crystalAligner v1.0 - Documentation

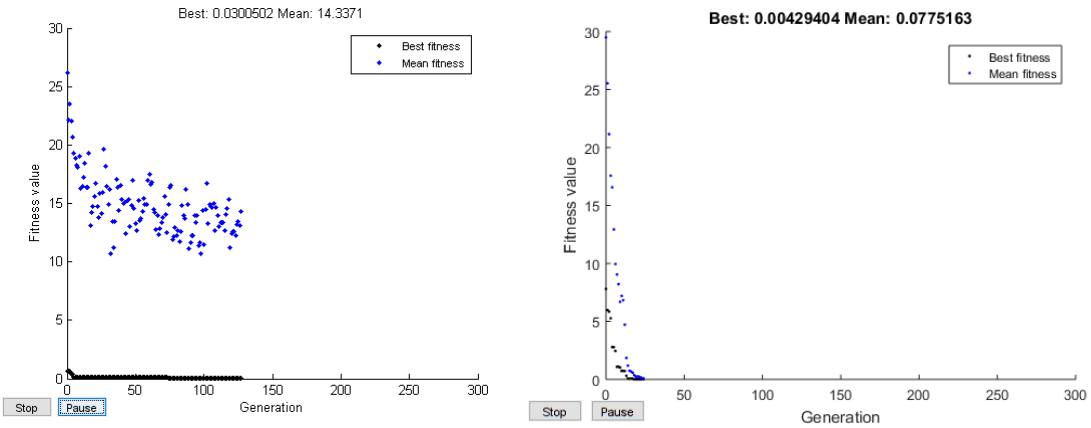
1 About

crystalAligner is a computer program for alignment of crystals in a scanning electron microscope. Given one or two crystal orientations (obtained from i.e. EBSD), one or two crystal directions and associated axes of the microscope coordinate system as alignment objectives and the rotational axes and limits of the microscope stage, the program optimizes the alignment of the crystal(s) with the microscope coordinate system by global optimization. The strength of *crystalAligner* is that it can utilize crystal symmetries and works with a heuristic global optimization algorithm. Even in the case of a standard microscope stage the alignment can therefore often be optimized to a satisfactory degree. A scientific treatment of the crystallographic aspects and the optimization procedure can be found in the associated publication [in Preparation].

2 Platform and Installation

The program was developed in MATLAB and requires the installation of the crystallographic texture analysis toolbox MTEX as well as the MATLAB global optimization toolboxes. The global optimization toolbox often comes preinstalled with MATLAB or can be obtained separately. If you are not having the toolbox installed you should receive an error message related to that. MTEX can be downloaded free of charge. The *crystalAligner* program has been tested with MTEX versions 4.2.1 to 5.1.1 and MATLAB version R2014a, R2015a and R2016b.

From version R2016b forward MATLAB changed the algorithm for linear constraints in *ga* (details here). Older versions of *ga* will experience problems of running into local minima (see figure below). While no ideal fix for this is possible, increasing the population size will mitigate this problem (try increasing the initial population size from 100 to 200 in examples E1 and E4). Multiobjective optimization problems using *gamultiobj* are not affected by this.



Comparison of the convergence of the mean and best fitness value in the MATLAB R2014a vs R2016b solution of problem 2 in example E4.

On an installed MATLAB system including the global optimization toolbox, MTEX can be installed by following these instructions. You will have to navigate to your MTEX folder, right-click on it and choose 'Add to Path' and 'Selected Folder and Subfolders' to add the MTEX directory to your

MATLAB path. If your *crystalAligner.m* file is not in your MTEX folder, you will also have to add the path of your *crystalAligner.m* file to the MATLAB path in the same way. The program can be executed by entering 'crystalAligner' into the command window and pushing enter. It is however recommended to open the *crystalAligner.m* file in the editor, as the settings of the optimization problem should be checked and if necessary adjusted. From here the program can be executed by clicking on the play button.

3 DIFFERENT PARAMETER SETTINGS

All necessary input to the computer program is given in the section 'Initialization' in the top of the program. The meaning of each parameter is well documented in the program. A more detailed description of parameters is also treated in the associated publication [in Preparation]. In the following, some example parameter sets are presented. The m-file *exampleHeaders.m* contains all Initialization sections of the examples and is supplied with the program. Keep in mind that the optimization algorithm is a genetic algorithm, meaning that the presented solutions are not going to be exactly identical for each execution of the code, but very similar.

E1: ALIGNMENT OF A EITHER OF TWO EQUIVALENT CRYSTAL PLANE NORMALS IN A CUBIC CRYSTAL

Two equivalent crystal plane normal in a cubic crystal with Euler orientation [233 145 33] are aligned in two separate single-objective optimization problems. The crystal represents a ferrite grain in a steel sample. The optimization objective is to align the z-axis of the microscope with either of the {111} or {100} equivalent plane normals. The stage axes represent a standard stage of an FEI scanning electron microscope with chosen soft-limits in the tilt axis of 0 – 20°. The genetic algorithm is set up to have a population size of 100 individuals. Optional text output for estimating the trench sizes for a FIB lift-out at an inclined surface are deactivated. It is found that the {100} can be perfectly aligned while the optimal {111} alignment still deviates about 20° from ideal alignment, caused by the significant constraints by the SEM stage rotation limits (for MATLAB distributions <2016b an optimal solution with non-fulfilled linear constraints might be computed, see error description in the text box on page 1).

Settings for running this example:

```
% ***************************** Crystals *********************************
% *** Crystal Alignment Objective 1
crys.o(1,:)     = [313 15 137];
crys.cs{1}      = 'cubic';
crys.alignAx(1) = zvector;
crys.Miller{1}  = [1 1 1; 1 0 0];
crys.type{1}    = 'hkl';
crys.sym(1)     = 1;
% ****************************** Stage ***********************************
stg.rot     = [xvector; zvector];
stg.LB      = [    0     -180  ];
stg.UB      = [   20      180  ];
stg.sign    = [    1       -1  ];
stg.order   = [    2        1  ];
% ************************ Genetic algorithm *****************************
optim.popSz = 100;
optim.funcTol = 0.1;
optim.maxStallGen = 10;
optim.iterOut = 0;
optim.wghtFac = [1,1];
```

```
optim.multiCore = 0;
optim.hybridFcn = 0;
optim.autoSol = 1;
% ***************************** Optional *********************************
%FIB liftout calculations
FIB.mode = 0;
FIB.trench.ang = 52;
FIB.trench.z = 15;
FIB.axs.tilt = 1;
FIB.axs.rot = 2;
%Output
optim.plot = 1;
```

Expected result output in command window:

```
************************************************************
-> Plotting stereographic projections and IPF of crystal 1 ...
************************************************************
 Optimization problem 1 - Parallel alignment of:
   -> cubic 111 hkl with microscope Z-axis
************************************************************

Optimization terminated: average change in the fitness value less than
options.FunctionTolerance.

------------------------------------------------------------
*** Optimization results ***
Optimal parallel alignment of
      -microscope Z-axis with crystal hkl 111:
   -> Rotation around microscope X-axis: 19.9°
   -> Rotation around microscope Z-axis: -44.6°
   -> Deviation from ideal alignment in Z-axis: 19.8 °
------------------------------------------------------------

************************************************************
 Optimization problem 2 - Parallel alignment of:
   -> cubic 100 hkl with microscope Z-axis
************************************************************

Optimization terminated: average change in the fitness value less than
options.FunctionTolerance.

------------------------------------------------------------
*** Optimization results ***
Optimal parallel alignment of
      -microscope Z-axis with crystal hkl 100:
   -> Rotation around microscope X-axis: 15.0°
   -> Rotation around microscope Z-axis: 133.1°
   -> Deviation from ideal alignment in Z-axis: 0.0 °
------------------------------------------------------------
```

E2: ALIGNMENT OF TWO EQUIVALENT CRYSTAL DIRECTIONS IN A CUBIC CRYSTAL

This case is equivalent to the application example 1 in the publication [in Preparation]. Here, two equivalent crystal directions in a cubic crystal with Euler orientation [61 42 9] are aligned. The crystal represents a β-phase grain in a near-β Ti alloy. The optimization objectives are to align the y-axis of the microscope with any of the [113] equivalent crystal directions and the z-axis of the microscope

with any of the [110] equivalent crystal directions. The stage axes represent a standard stage of an FEI scanning electron microscope with chosen soft-limits in the tilt axis of 0 – 20°. The genetic algorithm is set up to have a population size of 100 individuals and no weighting factors applied to the objectives for the choice of the optimal solution. Optional text output for estimating the trench sizes for a FIB lift-out at an inclined surface and plotting of graphs are activated.

Settings for running this example:

```
% ***************************** Crystals *********************************
% *** Crystal Alignment Objective 1
crys.o(1,:)     = [61 42 9];                                              ]
crys.cs{1}      = 'cubic';
crys.alignAx(1) = yvector;
crys.Miller{1}  = [1 1 3];
crys.type{1}    = 'uvw';
crys.sym(1)     = 1;
% *** Crystal Alignment Objective 2
crys.o(2,:)     = [61 42 9];
crys.cs{2}      = 'cubic';
crys.alignAx(2) = zvector;
crys.Miller{2}  = [1 1 0];
crys.type{2}    = 'uvw';
crys.sym(2)     = 1;
% ******************************* Stage **********************************
stg.rot     = [xvector; zvector];
stg.LB      = [    0     -180  ];
stg.UB      = [   20      180  ];
stg.sign    = [    1       -1  ];
stg.order   = [    2        1  ];
% ************************ Genetic algorithm *****************************
%Genetic algorithm
optim.popSz = 100;
optim.funcTol = 0.1;
optim.maxStallGen = 10;
optim.iterOut = 0;
optim.wghtFac = [1,1];
optim.multiCore = 0;
optim.hybridFcn = 0;
optim.autoSol = 1;
% ***************************** Optional *********************************
FIB.mode = 1;
FIB.trench.ang = 52;
FIB.trench.z = 15;
FIB.axs.tilt = 1;
FIB.axs.rot = 2;
%Output
optim.plot = 1;
```

Expected result output in command window:

```
************************************************************

 Optimization problem 1 - Parallel alignment of:
   -> cubic 113 uvw with microscope Y-axis
   -> cubic 110 uvw with microscope Z-axis
************************************************************

Optimization terminated: average change in the spread of Pareto solutions less than
options.FunctionTolerance.
```

```
************************************************************
Optimal solution found by shortest distance to origin
under consideration of weighting factors 1 and 1 for alignment with microscope Y-axis and Z-axis

------------------------------------------------------------
*** Optimization results ***
Optimal parallel alignment of
        - microscope Y-axis with cubic crystal uvw 113 and
        - microscope Z-axis with cubic crystal uvw 110:
  -> Rotation around microscope X-axis: 6.9°
  -> Rotation around microscope Z-axis: 132.2°
  -> Deviation from ideal alignment in Y-axis: 0.1 °
  -> Deviation from ideal alignment in Z-axis: 0.4 °
------------------------------------------------------------
*** FIB - FEI Helios ***
Apply:
  -> Relative rotation of 132.2°
  -> Tilt at lift-out position: 6.9°
  -> Tilt at trenching position: 58.9°
  -> Alternative: Tilt at trenching position: 45.1° + 180° relative rotation
Trench lengths for 15.0 µm trench depth (z) and 52.0° trench angle:
  -> Trench length (y) at 'downhill position': 16.6 µm
  -> Trench length (y) at 'uphill position': 22.7 µm
------------------------------------------------------------
```

E3: ALIGNMENT OF TWO CRYSTAL DIRECTIONS IN TWO CRYSTALS OF DIFFERENT STRUCTURE AND ORIENTATION

This case is equivalent to the application example 2 in the publication [in Preparation]. Here, crystal directions in a cubic and an orthorhombic crystal with Euler orientations [261 43 28] and [175 20 102] are aligned. They represent a β-parent grain and an α'' martensitic variant in a near-β Ti deformation microstructure. The optimization objectives are to align the x-axis of the microscope with any of the <110> equivalent directions in β and the z-axis of the microscope with the [001] direction of the martensitic α''. The stage axes represent a standard stage of an FEI scanning electron microscope with chosen soft-limits in the tilt axis of 0 – 20°. The genetic algorithm is set up to have a population size of 500 individuals and no weighting factors applied to the objectives for the choice of the optimal solution. Optional text output for estimating the trench sizes for a FIB lift-out at an inclined surface is deactivated while plotting of graphs is activated.

Settings for running this example:

```
% ****************************** Crystals *********************************
% *** Crystal Alignment Objective 1
crys.o(1,:)     = [261 43 28];
crys.cs{1}      = 'cubic';
crys.alignAx(1) = xvector;
crys.Miller{1}  = [0 1 1];
crys.type{1}    = 'uvw';
crys.sym(1)     = 1;
% *** Crystal Alignment Objective 2
crys.o(2,:)     = [175 20 102];
```

```
crys.cs{2}      = 'orthorhombic';
crys.alignAx(2) = zvector;
crys.Miller{2}  = [0 0 1];
crys.type{2}    = 'uvw';
crys.sym(2)     = 0;
% ****************************** Stage ***********************************
stg.rot     = [xvector; zvector];
stg.LB      = [    0     -180  ];
stg.UB      = [   20      180  ];
stg.sign    = [    1       -1  ];
stg.order   = [    2        1  ];
% ************************ Genetic algorithm *****************************
optim.popSz = 500;
optim.funcTol = 0.1;
optim.maxStallGen = 10;
optim.iterOut = 0;
optim.wghtFac = [1,1];
optim.multiCore = 0;
optim.hybridFcn = 0;
optim.autoSol = 1;
% ***************************** Optional *********************************
%FIB liftout calculations
FIB.mode = 0;
FIB.trench.ang = 52;
FIB.trench.z = 15;
FIB.axs.tilt = 1;
FIB.axs.rot = 2;
%Output
optim.plot = 1;
```

Expected result output in command window:

```
************************************************************
 Optimization problem 1 - Parallel alignment of:
   -> cubic 011 uvw with microscope X-axis
   -> orthorhombic 001 uvw with microscope Z-axis
************************************************************

Optimization terminated: average change in the spread of Pareto solutions
less than options.FunctionTolerance.

************************************************************
Optimal solution found by shortest distance to origin
under consideration of weighting factors 1 and 1 for alignment with
microscope X-axis and Z-axis

------------------------------------------------------------
*** Optimization results ***
Optimal parallel alignment of
      - microscope X-axis with cubic crystal uvw 011 and
      - microscope Z-axis with orthorhombic crystal uvw 001:
   -> Rotation around microscope X-axis: 19.5°
   -> Rotation around microscope Z-axis: 7.1°
   -> Deviation from ideal alignment in X-axis: 6.3 °
   -> Deviation from ideal alignment in Z-axis: 4.1 °
------------------------------------------------------------
```

E4: EXTENSION TO EXAMPLE E1 BY ADDING A SUBSTAGE

In example E1 it was found that the {100} can be perfectly aligned while the optimal {111} alignment still deviates about 20° from ideal alignment, caused by the significant constraints by the SEM stage rotation limits. Let's investigate the effect of adding a sub-stage on the achievable alignment for {111}. Compared to the setup for example E1 a rotation axis around the microscope y-axis was added (stg.rot). The rotation limits for this axis are defined as -45 – 45°. Here it is important to consider the hierarchy of the rotational axes: The sub-stage represents the 1st tilt rotation which should be aligned with the microscope y-axis as an initial position, the microscope z-axis the 2nd rotation and the microscope x-axis the 3rd rotation (stg.order = [3 1 2). The additional degree of freedom compared to the setup in example E1 leads to perfect alignment of the {111} with the microscope z coordinate axis. Repetitive execution of the code will produce different sets of stage-rotations that lead to the same angular deviation, which is a good indicator of the expansion of the solution space by adding a third rotational axis. As an additional example one could try to set the limits of the substage to i.e. 45 – 45°, which would simulate the use of a fixed 45° pre-tilt sample holder.

Settings for running this example:

```
% *** Crystal Alignment Objective 1
crys.o(1,:)     = [313 15 137];
crys.cs{1}      = 'cubic';
crys.alignAx(1) = zvector;
crys.Miller{1}  = [1 1 1; 1 0 0];
crys.type{1}    = 'hkl';
crys.sym(1)     = 1;
% ****************************** Stage ***********************************
stg.rot     = [xvector; yvector; zvector];
stg.LB      = [    0     -45      -180  ];
stg.UB      = [   20      45       180  ];
stg.sign    = [    1      1        -1   ];
stg.order   = [    3      1         2   ];
% ************************* Genetic algorithm ****************************
optim.popSz = 100;
optim.funcTol = 0.1;
optim.maxStallGen = 10;
optim.iterOut = 0;
optim.wghtFac = [1,1];
optim.multiCore = 0;
optim.hybridFcn = 0;
optim.autoSol = 1;
% ***************************** Optional *********************************
%FIB liftout calculations
FIB.mode = 0;
FIB.trench.ang = 52;
FIB.trench.z = 15;
FIB.axs.tilt = 1;
FIB.axs.rot = 2;
%Output
optim.plot = 1;
optim.plot = 1;
```

Expected result output in command window:

```
************************************************************
 Optimization problem 1 - Parallel alignment of:
   -> cubic 111 hkl with microscope Z-axis
```

```
************************************************************

Optimization terminated: average change in the fitness value less than
options.FunctionTolerance.

------------------------------------------------------------
*** Optimization results ***
Optimal parallel alignment of
      -microscope Z-axis with crystal hkl 111:
   -> Rotation around microscope X-axis: 13.4°
   -> Rotation around microscope Y-axis: -37.7°
   -> Rotation around microscope Z-axis: 28.9°
   -> Deviation from ideal alignment in Z-axis: 0.0 °
------------------------------------------------------------

************************************************************
 Optimization problem 2 - Parallel alignment of:
   -> cubic 100 hkl with microscope Z-axis
************************************************************

Optimization terminated: average change in the fitness value less than
options.FunctionTolerance.

------------------------------------------------------------
*** Optimization results ***
Optimal parallel alignment of
      -microscope Z-axis with crystal hkl 100:
   -> Rotation around microscope X-axis: 8.1°
   -> Rotation around microscope Y-axis: 12.7°
   -> Rotation around microscope Z-axis: 75.1°
   -> Deviation from ideal alignment in Z-axis: 0.0 °
------------------------------------------------------------
```

E5: ALIGNMENT OF A SET OF CRYSTAL PLANE NORMALS AND CRYSTAL DIRECTIONS IN A CUBIC AND HEXAGONAL CRYSTAL OF DIFFERENT ORIENTATION

This example is an adaption of example 3 to demonstrate the use of 4-indicee Miller notation for hexagonal crystals and the distinction of crystal directions and plane normal. Here, a set of crystal plane normals in a cubic crystal and a crystal direction in a hexagonal crystal with Euler orientations [246 36 75] and [91 94 13] are aligned. The optimization objectives are to align the x-axis of the microscope with any of the {011} equivalent plane normals in the cubic crystal and the z-axis of the microscope with the <11-20> equivalent crystal directions of the hexagonal crystal. The stage axes represent a standard stage of an FEI scanning electron microscope with chosen soft-limits in the tilt axis of 0 – 20°. The genetic algorithm is set up to have a population size of 200 individuals and no weighting factors applied to the objectives for the choice of the optimal solution. Optional text output for estimating the trench sizes for a FIB lift-out at an inclined surface is deactivated while plotting of graphs is activated.

Settings for running this example:

```
% ***************************** Crystals *********************************
% *** Crystal Alignment Objective 1
crys.o(1,:)     = [246 36 75];
crys.cs{1}      = 'm-3m';
```

```
crys.alignAx(1) = xvector;
crys.Miller{1}  = [0 1 1];
crys.type{1}    = 'hkl';
crys.sym(1)     = 1;
% *** Crystal Alignment Objective 2
crys.o(2,:)     = [91 94 13];
crys.cs{2}      = 'P63/mmc';
crys.alignAx(2) = zvector;
crys.Miller{2}  = [1 1 -2 0];
crys.type{2}    = 'uvtw';
crys.sym(2)     = 1;
% ****************************** Stage ***********************************
stg.rot     = [xvector; zvector];
stg.LB      = [    0     -180  ];
stg.UB      = [   20      180  ];
stg.sign    = [    1       -1  ];
stg.order   = [    2        1  ];
% ************************ Genetic algorithm *****************************
optim.popSz = 200;
optim.funcTol = 0.1;
optim.maxStallGen = 10;
optim.iterOut = 0;
optim.wghtFac = [1,1];
optim.multiCore = 0;
optim.hybridFcn = 0;
optim.autoSol = 1;
% ***************************** Optional *********************************
FIB.mode = 0;
FIB.trench.ang = 52;
FIB.trench.z = 15;
FIB.axs.tilt = 1;
FIB.axs.rot = 2;
optim.plot = 1;
```

Expected result output in command window

```
************************************************************
 Optimization problem 1 - Parallel alignment of:
   -> m-3m 011 hkl with microscope X-axis
   -> P63/mmc 330 uvw with microscope Z-axis
************************************************************

Optimization terminated: average change in the spread of Pareto solutions
less than options.FunctionTolerance.

************************************************************
Optimal solution found by shortest distance to origin
under consideration of weighting factors 1 and 1 for alignment with
microscope X-axis and Z-axis

------------------------------------------------------------
*** Optimization results ***
Optimal parallel alignment of
      - microscope X-axis with m-3m crystal hkl 011 and
      - microscope Z-axis with P63/mmc crystal uvw 330:
   -> Rotation around microscope X-axis: 8.3°
   -> Rotation around microscope Z-axis: 147.7°
   -> Deviation from ideal alignment in X-axis: 10.1 °
   -> Deviation from ideal alignment in Z-axis: 10.4 °
```

4 TROUBLESHOOTING

4.1 ERROR 'NUMBER OF AXIS-CRYSTALDIRECTION PAIRS OF ALIGNMENT OBJECTIVES NOT EQUAL'

You have probably given more microscope axes for alignment (axs.align) than number of crystal directions for alignment in (dir.ax) or vice versa. The number of alignment axes, crystal directions and symmetry flags in the initialization part 'Alignment Objective(s)' have to have equal dimensions as demonstrated in the examples.

4.2 ERROR 'NUMBER OF CRYSTALSYSTEM-ORIENTATION PAIRS NOT EQUAL'

In the initialization section 'General settings -> Crystal' it is important to list an equal amount of crystal structures (crys.cs) and crystal orientations (crys.o).

4.3 ERROR WHEN STARTING AND INITIALIZING MTEX

The script attempts to automatically start and initialize MTEX. If an error occurs try to restart MATLAB and start and initialize MTEX manually by executing 'startup_mtex' before running the *crystalAligne*r program.

4.4 OUTPUT 'LINEAR CONSTRAINTS ARE NOT SATISFIED WITHIN CONSTRAINT TOLERANCE'

This is an error that occurs for single-objective optimization (examples E1 and E4) and is related to an implementation of the *ga* algorithm in MATLAB versions older than R2016b (see textbox on page 1). From R2016b on the *ga* algorithm will strictly fulfil the set linear constraints, even if the achieved alignment is highly non-optimal. It likely means that you ran into a local minimum; while installing a newer version of MATLAB would be the best way to go a quick-and-dirty way of avoiding this is to increase the population size. This will slow down the code performance.

4.5 OTHER ISSUES

If there are other issues, comments or suggestions you would like to share with me you can contact me via contactnospam@fniessen.com (remove the nospam to make this email address work) and I will be happy to help.