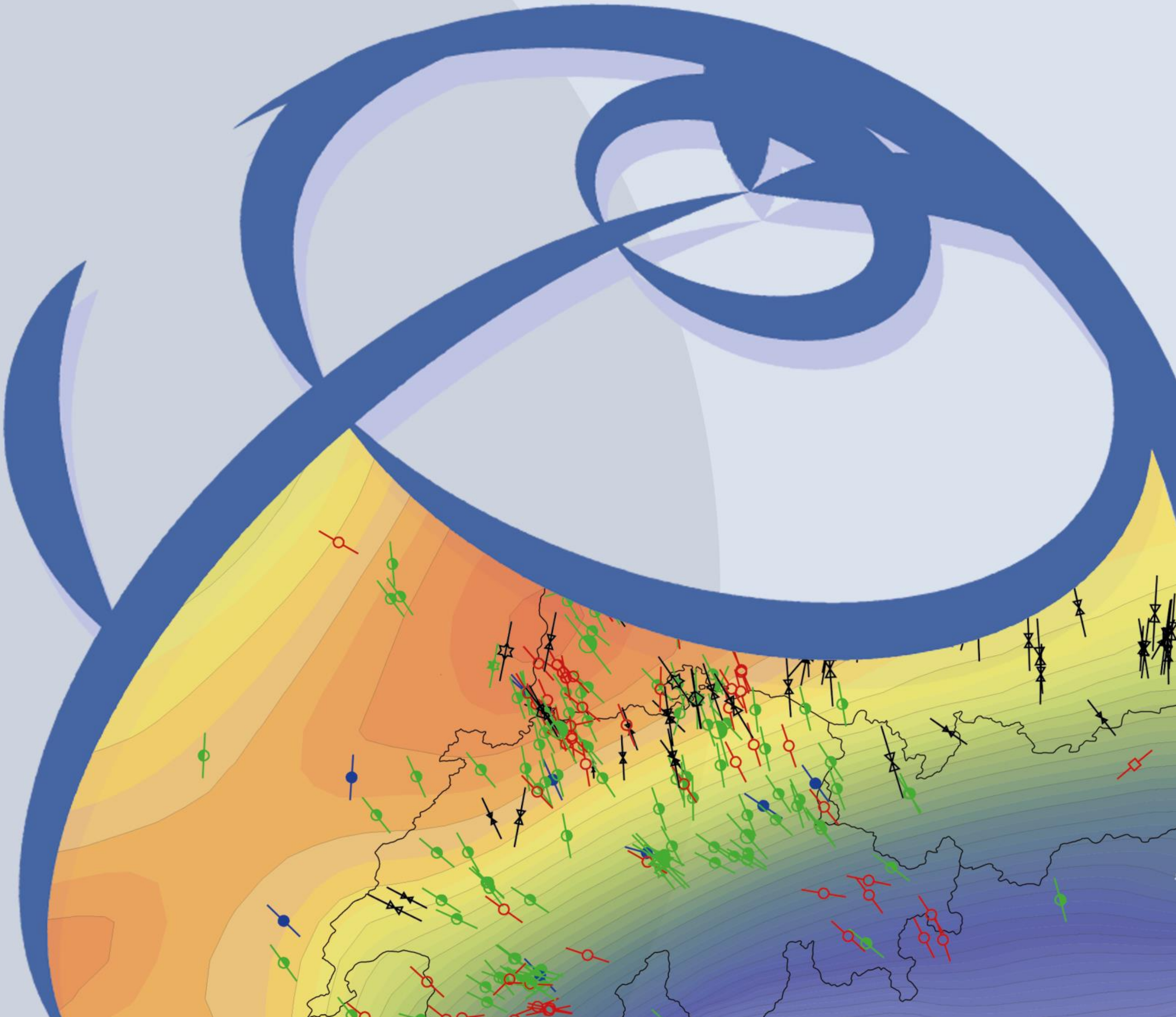
WSM WORLD STRESS MAP

WSM Technical Report 18-01

Manual of the Matlab Script FAST Calibration v1.0

Moritz Ziegler



Recommended citation of the report:

Ziegler, M. O. (2018). Manual of the Matlab Script FAST Calibration v1.0. World Stress Map Technical Report 18-01, GFZ German Research Centre for Geosciences.
DOI: http://doi.org/10.2312/wsm.2018.003

The software described in this report, including data examples, is published as:

Ziegler, M. O. (2018). Matlab Script FAST Calibration v1.0. GFZ Data Services.
DOI: http://doi.org/10.5880/wsm.2018.003

The software is also available for download on GitHub:

http://github.com/MorZieg/FAST_Calibration

Imprint

World Stress Map Project
GFZ German Research Centre for Geosciences

Telegrafenberg
D-14473 Potsdam
Published in Potsdam, Germany
September 2018
http://doi.org/10.2312/wsm.2018.003

WSM Technical Report 18-01

Manual of the Matlab Script FAST Calibration v1.0

Moritz Ziegler
GFZ German Research Centre for Geosciences, Potsdam, Germany

Helmholtz Centre
POTSDAM

Table of Contents

List of Tables

List of Figures

Abstract

The 3D geomechanical-numerical modelling of the in-situ stress state requires observed stress information at reference locations within the model area to be compared to the modelled stress state. This comparison of stress states and the ensuing adaptation of the displacement boundary conditions provide a best fit stress state in the entire model region that is based on the available stress information. This process is also referred to as calibration. Depending on the amount of available information and the complexity of the model the calibration is a lengthy process of trial-and-error modelling and analysis.

The Fast Automatic Stress Tensor Calibration (FAST Calibration) is a method and a Matlab script that facilitates and speeds up the calibration process. The method requires only three model scenarios with different boundary conditions. The modelled stress states at the locations of the observed stress state are extracted. Then they are used to compute the displacement boundary conditions that are required in order to achieve the best fit of the modelled to the observed stress state. Furthermore, the influence of the individual observed stress information on the resulting stress state can be weighted.

The script files are provided for Download at http://github.com/MorZieg/FAST_Calibration. Table 0-1 gives an overview of the folder structure and input files with a short explanation.

Tab. 0-1 Structure of the GitHub repository.

The folders and files in the GitHub repository http://github.com/MorZieg/FAST_Calibration. The page number directs to the documentation in this manual (if available).

File Name	Explanation	Page
root_calibration.m	Matlab script for the calibration of a geomechanical-numerical model on stress data records.	8
branch_calibration.m	Matlab script for the calibration of a geomechanical-numerical model on the stress state from another model.	11
CITATION.bib	The recommended citation for the software.	
LICENSE	The full GPL v3.0 license text.	
README.md	Readme file that contains relevant information on the usage of the software.	
calib_functions/	Folder that contains additionally required functions (Tab. 0-2).	
Examples/	Folder that contains example files for a calibration and multi-stage approach (Tab. 0-3).	

Tab. 0-2 Content of the folder "calib_functions".

Short explanation of the additionally required functions. The page number directs to the documentation in this manual (if available).

File Name	Explanation	Page
write_macro.m	Matlab function that writes a Tecplot 360 EX macro.	15
nodes2calibrationpoints.m	Matlab function that controls the calibration points in a branch calibration approach.	17

Tab. 0-3 Structure of the folder “examples”.

Short explanation of the files provided for an exemplified calibration and multi-stage approach.

File Name	Explanation
root_test.geom	Geometry of the geomechanical-numerical root model.
root_initial.inp root_initial.odb	Initial lithostatic stress state for the root model. Abaqus input file and output database.
root_calibration.inp root_calibration.odb	Three model scenarios with different boundary conditions. Abaqus input file (.inp) and output database (.odb).
root_final.inp root_final.odb	Calibrated root model. Abaqus input file (.inp) and output database (.odb).
root_nodes.csv	Nodes section from the root_test.geom input file.
branch_test.geom	Geometry of the geomechanical-numerical branch model.
branch_initial.inp branch_initial.odb	Initial lithostatic stress state for the branch model. Abaqus input file and output database.
branch_calibration.inp branch_calibration.odb	Three model scenarios with different boundary conditions. Abaqus input file (.inp) and output database (.odb).
branch_final.inp branch_final.odb	Calibrated branch model. Abaqus input file (.inp) and output database (.odb).

1 Introduction

The knowledge of the contemporary stress state in a rock volume is required for many geotechnical and scientific applications such as hydrocarbon and geothermal reservoir management, underground storage facilities for gas or waste material, and tunnelling, amongst others (Cornet, 2015). However, the available orientation and magnitude data on the stress state is quite sparse and in general not sufficient for a robust assessment of the local stress state (Heidbach, et al., 2010). Therefore, 3D geomechanical-numerical modelling is used to assess the stress state in a confined area. Such a modelling approach is based on the knowledge of the subsurface structure (geological layers, faults) and rock properties as well as on model-independent stress information within the model volume. The implementation of the stress state can be achieved by Dirichlet (displacement) boundary conditions which are altered until the observed stress components at discrete points are fitted. Herein, this process is referred to as model calibration. Once the model is calibrated the assumption is made that the stress state in the rest of the model is also a legitimate representation of the in-situ stress state.

Since stress data are required to be within the model area the size of the model depends on the availability and location of the stress information. If the density of stress information in the model area is high only a small model area is needed. However, if the stress information density is low, the size of the model increases to include sufficient stress information points for a successful model calibration. Usually the latter is the case and only little stress information is available. This calls for a large model area potentially in addition to a fine resolution that is needed to represent e.g. thin layers, complex structures, injection and production points or even boreholes and drillpaths. Such a model can easily exceed tens of millions of hexahedral finite elements. This is manageable for a single best fit model but in order to quantify the model uncertainties the full parameter space has to be explored. Then, the large number of finite elements in each individual scenario becomes a limiting factor. To ensure a reasonable computation time, gradients in element size or nesting schemes are required (Fig. 1-1a,b). Yet, these approaches require a large effort in the process of geometry preparation and discretization.

Ziegler et al. (2016) present the multi-stage approach that uses two differently sized models in order to have both a high resolution and sufficient data records for calibration while the computation time and physical size of the model remains acceptable. The multi-stage approach is beneficiary in terms of the simplified mesh generation and faster computation time but also since the large scale model provides the stress state in a wide area that can be used for several small scale reservoir models (Fig. 1-1c).

The FAST-Calibration (Fast Automatic Stress Tensor Calibration) is a Matlab tool that controls the statistical calibration of a 3D geomechanical-numerical model of the stress state following the approach described by Reiter and Heidbach (2014), Hergert et al. (2015), and Ziegler et al. (2016). It is mainly designed to support the multi-stage modelling procedure presented by Ziegler et al. (2016). However, it can also be used for the calibration of a single-stage model. The tools run in Matlab 2017a and higher and are meant to work with the visualization software Tecplot 360 EX 2015 R2 and higher (https://www.tecplot.com/products/tecplot-360/) in conjunction with the Tecplot 360 Add-on GeoStress (Stromeyer and Heidbach, 2017a,b). The user should be familiar with 3D geomechanical-numerical modelling, Matlab, Tecplot 360 EX, including a basic knowledge of Tecplot 360 EX macro functions, and the Tecplot 360 EX Add-on GeoStress. This FAST Calibration manual provides an overview of the scripts and is designed to help the user to adapt the scripts for their own needs.

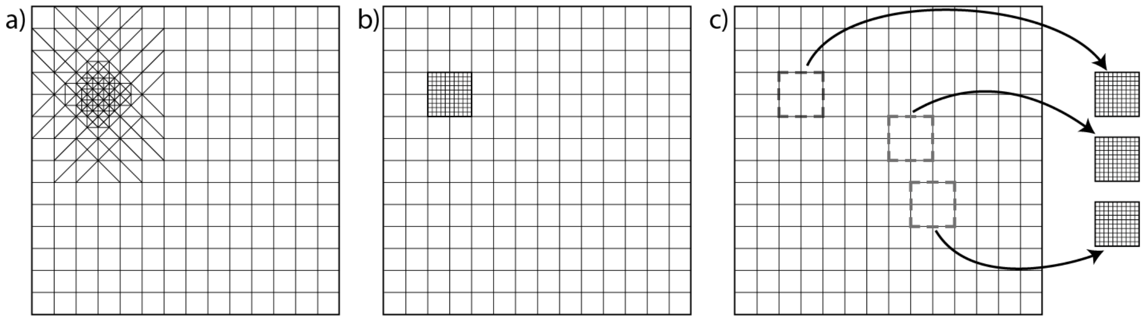


Fig. 1-1 Embedded, nesting and multistage approach.

Different approaches to cope with a large scale model area with only a small area with a high resolution. a) A gradient in element size. b) A nested modelling approach. c) The multi-stage approach that uses two or more models.

The main challenge in combining the visualization software Tecplot 360 EX with the computational power of Matlab is the exchange of data between the applications. Matlab requires the modelled stress state at discrete points in order to compare the modelled stress state with observed data records. This is achieved by a macro that is written by Matlab and contains the exact coordinates of the observed reference stress state. Once the macro is executed in Tecplot 360 EX the modelled stress state at the specified locations is written into a datafile. This datafile is then imported to Matlab which processes the contained data.

The FAST Calibration is divided into two scripts. The first controls the calibration of 3D geomechanical-numerical models that use the Finite Element method on observed stress information. The second script controls the calibration of 3D geomechanical-numerical models using the stress state retrieved from a larger stress model following the multi-stage procedure described by Ziegler et al. (2016). With these two scripts the multi-stage stress modelling approach is sped up and simplified. As well the calibration of a single stress model is facilitated. In the following the technical basics of 3D geomechanical-numerical modelling and the FAST-Calibration Matlab scripts are presented and their functionality is discussed.

2 Technical basics of stress modelling

The modelling of the in-situ stress state is based on knowledge of (1) the geometry of the geologic units and faults, (2) the rock properties, and (3) the stress information at individual points within the model volume. The 3D geological model of the subsurface is the base for the geomechanical model. It contains the topography and the geomechanically relevant lithological units. A discretisation for the finite element method is performed on the continuous geologic model. The discretized geometry is populated with material properties of the according rock units; the density, the Young's module, and the Poisson ratio.

Pointwise information on the orientation of the reduced stress tensor (the orientation of S_{Hmax}) are provided by the World Stress Map (WSM) (Heidbach et al., 2016) and information on the magnitudes of principal (horizontal) stresses can be accessed from the WSM or other scientific publications and are derived from, e.g., overcoring or hydraulic fracturing experiments. In most cases only the orientation of maximum horizontal stress S_{Hmax}, provided by the World Stress Map database (Heidbach et al., 2016), is available. Assuming that the vertical stress S_V is a principal stress, the S_{Hmax} orientation determines the orientation of the stress tensor. However, for the model calibration the stress tensor orientation is essential, but not sufficient. Besides the magnitude of S_V which is easy to fit when the density of the model is known, magnitudes of S_{Hmax} and in S_{hmin} are needed for the model calibration.

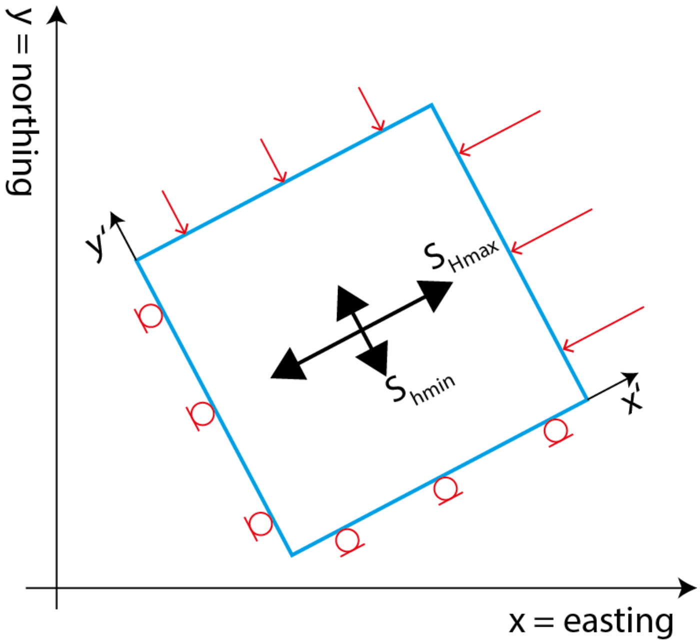


Fig. 2-1 Map view of a model area (blue box) in a geographical coordinate system.

The boundary conditions (red) are applied in a rotated coordinate system (x' and y') parallel to the principal horizontal stress axes. Model boundaries are oriented parallel to the prevailing orientations of the principal horizontal stress axes (Fig. 2-1). Information on the orientation is available from data of the World Stress Map database (Heidbach et al., 2016). The application of even small displacement boundary conditions (Dirichlet type) perpendicular to the model boundaries are usually sufficient to achieve a good fit of the overall stress orientation.

The stress magnitudes are calibrated on available information of the magnitudes of S_{Hmax} and S_{hmin}. The calibration of the horizontal stresses is achieved by a comparison of the modelled stress state to the S_{Hmax} and S_{hmin} magnitudes. The Dirichlet boundary conditions (displacement) are altered until a good fit of the modelled stress state to the observed stress information at the calibration points is achieved. The lateral variations in the stress field are hence entirely dependent on the variations in rock properties and the lithology. The fact that the models calibration depends on only two variables (deformation in the orientation of S_{Hmax} and S_{hmin}) facilitates the calibration as the best-fit boundary conditions can be found by a system of linear equations. The automated setup and solving of this linear equation system is the core of the FAST-Calibration tools.

Ziegler et al. (2016) propose the so-called multi-stage modelling approach. In this approach a large scale model called root model has a low resolution but within its perimeter data records for calibration are available. The root model is calibrated against the stress information and provides a good representation of the stress state in its entire volume. A second model with a high resolution and only a small extent is geographically located within the root model. This so called branch models extent does not necessarily coincide with the availability of any stress information. Therefore the branch model is calibrated on the modelled stress state from the root model. Fig. 2-2 displays the consecutive steps of a multi-stage approach of calibrated 3D geomechanical-numerical modelling of the stress state (Ziegler et al, 2016). To calibrate a single model (no multi-stage) only Fig. 2-2a-d are relevant. In the following, the single- and multi-stage approach is described.

Three different root test models with arbitrary but reasonable Dirichlet boundary conditions are solved (Fig. 2-2a). At several locations within the model area stress information is available from field observations (Fig. 2-2a). For each of the three test models the modelled stress state at the location of the stress information is compared to the actual stress state provided by the stress data records and for each test model a mean deviation for S_{Hmax} and S_{hmin}, respectively, is calculated (Fig. 2-2c). A linear equation system is setup in the domain of the boundary conditions (x and y) and the deviations of observed and modelled data (z) (Fig. 2-2d). For the smallest deviation of each, S_{Hmax} and S_{hmin}, an infinite number of corresponding boundary conditions are found. However, only one set of boundary conditions satisfies both the requirements for the smallest deviation in observed and modelled S_{Hmax} and S_{hmin}. This set of boundary conditions is applied to compute the best-fit model (see Ziegler et al., 2016).

The same concept is followed to calibrate a branch stress model which lies in a reservoir scaled area where no stress information is available. Hence, such a model is calibrated on the stress state provided by the larger root model. Three branch test models with arbitrary but reasonable Dirichlet boundary conditions are computed (Fig. 2-2e). At defined locations within the best-fit root model which also lie within the branch model's extent the reference stress state is collected (Fig. 2-2f). The modelled stress state from the three test models is compared to the reference stress state from the best-fit root model and the deviation between the root model stress state and the branch model test stress state is computed for each test model for S_{Hmax} and S_{hmin}, respectively (Fig. 2-2g). A linear equation system is setup in the domain of the boundary conditions (x and y) and the deviations of the root stress state for calibration and the branch stress state (z) (Fig. 2-2h). In Fig. 2-2h this is indicated by the intersection of the two bold lines of zero deviation. The boundary conditions at this point are used to compute the best-fit branch model.

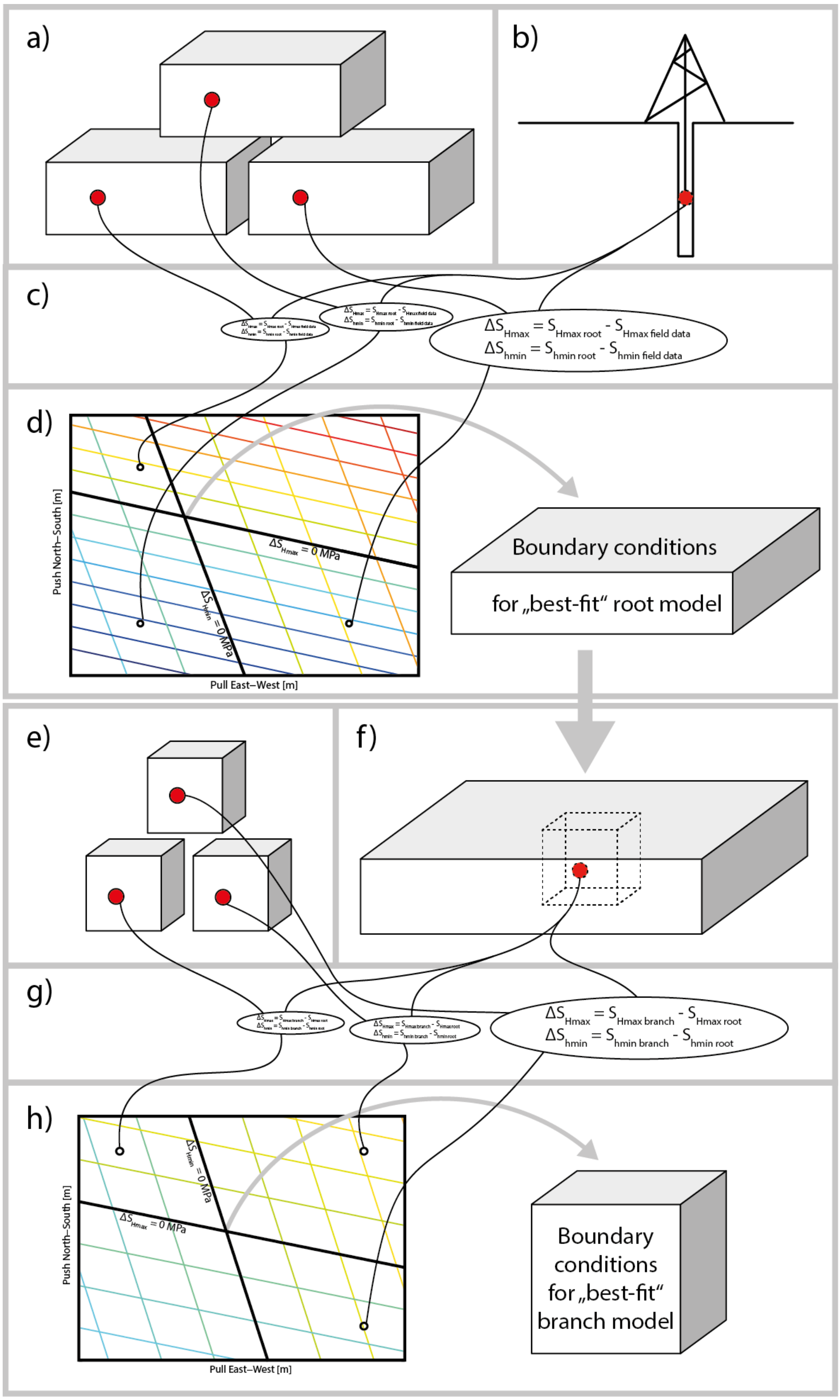


Fig. 2-2 The basic workflow of the multi-stage stress modelling approach.
Detail are presented in the publication of Ziegler et al. (2016).

3 Application of the tool

3.1 Preparation

The script can be downloaded from http://github.com/MorZieg/FAST_Calibration. The following preparations are required in order to successfully use the FAST-Calibration tool.

1. Create a directory with the following files and subdirectories:
   - root_calibration.m
   - branch_calibration.m
   - data (empty subfolder)
   - calib_functions (subfolder)
   - nodes2calibrationpoints.m
   - write_macro.m

2. At least three test model scenarios with arbitrary but reasonable boundary conditions are required for each model that should be calibrated. The test model scenarios should be loaded in Tecplot 360 EX and at least S_{Hmax} and S_{hmin} need to be computed. This is most easily done by the Tecplot 360 EX Addon GeoStress (Stromeyer and Heidbach, 2016). If S_{Hmax} and S_{hmin} are derived by another tool it is imperative that they are named “SHmax” and “Shmin” (not case sensitive). If the multi-stage approach is applied it is required that the differently sized models are in the same coordinate system and the smaller models extent is covered by the larger model.
3. If the multi-stage approach is applied a *.csv file is required which contains all nodes and their location of each root model that is going to provide the modelled stress state for a branch model. Four columns are required: Node number, X coordinate, Y coordinate, Z coordinate (see the example file). The *NODES section of an Abaqus input file is suitable.
4. To calibrate a branch model the best-fit root model is required to be loaded in Tecplot 360 EX and at least S_{Hmax} and S_{hmin} need to be computed by the Tecplot 360 EX Addon GeoStress.

3.2 Calibration on horizontal stress magnitude information (root model calibration)

The calibration of a model on horizontal stress magnitude information is conducted with help of the script root_calibration.m. The script is divided into two parts. The first part creates a Tecplot 360 EX macro. Run in Tecplot 360 EX it extracts the modelled stress state at the calibration points from the three different test models and writes it to a data file in the data subfolder that is accessible for Matlab. The second part of the script reads the data from this file, compares the modelled stress state with the observed stress magnitudes (S_{Hmax} and S_{hmin}), sets up and solves the linear equation system, and provides the best-fit boundary conditions for the root model to match the observed stress data records. The locations of the stress magnitude observations may be independent from each other. Since usually more S_{hmin} than S_{Hmax} data records are available it is likely that mainly S_{hmin} data records are used for the calibration. However, at least one S_{Hmax} magnitude is required. A schematic overview of the procedure is provided in Fig. 3.2-1.

In the first part of the script the user needs to define several variables to enable the execution of the script. Table 3.2-1 lists the required input. In the provided script file exemplified variables are defined.

Tab. 3.2-1 Required input variables for root_calibration.m.

Variable	**Description**
Folder	Provide the directions to the folder in your system which contains the script data. It is important to include the full path since this information is not required for Matlab (which supports relative paths) but for the Tecplot 360 EX macro (which does NOT support relative paths).
Shmax	Define the x, y, and z coordinates of S_{Hmax} data records for calibration. Furthermore the magnitude of S_{Hmax} (in MPa) and the confidence between 0 (low) and 1 (high) is specified. Please make sure that the coordinate system is the same as in the model. Note that z is not the True Vertical Depth (TVD) but the actual z-coordinate from the models coordinate system.
Shmin	Define the coordinates, magnitude, and confidence of S_{hmin} magnitude data.
Name	Provide a name for the macro file and data files the macro will create.
X	Enter the displacement in x' (S_{hmin} or S_{Hmax}) direction that is prescribed at the different test scenarios. Make sure to assign the values in the correct order.
Y	Enter the displacement in y' (S_{Hmax} or S_{hmin}) direction that is prescribed at the different test scenarios. Make sure to assign the values in the correct order.

Once the variables are defined this section of the script can be run. A Tecplot 360 EX macro file of the specified name is created and written to the current working directory of Matlab (it is encouraged to use the specified folder). The macro exports the modelled stress state at each location of a stress data record for each of the test scenarios. Therefore, load the geomechanical model with test boundary conditions, derive S_{Hmax} and S_{hmin} with GeoStress, and execute the macro in Tecplot 360 EX. Then proceed to the second part of the script.

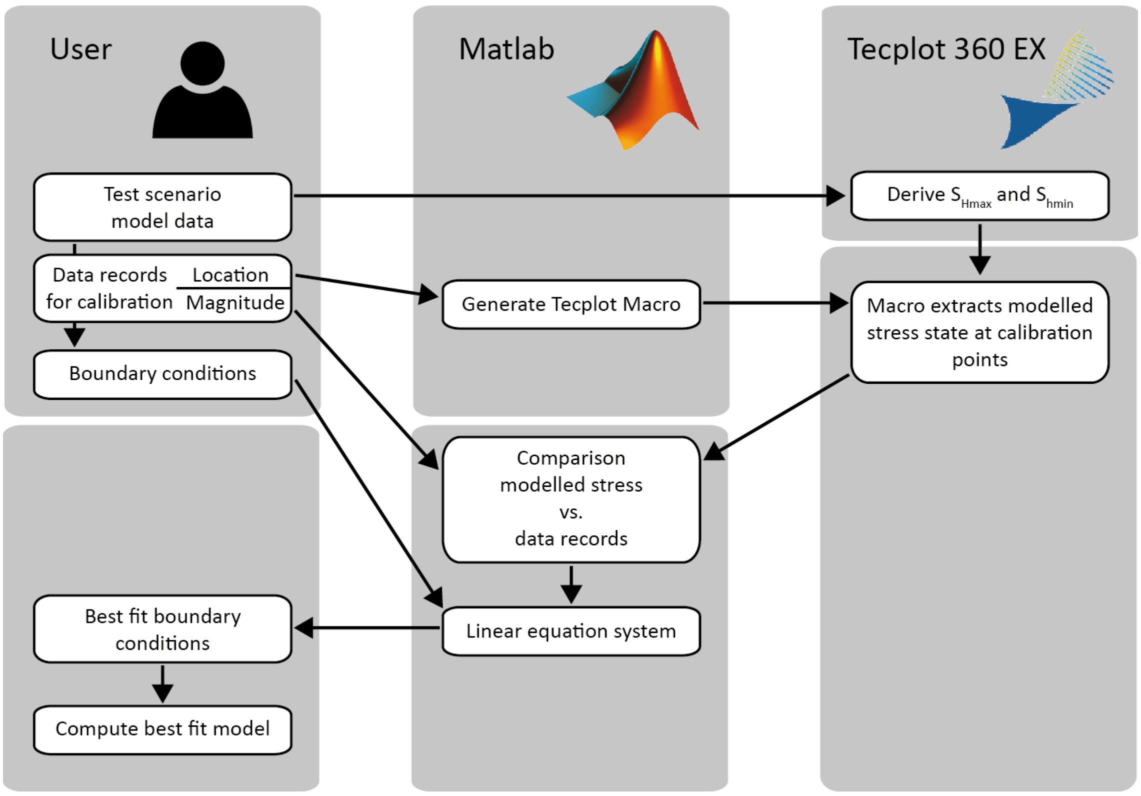


Fig. 3.2-1 Schematic chart of the script workflow

Workflow of the calibration of a model on stress data records in two steps

The second part of the script reads the data that was exported from Tecplot 360 EX. The modelled S_{Hmax} and S_{hmin} data is compared to the actual stress data records. The deviation of the modelled and observed data record for each boundary condition scenario (bc) at each location (x, y, z) with available observed stress data records is computed by

$$\Delta S_{Hmax}(x,y,z,bc) = S_{Hmax,modelled}(x,y,z,bc) - S_{Hmax,observed}(x,y,z)$$

exemplified here for S_{Hmax}. The weighted mean deviation of the modelled and observed S_{Hmax} and S_{hmin} magnitudes are computed for each boundary condition scenario by

$$\widetilde{\Delta S}_{Hmax}(bc) = \frac{\sum_i w_i \, \Delta S_{Hmax}(x_i, y_i, z_i, bc)}{\sum_i w_i}$$

with i the number of data records and $\mathrm{w_i}$ the individual weighting of each data record. Now for each boundary condition scenario a mean deviation of S_{Hmax} and S_{hmin} exist. An example of the information that is available now is shown in Tab. 3.2-2.

Tab. 3.2-2 Example for the information required for the generation of the planes.
At least three different boundary condition scenarios are required. For each scenario the displacement in x' and y' and $\widetilde{\Delta S}_{\mathrm{Hmax}}$ and $\widetilde{\Delta S}_{\mathrm{hmin}}$ are available.

bc	**x' displ.**	**y' displ.**	$\widetilde{\Delta S}_{Hmax}$	$\widetilde{\Delta S}_{hmin}$
1	-10	5	-12.3	8.7
2	-30	25	2.5	-3.2
3	-30	5	-14.4	-4.8

This information is used to derive the boundary conditions that fulfil the requirement of no deviation between the observed and modelled $\widetilde{\Delta S}_{\mathrm{Hmax}}$ and $\widetilde{\Delta S}_{\mathrm{hmin}}$ magnitudes. It is solved by a system of linear equations that can be visualised as two intersecting planes.

For each S_{Hmax} and S_{hmin} the equation of the plane that is defined by the displacement boundary conditions in x' and y' direction and the mean deviation of S_{Hmax} and S_{hmin}, respectively, are set up in $\mathbb{R}^3(\mathrm{x}', \mathrm{y}', \widetilde{\Delta S}_{\mathrm{Hmax}})$ and $\mathbb{R}^3(\mathrm{x}', \mathrm{y}', \widetilde{\Delta S}_{\mathrm{hmin}})$, respectively, in coordinate form (Fig. 3.2-2):

$$n_1\, x' + n_2\, y' + n_3\, \widetilde{\Delta S}_{Hmax} = d$$

with $\vec{\mathrm{n}}$ as the normal vector of the plane and d as

$$d = \vec{p} \cdot \vec{n}$$

with $\vec{\mathrm{p}}$ the planes position vector.

Then $\widetilde{\Delta S}_{\mathrm{Hmax}}$ is set to $\widetilde{\Delta S}_{\mathrm{Hmax}} = 0$ which represents the line with no deviation between observed and modelled stress state (black solid lines in Fig. 3.2-2).

$$n_1\, x' + n_2\, y' \qquad = d$$

Then the equation is transformed to represent this line in $\mathbb{R}^2(\mathrm{x}', \mathrm{y}')$.

$$y' = \frac{d - n_1 x'}{n_2}$$

$\widetilde{\Delta S}_{hmin}$ is set to $\widetilde{\Delta S}_{hmin} = 0$ which defines the according line of zero deviation of modelled from observed $\widetilde{\Delta S}_{hmin}$ magnitude. Both lines of zero deviation are now defined in $\mathbb{R}^2(x', y')$. At their intersection both, $\widetilde{\Delta S}_{Hmax}$ and $\widetilde{\Delta S}_{hmin}$, are zero and hence x' and y' at the intersection define the best-fit boundary conditions (indicated by the red circle and dashed red lines in Fig. 3.2-2).

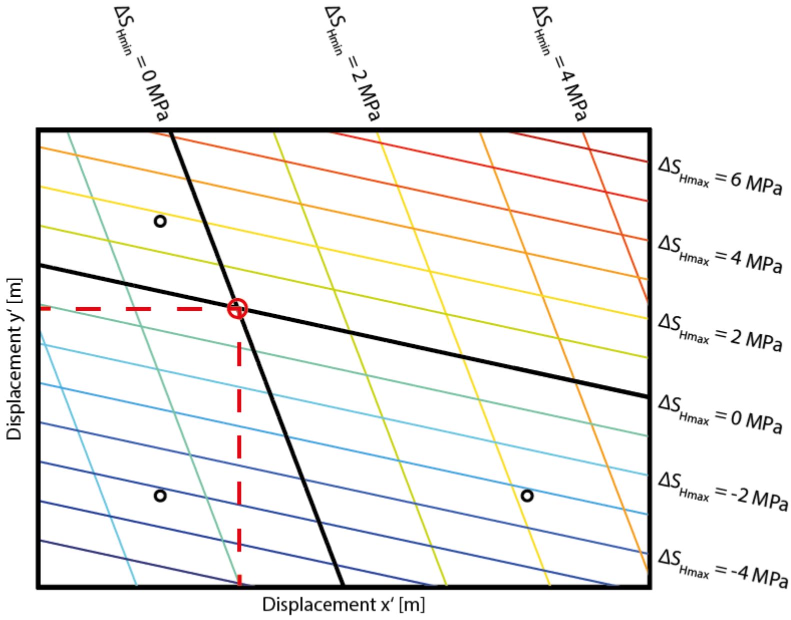


Fig. 3.2-2 Best-fit boundary conditions.

Red dashed line and red circle are displayed in $\mathbb{R}^2(x', y')$. Three test scenarios (black circles) provide mean deviations of modelled and observed S_{Hmax} and S_{hmin} magnitudes for specific boundary conditions. For both, S_{Hmax} and S_{hmin}, the plane spanned by the boundary conditions and the deviations in $\mathbb{R}^3(x', y', \widetilde{\Delta S}_{Hmax})$ and $\mathbb{R}^3(x', y', \widetilde{\Delta S}_{hmin})$, respectively, are sought (represented here by the colour coded isolines). The best-fit boundary conditions are found where the isolines z=0 of the two planes intersect.

3.3 Calibration of another model (branch calibration)

The calibration of a model on stress data provided by another (larger) model is achieved by the script branch_calibration.m. Basically the script works in the same way as the script to calibrate the root model. A schematic overview of the scripts functions and steps is displayed in Fig. 3.3-2.

The main difference between the root and branch calibration is that for the calibration of a model on data provided by another model a virtually unlimited amount of calibration points is available. However, the calibration points need to be chosen with care (Ziegler et al, 2016). It is important to (1) only choose calibration points which are nodes in the model that provides the reference calibration points. Otherwise the interpolation of the visualization software from nodes into the volume may result in erroneous deviations. (2) The points used for calibration should be located at the border of the branch model. Otherwise the stress field from the larger and less detailed root model is imposed everywhere in the branch model. Thereby no lateral stress variations due to features that are only present in the branch model would be allowed (Fig. 3.3-1). Suitable calibration points are provided via subroutine nodes2calibrationpoints.m (for details see the corresponding section on the function).

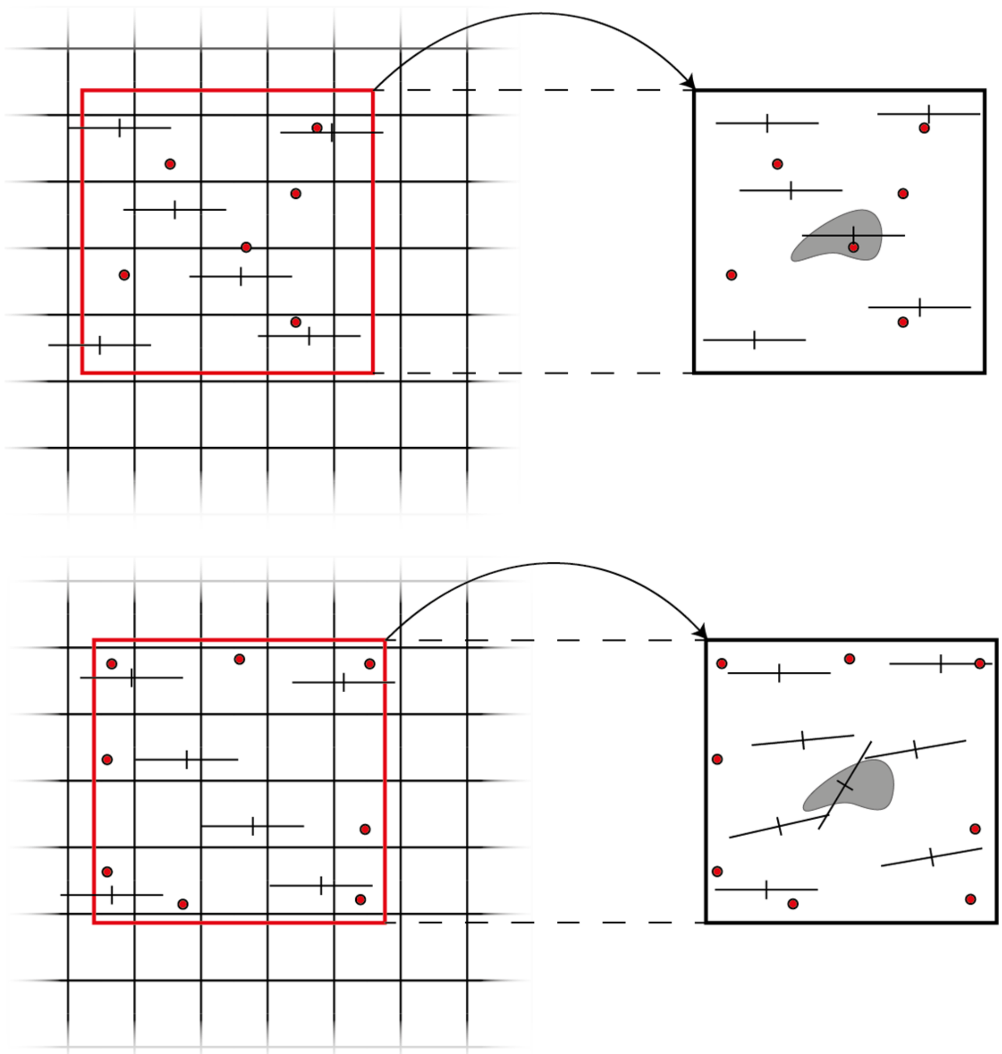


Fig. 3.3-1 Distribution of calibration points.

Red dots are the calibration points that are randomly distributed (top) versus calibration points at the borders of the branch model (bottom) in the root model (left) and branch model (right). The lines indicate the orientation of S_{Hmax} (long line) and S_{hmin} (short line). A calibration point from the root model (left) that is located within a different material in the middle of the branch model (right) imposes an erroneous stress state on the branch model (top). If the calibration points are at the borders of the branch model material contrasts only present in the branch model are free to influence the stress state (bottom).

In the first part of the script the user needs to define several variables to enable the script to run. Table 3.3-1 lists the required input. In the provided script file exemplified variables are defined.

Tab. 3.3-1 Required input variables for branch_calibration.m.

Variable	Description
folder	Provide the directions to the folder in your system which contains the script data. It is important to include the full path since this information is not used in Matlab but for the Tecplot 360 EX macro which does not support relative paths.
branch_corner	Define the corners of the branch model in the model coordinate system in a clockwise manner. Make sure that both models are in the same coordinate system.
root_nodes	Provide the filename and location of the *.csv file that contains the nodes of the root model.
type	Specify where the calibration points should be located in the branch model. The three options are: 'border': The calibration points are distributed at the sidewalls of the model. 'corner': The calibration points are mostly located in the corners of the model. 'random': The calibration points are randomly distributed within the entire model (not recommended in a classical multistage application).
num	Define the desired number of calibration points. Depending on the type the number of output calibration points may vary slightly.
distrib	This variable defines the density of the calibration points in the corners. A value between 10 and 30 is recommended. (Only for type corner.)
minelem	This variable defines the distance between the boundary of the model and the closest calibration point.
Zmax	The minimum topographic elevation in the entire branch model. This is at the same time the highest topographic isoline that is present in the entire model. No calibration points are situated above this value.
Zmin	The bottom of the branch model. This is especially important if the branch model does not extend as deep as the root model. No calibration points are situated below this value.
x	Enter the displacement in x' (S_{hmin} or S_{Hmax}) direction that is prescribed in the different test scenarios. Make sure to assign the values in the correct order.
y	Enter the displacement in y' (S_{Hmax} or S_{hmin}) direction that is prescribed in the different test scenarios.

In the first part of the script the function nodes2calibrationpoints.m is called and provides the desired amount of calibration points at nodes in the root model. The distribution is according to the specified type. A map with the derived calibration points is displayed once they are available. Please note that the subroutine may take some time to compute especially for large models. Furthermore, the function write_macro.m is called twice and generates two Tecplot 360 EX macros which are written to Matlabs current working directory. The first macro is used to extract the best-fit stress state at the calibration points from the root model. The second macro extracts the comparison stress data from the branch test models.

Before the second part of the script is executed, the best-fit root model and the branch test models need to be loaded in Tecplot 360 EX and the S_{hmin} and S_{Hmax} magnitudes are derived by the Tecplot 360 EX Addon GeoStress. Then the macros are called in order to export the modelled stress state at the designated calibration points.

The second part of the script reads the data that was exported from Tecplot 360 EX. The modelled stress data from the branch model test scenarios is compared to the reference stress data from the root model. Therefore for each test scenario a deviation of the modelled S_{Hmax} and S_{hmin} magnitude from the reference magnitudes is computed for each calibration point and a mean deviation is computed. Then, the system of linear equations is setup and solved (for details see previous section). Finally, the best-fit boundary conditions are presented. They are used to compute the best-fit branch model.

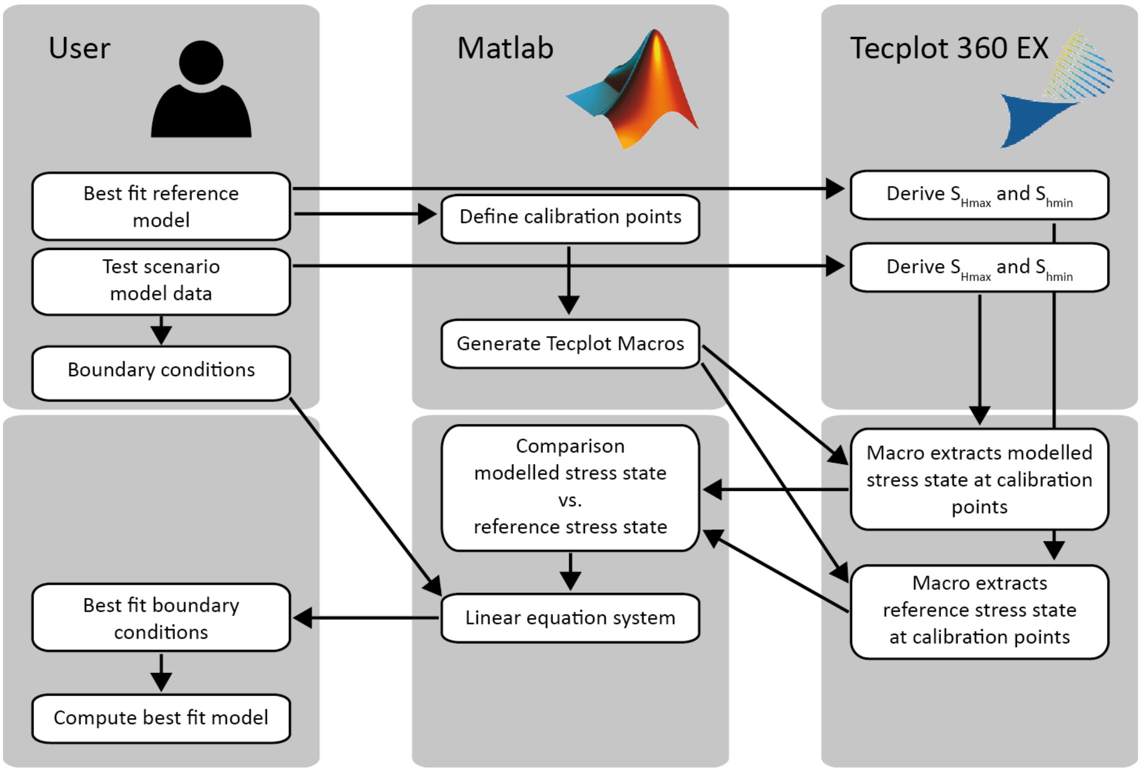


Fig. 3.3-2 Schematic chart of the script workflow.
Workflow for the calibration of a model on a modelled reference stress state in two steps.

3.4 Writing Tecplot 360 EX macro function

The function write_macro.m generates a Tecplot 360 EX macro that exports data records from given locations in the model, usually at calibration points. The function requires five input variables (Tab. 3.4-1) which are automatically provided and transmitted by the root and branch calibration scripts. Since the calibration points are not necessarily at nodes the variables are interpolated from the nodes to the exact coordinates in the volume which is done by the macro generated by this function.

Tab. 3.4-1 Required input variables for the function write_macro.m.

Variable	Description
shmax	An n x 3 Matrix with the x, y, and z coordinates of the S_{Hmax} datapoints.
shmin	An n x 3 Matrix with the x, y, and z coordinates of the S_{hmin} datapoints.
name	The desired name of the macro without the filetype extension ".mcr".
mod	The number of modelled stress states (steps) from which data should be exported. For each stress state the data is exported at all locations specified in S_{Hmax} and S_{hmin}.
folder	Provide the directions to the folder in your system which contains the script data. It is important to include the full path since this information is not used in Matlab but for the Tecplot 360 EX macro which does not support relative paths.

With these variables the function creates a Tecplot 360 EX macro with the following structure.

1. The internal number of the variables S_{Hmax} and S_{hmin} in Tecplot 360 EX are sought and stored in the macro variable S_{HMAX} and S_{HMIN}.

```
$!GETVARNUMBYNAME |SHMAX|
NAME = "SHmax”
```

2. For each datapoint specified in the variable A an individual 1D zone (point) is created. The following example is repeated accordingly often.

```
$!CREATERECTANGULARZONE
IMAX = 1
JMAX = 1
KMAX = 1
X1 = 6.621267e+05
Y1 = 5.300777e+06
Z1 = -1.259692e+02
X2 = 6.621267e+05
Y2 = 5.300777e+06
Z2 = -1.259692e+02
```

3. The variables are linearly interpolated on the zones defined in step 1. The following example is repeated accordingly often.

```
$!LINEARINTERPOLATE
SOURCEZONES =  [4]
DESTINATIONZONE = 124
VARLIST =  [SHMAX]
LINEARINTERPCONST = 0
LINEARINTERPMODE = DONTCHANGE
```

4. The variable values in the 1D zones are exported to a comma-separated datafile.

```
$!EXTENDEDCOMMAND
COMMANDPROCESSORID = 'excsv'
COMMAND = 'FrOp=1:ZnCount=6:ZnList=[4-9]:VarCount=1:VarList=[SHMAX]:ValSep=",":FNAME="C:\Documents\Geom-Num-Model\Calibration\data\root_shmax.csv"'
```

5. The 1D zones are deleted from the Tecplot 360 EX file.

```
$!DELETEZONES [101-500]
```

Steps 2 – 5 are repeated once for the S_{Hmax} datapoints and once for the S_{hmin} datapoints. Once the macro is written to the current output directory of Matlab the text message “Macro file created” is returned to the caller.

3.5 Calibration point function

The function nodes2calibrationpoints.m provides suitable calibration points for the calibration of a geomechanical-numerical branch model on the stress state provided by a root model. Therefore, a user-defined number of calibration points is selected out of all nodes from the root model that are geographical located within the boundaries of the branch model.

The function provides three different possibilities for the lateral distribution of calibration points (Fig. 3.5-1). The nodes for calibration can be located (1) equally distributed at the borders of the branch models, (2) in the corners of the branch model, and (3) randomly distributed in the entire branch model (which is not recommended in a classical multistage application). The depth distribution of the calibration points is random. The minimum distance of the calibration nodes from the border of the branch model can be specified in order to mitigate boundary effects. Furthermore, if the nodes are sought in the corner the density of the nodes can be controlled (Fig. 3.5-1). The subroutine uses the Matlab function inpolygon in order to determine whether a node is within the specified boundaries of the branch model. Therefore, the model boundaries can be arbitrarily oriented and are not required to be parallel to the model coordinate systems axes. The function requires the input information specified in Table 3.5-1.

Tab. 3.5-1 Required input variables for the function nodes2calibpoints.m.

Variable	Description
corner	Clockwise X and Y coordinates of the four corners of the branch model.
filename	Name of the file that contains the nodes of the root model.
num	Number of desired calibration points.
distrib	This variable defines how close the calibration points should be in the corners. A value between 10 and 30 is recommended. (Only for type corner.)
type	The type of distribution for the calibration points, 'border', ‘corner', or 'random'.
minelem	This variable defines the distance between the boundary of the model and the closest calibration point.
Zmax	The minimum topographic elevation in the entire branch model. This is at the same time the highest topographic isoline that is present in the entire model. No calibration points are situated above this value.
Zmin	The bottom of the branch model. This is especially important if the branch model does not extend as deep as the root model. No calibration points are situated below this value.

The function returns a num x 3 shaped Matlab variable with the coordinates of the derived calibration points. Please note that depending on the chosen type of calibration point distribution the number of calibration points may vary slightly due to technical reasons.

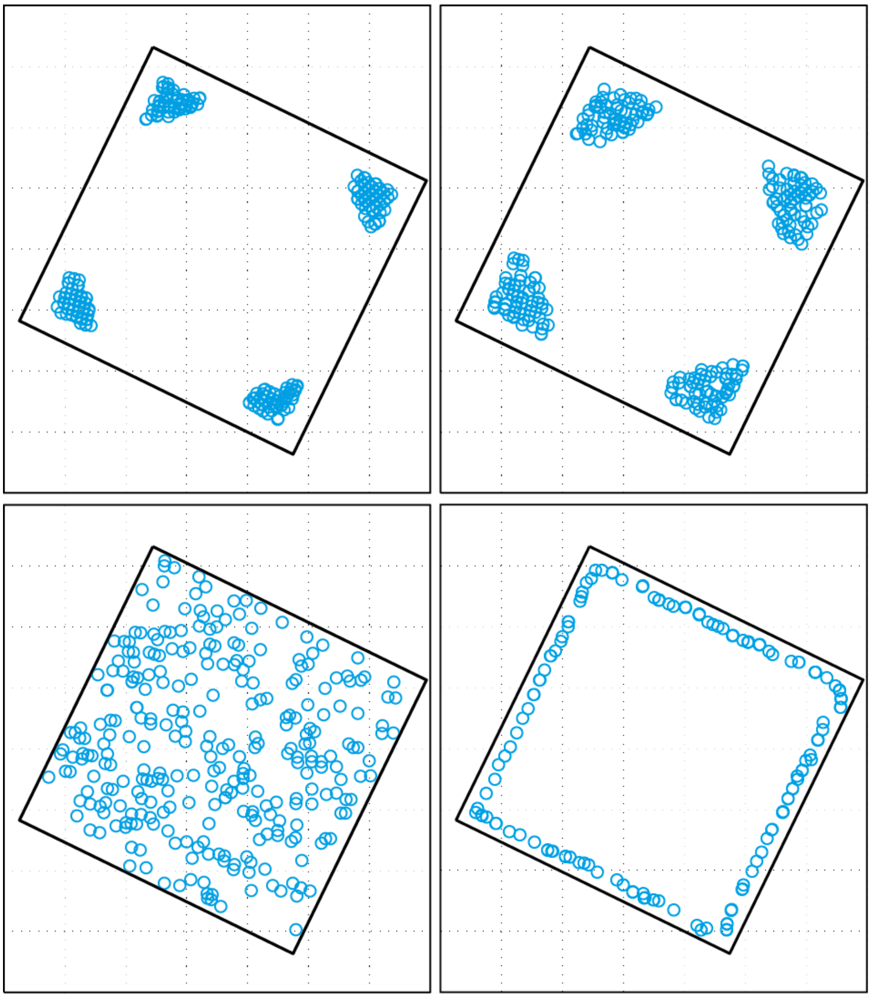


Fig. 3.5-1 Different possibilities for calibration point positioning.

Top left to bottom left clockwise: Calibration points at the corner with distrib = 20 and distrib = 50 and minelem 2000, at the borders of the model with minelem = 1000, and randomly distributed with minelem 500.

4 Example

In order to test the FAST Calibration and multistage modelling an example is provided. The example consists of a small root and branch model. The root model has an extent of 10×10×5 km³ and the branch model has an extent of 4×4×4 km³. Detailed coordinates are listed in Tab. 4-1. The models are presented as Abaqus solver input files (.inp) and output files (.odb). For both, root and branch, an initial stress state, a calibration model with three boundary condition scenarios, and a final model are presented. The geometry is in separate files (.geom). Furthermore, the nodes of the root model are represented in a .csv file. The exemplary parameters that are provided in the Matlab scripts match these examples.

Tab. 4-1 Extent and coordinates of the example models.

Model	Coordinates	Min [m]	Max [m]
Root	X	0	10,000
	Y	0	10,000
	Z	-5,000	0
Branch	X	2,000	6,000
	Y	2,000	6,000
	Z	-4,000	0

The .odb files can be directly loaded in Tecplot 360 EX and the FAST Calibration can hence be tested without using Abaqus to solve the models. Please note that the models were solved using Abaqus 6.14. Output files from this version of Abaqus can only be read from Tecplot 360 EX 2017 onwards. For compatibility with older Tecplot 360 EX versions or in order to experience the entire workflow the input files can be rerun in Abaqus (older Tecplot 360 EX versions require Abaqus 6.11 output files). The following steps are required:

1. Abaqus: Run root_initial.inp and root_calibration.inp.
2. Matlab: Execute the first part of root_calibration.m. A Tecplot 360 EX macro file is written in the current Matlab directory.
3. Tecplot 360 EX: Load root_calibration.odb, compute S_{Hmax} and S_{hmin} with the GeoStress Addon, and execute the macro root.mcr that was generated by Matlab in 2. S_{Hmax} and S_{hmin} at the calibration points are written to two files in the /data/ subdirectory.
4. Matlab: Execute the second part of root_calibration.m. Matlab provides the best-fit boundary conditions for the root model.
5. Open a text editor and set the boundary conditions in root_final.inp according to the output of Matlab in 4.
6. Abaqus: Run root_final.inp.

Now, the root model is successfully calibrated on stress data records. For a multistage procedure continue as follows:

7. Abaqus: Run branch_initial.inp and branch_calibration.inp.
8. Matlab: Execute the first part of branch_calibration.m. Two Tecplot 360 EX macros files are written in the current Matlab directory.
9. Tecplot 360 EX: Load branch_calibration.odb, compute S_{Hmax} and S_{hmin} with the GeoStress Addon, and execute the branch_calib.mcr macro that was generated by Matlab in 8. S_{Hmax} and S_{hmin} at the calibration points are written to two files in the /data/ subdirectory.
10. Tecplot 360 EX: Load root_final.odb, compute S_{Hmax} and S_{hmin} with the GeoStress Addon, and execute the macro root_branch_calib.mcr that was generated by Matlab in 8. S_{Hmax} and S_{hmin} at the calibration points are written to two files in the /data/ subdirectory.
11. Matlab: Execute the second part of branch_calibration.m. Matlab provides the best-fit boundary conditions for the branch model.

12. Open a text editor and set the boundary conditions in branch_final.inp according to the output of Matlab in 10.
13. Abaqus: Run branch_final.inp.

5 Acknowledgements

The authors would like to thank Karsten Reiter and Tobias Hergert who contributed to the script and Robin Seithel for initial testing. The work leading to these results has received funding from the European Community's Seventh Framework Programme under grant agreement No. 608553 (Project IMAGE) and Germany's Federal Ministry for Education and Research under the FONA/GEO:N programme under grant agreement No. 03G0869A (Project SUBI).

6 References

Cornet, F..H. 2015. Elements of Crustal Geomechanics. Cambridge University Press. http://doi.org/10.1007/s40948-015-0016-9

Heidbach, O., Tingay, M., Barth, A., Reinecker, J., Kurfeß, D., & Müller, B. 2010. Global crustal stress pattern based on the World Stress Map database release 2008. Tectonophysics, 482(1-4), 3–15. http://doi.org/10.1016/j.tecto.2009.07.023

Heidbach, O., Rajabi, M., Reiter, K., Ziegler, M., & the WSM Team. 2016. World Stress Map Database Release 2016. GFZ Data Services. http://doi.org/10.5880/WSM.2016.001

Hergert, T., Heidbach, O., Reiter, K., Giger, S. B., & Marschall, P. 2015. Stress field sensitivity analysis in a sedimentary sequence of the Alpine foreland, northern Switzerland. Solid Earth, 6(2), 533–552. http://doi.org/10.5194/se-6-533-2015

Reiter, K., & Heidbach, O. 2014. 3-D geomechanical-numerical model of the contemporary crustal stress state in the Alberta Basin (Canada). Solid Earth, 5(2), 1123–1149. http://doi.org/10.5194/se-5-1123-2014

Stromeyer, D., & Heidbach, O. 2017a. Manual of the Tecplot 360 Add-on GeoStress. World Stress Map Technical Report 17-01. GFZ German Research Centre for Geosciences. http://doi.org/10.2312/wsm.2017.001

Stromeyer, D., & Heidbach, O. 2017b. Tecplot 360 Add-on GeoStress. GFZ Data Services. http://doi.org/10.5880/wsm.2017.001

Zang, A., Stephansson, O., Heidbach, O., & Janouschkowetz, S.. 2012. World Stress Map Database as a Resource for Rock Mechanics and Rock Engineering. Geotechnical and Geological Engineering, 30(3), 625–646. http://doi.org/10.1007/s10706-012-9505-6

Ziegler, M. O., Heidbach, O., Reinecker, J., Przybycin, A. M., & Scheck-Wenderoth, M.. 2016. A multi-stage 3-D stress field modelling approach exemplified in the Bavarian Molasse Basin. Solid Earth, 7(5), 1365–1382. http://doi.org/10.5194/se-7-1365-2016