the solution by a linear model. Here, we use an approximation where the covariance matrix associated with the parameter estimates is based $\left(J^T J\right)^{-1}$, where J is the Jacobian matrix of the x and y residuals, weighted by the triangular matrix of

the Cholesky factorization of the covariance matrix associated with the experimental data. ODRPACK uses the following implementation [13]

$$\hat{V} = \hat{\sigma}^2 \left[\sum_{i=1}^{n} \frac{\partial f(x_i + \delta_i; \beta)}{\partial \beta^T} w_{\varepsilon_i}^2 \frac{\partial f(x_i + \delta_i; \beta)}{\partial \beta} + \frac{\partial f(x_i + \delta_i; \beta)}{\partial \delta^T} w_{\delta_i}^2 \frac{\partial f(x_i + \delta_i; \beta)}{\partial \delta} \right] \tag{64}$$

The residual variance $\hat{\sigma}^2$ is estimated as

$$\hat{\sigma}^2 = \frac{1}{n-p} \sum_{i=1}^{n} \left[(y_i - f(x_i + \delta; \beta))^T w_{\varepsilon_i}^2 (y_i - f(x_i + \delta; \beta)) + \delta_i^T w_{\delta_i}^2 \delta_i \right] \tag{65}$$

where $\beta \in \mathbb{R}^p$ and $\delta_i \in \mathbb{R}^m,\ i = 1, \ldots, n$ are the optimized parameters,

A.2 Total least squares - Deming fit

The Deming fit is a special case of orthogonal regression which can be solved analytically. It seeks the best fit to a linear relationship between the x- and y-values

$$y^* = ax^* + b, \tag{66}$$

by minimizing the weighted sum of (orthogonal) distances of datapoints from the curve

$$S = \sum_{i=1}^{n} \frac{1}{\sigma_\epsilon^2} (y_i - ax_i^* - b)^2 + \frac{1}{\sigma_\eta^2} (x_i - x_i^*)^2,$$

with respect to the parameters a, b, and x_i^*. The weights are the variances of the errors in the x-variable (σ_η^2) and the y-variable (σ_ϵ^2). It is not necessary to know the variances themselves, it is sufficient to know their ratio

$$\delta = \frac{\sigma_\epsilon^2}{\sigma_\eta^2}. \tag{67}$$

The solution is

$$a = \frac{1}{2s_{xy}} \left[s_{yy} - \delta s_{xx} \pm \sqrt{(s_{yy} - \delta s_{xx})^2 + 4\delta s_{xy}^2} \right] \tag{68}$$

$$b = \bar{y} - a\bar{x} \tag{69}$$

$$x_i^* = x_i + \frac{a}{\delta + a^2} (y_i - b - ax_i), \tag{70}$$

where

$$\bar{x} = \frac{1}{n} \sum_{i=1}^{n} x_i \tag{71}$$

$$\bar{y} = \frac{1}{n} \sum_{i=1}^{n} y_i \tag{72}$$

$$s_{xx} = \frac{1}{n} \sum_{i=1}^{n} (x_i - \bar{x})^2 \tag{73}$$

$$s_{yy} = \frac{1}{n} \sum_{i=1}^{n} (y_i - \bar{y})^2 \tag{74}$$

$$s_{xy} = \frac{1}{n} \sum_{i=1}^{n} (x_i - \bar{x})(y_i - \bar{y}). \tag{75}$$

A.3 Least squares - 3/2 power fit

We seek the best fit

$$y = ax^{3/2} + b, \tag{76}$$

by minimizing the sum of (vertical) distances of datapoints from the curve

$$S = \sum_{i=1}^{n} (y_i - ax_i^{3/2} - b)^2,$$

with respect to the parameters a, b. The solution is

$$a = \frac{\overline{x^{3/2}y} - \overline{x^{3/2}}\,\bar{y}}{\overline{x^3} - \left(\overline{x^{3/2}}\right)^2} \tag{77}$$

$$b = \bar{y} - a\overline{x^{3/2}} \tag{78}$$

where

$$\overline{x^{3/2}y} = \frac{1}{n}\sum_{i=1}^{n} x_i^{3/2} y_i \tag{79}$$

$$\overline{x^{3/2}} = \frac{1}{n}\sum_{i=1}^{n} x_i^{3/2} \tag{80}$$

$$\overline{x^3} = \frac{1}{n}\sum_{i=1}^{n} x_i^3 \tag{81}$$

$$\bar{y} = \frac{1}{n}\sum_{i=1}^{n} y_i \tag{82}$$