CrystAligner v1.0 - Documentation

1 About

CrystAligner 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 *CrystAligner* 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 and the MATLAB global optimization toolbox. The global optimization toolbox often comes preinstalled with MATLAB or can be obtained separately. MTEX can be downloaded free of charge. The CrystAligner program has been tested with MTEX versions 4.2.1 to 5.1.1 and MATLAB version R2016b.

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 *CrystAligner.m* file is not in your MTEX folder, you will also have to add the path of your *CrystAligner.m* file to the MATLAB path in the same way. The program can be executed by entering 'CrystAligner' into the command window and pushing enter. It is however recommended to open the *CrystAligner.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.

E1: Alignment of a either of two equivalent crystal directions in a cubic crystal

Two equivalent crystal direction 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 eitherof the <111> or <100> equivalent 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 50 individuals. Optional text output for estimating the trench sizes for a FIB lift-out at an inclined surface and plotting of graphs 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.

```
% *** General settings ***
%Crystal
crys.o = [313 15 137];
crys.cs = {'cubic'};
crys.ss = 'orthorhombic';
%Alignment Objective(s)
axs.align = [zvector];
dir.ax{1} = [1 1 1; 1 0 0];
axs.sym   = [  1  ];
%SEM stage definition
axs.rot = [xvector; zvector];
optim.LB =[    0     -180  ];
optim.UB =[   20      180  ];
axs.invRot = [ 0       1   ];

% *** Genetic algorithm - optimization settings
%Genetic algorithm
optim.popSz = 50;
optim.funcTol = 0.01;
optim.maxStallGen = 10;
optim.iterOut = 0;
%Multiobjective genetic algorithm settings
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 = 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 directions and the z-axis of the microscope with any of the <110> equivalent 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 50 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.

```
% *** General settings ***
%Crystal
crys.o = [61 42 9];
```

```
crys.cs = {'cubic'};
crys.ss = 'orthorhombic';
%Alignment Objective(s)
axs.align = [yvector; zvector];
dir.ax{1} = [1 1 3];
dir.ax{2} =           [1 1 0];
axs.sym   = [   1        1  ];
%SEM stage definition
axs.rot = [xvector; zvector];
optim.LB =[    0     -180  ];
optim.UB =[   20      180  ];
axs.invRot = [ 0       1   ];

% *** Genetic algorithm - optimization settings
%Genetic algorithm
optim.popSz = 50;
optim.funcTol = 0.01;
optim.maxStallGen = 10;
optim.iterOut = 0;
%Multiobjective genetic algorithm settings
optim.wghtFac = [1,1];
optim.multiCore = 0;
optim.hybridFcn = 0;
optim.autoSol = 1;

% *** Optional - FIB liftout calculations
FIB.mode = 1;
FIB.trench.ang = 52;
FIB.trench.z = 15;
FIB.axs.tilt = 1;
FIB.axs.rot = 2;

% *** Output
optim.plot = 1;
```

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] axis 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.

```
% *** General settings ***
%Crystal
crys.o = [261 43 28; 175 20 102];
crys.cs = {'cubic','orthorhombic'};
crys.ss = 'orthorhombic';
%Alignment Objective(s)
axs.align = [xvector; zvector];
dir.ax{1} = [0 1 1];
```

```
dir.ax{2} =           [0 0 1];
axs.sym   = [   1       0  ];
%SEM stage definition
axs.rot = [xvector; zvector];
optim.LB =[    0     -180  ];
optim.UB =[   20      180  ];
axs.invRot = [ 0       1   ];

% *** Genetic algorithm - optimization settings
%Genetic algorithm
optim.popSz = 500;
optim.funcTol = 0.01;
optim.maxStallGen = 10;
optim.iterOut = 0;
%Multiobjective genetic algorithm settings
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;
```

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 (axs.rot). The rotation limits for this axis are defined as -45 – 45°. The additional degree of freedom compared to the setup in example E1 leads to perfect alignment of the <111> direction with the microscope z coordinate axis.

```
% *** General settings ***
%Crystal
crys.o = [313 15 137];
crys.cs = {'cubic'};
crys.ss = 'orthorhombic';
%Alignment Objective(s)
axs.align = [zvector];
dir.ax{1} = [1 1 1; 1 0 0];
axs.sym   = [  1  ];
%SEM stage definition
axs.rot = [xvector; yvector; zvector];
optim.LB =[    0       -45    -180  ];
optim.UB =[   20        45     180  ];
axs.invRot = [ 0        0       1   ];

% *** Genetic algorithm - optimization settings
%Genetic algorithm
```

```
optim.popSz = 50;
optim.funcTol = 0.01;
optim.maxStallGen = 10;
optim.iterOut = 0;
%Multiobjective genetic algorithm settings
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 = 0;
```

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 CrystAligner program.

4.4 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.