MATLAB package: Cell-Cycle-Chromatin-Dynamic-Organization, CCCDO, user guide

Part 0: installation of Code package folder

unzip “CCCDO_startup .m” and the folder CCCDO_mater to your MATLAB folder path, usually: “C:\ … \ MATLAB “

Part 1: Modify the start-up file

1. Open the file “CCCDO_startup .m”
2. For the first installation only change: “ `InstallRun = 1`”
   ```
   %% %%%%%%%%%% for first installation only change to %%%%%%%%%%%%%%
   InstallRun = 0;
   % InstallRun = 1;
   ```
   this action will create the folder tree required for analysis, as described in the part 2. Return: “`InstallRun = 0;`” immediately after installation.
3. Define directory were the data should be stored. In first installation the parent folder will be created in this folder.
   ```
   %% %%%%%%%%%%%%%%%%%% target folder for installation %%%%%%%%%%%%%
   global TargetFolder
   TargetFolder = ['Q:\Data and code from papers\2021 - Cell cycle
   dependent',...
      ' chromatin dynamic organization during the interphase'];
   cd(TargetFolder);
   %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
   ```
4. Define the MATLAB folder path,
   ```
   %% %%%%%%%%%%%%%%%%%%%% define MATLAB function folder %%%%%%%%%%%%
   global MATLABFolder
   MATLABFolder = 'C:\Users\naortal\Documents\MATLAB';
   %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
   ```

5. Define parent folder name. {“`ParentFolderName = 'ExampleFolder';`”} this folder will be created in the target folder user defined earlier.
   ```
   %% %%%%%%%%%% make Data storage folder if InstallRun is true
   %%%%%%%%%%%%%%%%
   if InstallRun==1
       % cell cycle phase names
       PhaseNames = {'G0','S','G2','LateG2','UT'};
       % cell lines names
       CellTypeName = {'MEF3T3','MEFLmnaKO'};
       % parent folder name
       ParentFolderName = 'ExampleFolder';
   ```

Part 2: User need to arrange the parent folder content, before starting running any analysis, as follow:

Name	Type	Name	Type
Level 1			
MEF3T3_G0	Experiment folder	MEFLmnaKO_G0	Experiment folder
MEF3T3_S	Experiment folder	MEFLmnaKO_S	Experiment folder
MEF3T3_G2	Experiment folder	MEFLmnaKO_G2	Experiment folder
MEF3T3_LateG2	Experiment folder	MEFLmnaKO_LateG2	Experiment folder
MEF3T3_UT	Experiment folder	MEFLmnaKO_UT	Experiment folder
EnsembleAnalysis	Analysis folder		
Level 2: content of an Experiment folder			
Analysis	Level 3 folder		
Level 4 folders: “ Cell_* ”, single cell data, includes csv files resulted from trajectory reconstruction, example file name for cell 6: “MEF3T3_S_Cell_6_*.csv”			
*_localizations.csv	File in Cell_*	Localization results: output of the DeepStorm3D net	
*_homographies.csv	File in Cell_*	Affinity matrixes, for each frame , which compensate for the lateral nucleus drift	
*_localizations_warped.csv	File in Cell_*	Compensated localization -after Affinity matrixes been applied on the raw localization	
*_trajectories.csv	File in Cell_*	Reconstructed trajectories of the compensated localization	
*_msds.csv	File in Cell_*	Mean square displacement vector (3D)	
Level 2: content of the folder “EnsembleAnalysis”			
Axial	Ensemble analysis based on the axial (Z) information		
Lateral	Ensemble analysis based on the lateral (XY) information		
Full	Ensemble analysis based on 3D information		
SummaryAnalysis	Constraining volume analysis		

Part 3: User parameters definition file

Open the file “UserParameter.m”

Part A: define general parameters for all experiment

1. Define the length of imaging (in frames)
2. Define the threshold for minimal length of a trajectory
3. Define the parent folder where all data and analysis results are saved. This folder should be arranged precisely as “Example folder” attached to this package

```
%% %%%%%%%%%%%%%%%%%%%% Part A: general parameters %%%%%%%%%%%%%%%%%%%%
% define number of frames in original image sequence
NumFrames = 1500;
% define threshold for minimal length of a trajectory
MinLength = 150;
% define the main folder for data storage and analysis
ParentFolder = ['Q:\Data and code from papers\',...
            '2021-Cell cycle dependent chromatin dynamic organization',...
            ' during the interphase\ExperimentalData'];
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

Part B: define the privet parameters for each experiment

Task indexes: 1,2

This part has 10 section: 2 cell lines (‘MEF 3T3’,’MEF LmnaKO’) and 5 phases (‘G0’,’S’,’G2’,’LateG2’,’UT’).
For each section user need to define:

1. Excluded cells (“`ExcludedCells`”): cells to be excluded from analysis. If none define =[];
2. Number of cells (“`numCells`”): total number of cells, including those need to be excluded.
3. Trajectories to be excluded (“`ExcludeTrj`”): after running QC analysis user may exclude manually specific trajectories due to not accurate linear fit. If none, as during the first run, leave the struct empty(“`ExcludeTrj=struct();`”)

```
%% %%%%%%%%%%%%%%%%%%%% MEF3T3 G0 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
                ExperimentFolder = [ParentFolder,'\MEF3T3_G0'];
                CellType='MEF3T3';
                ExcludedCells = [1:5,8:17,20:25,28,30,32];
                numCells=34;
                Phaseidx=1;
                %%%%%%%%%%% define here trj indexes to be excluded %%%%%%%%%
                ExcludeTrj=struct();
                ExcludeTrj.Cell_19 =[6];
                ExcludeTrj.Cell_26 =[5];
                ExcludeTrj.Cell_27 =[10];
                ExcludeTrj.Cell_33 =[6];
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

Part C: define folders for ensemble MSD analysis. **User do not need to edit this section!**
Task index: 3

```
%% %%%%%%%% Part C: define folders for ensemble MSD %%%%%%%%%%%%%%%
        % user do not need to change any parameter in this section!
```

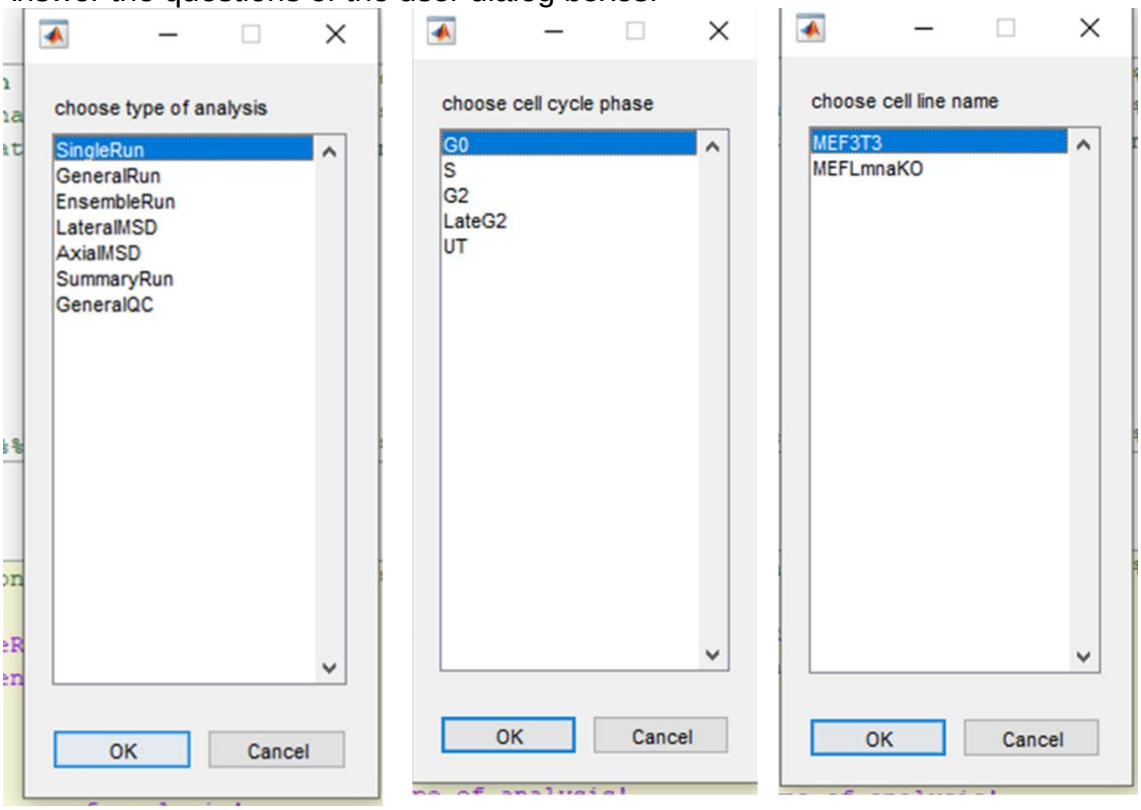
Part D: define time for constraining volume analysis
Task index: 4
User need to specify time in sec for calculate the constraining volume (“TimeAve = 25;”)

```
%% Part D: specify time for calculate the constraining volume %%%%%%%%%%%%%
        % time in sec
        TimeAve = 25;
        varargout{1}=TimeAve;
        % experiment folder
        ExperimentFolder = [ParentFolder,'\EnsembleAnalysis'];
        % redundant parameters are empty
        CellType=[];
        ExcludedCells=[];
        numCells=[];
        Phaseidx=[];
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% end task number 4 %%%%%%%%%%%%%%%%%%%%%%%%%%%
```

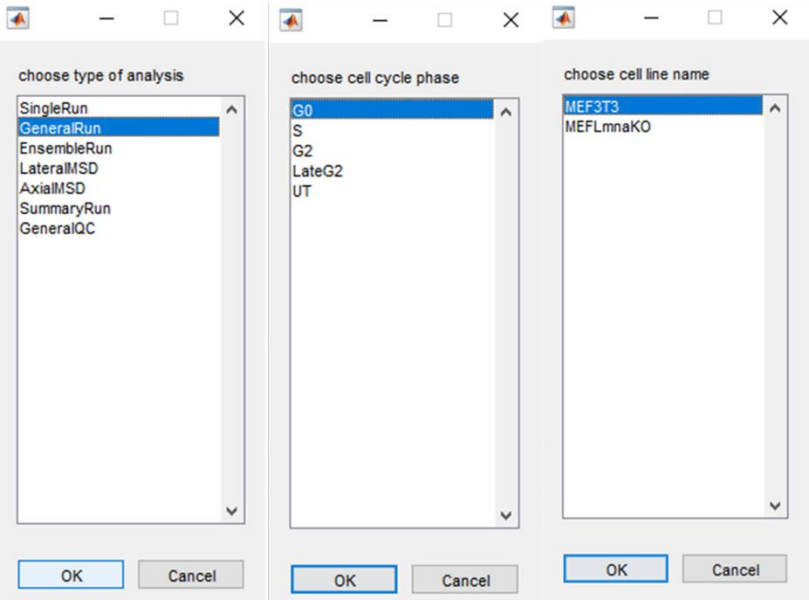
Part 4: analysis pipeline

Step0: save all CSV files with trajectories information as described in part 2.
Step 1: Run mode: `'SingleRun'`
Rearrange and save each cell data, for specific “experiment folder” (see part 2).
Run file: “CCCDO_startup.m” then run function: “Main.m”

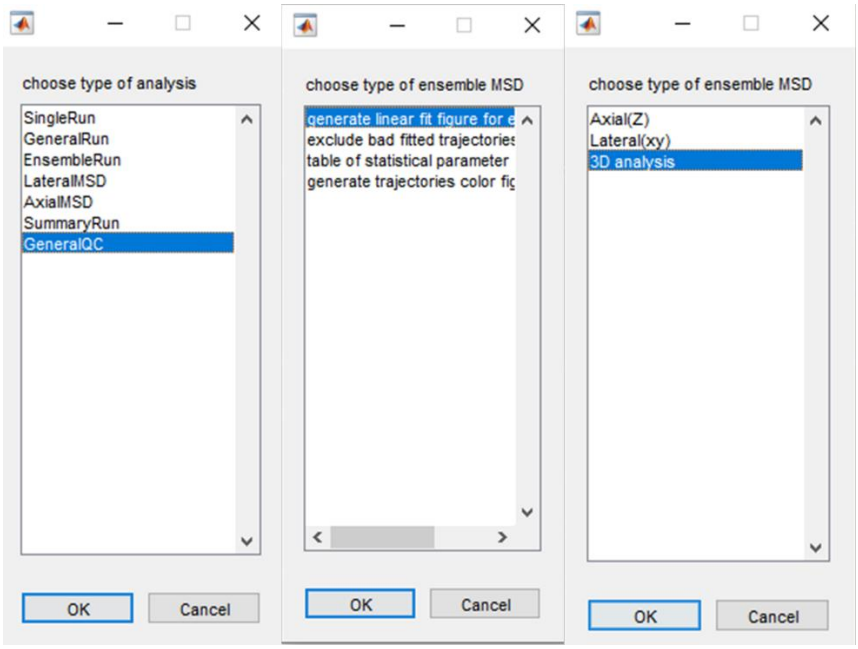
Answer the questions of the user dialog boxes:



Step 2: RunMode: `'GeneralRun'`
This function summarizes all trajectories data from single experiment. Run file: “CCCDO_startup.m” then run function: “Main.m”. Answer the questions of the user dialog boxes:



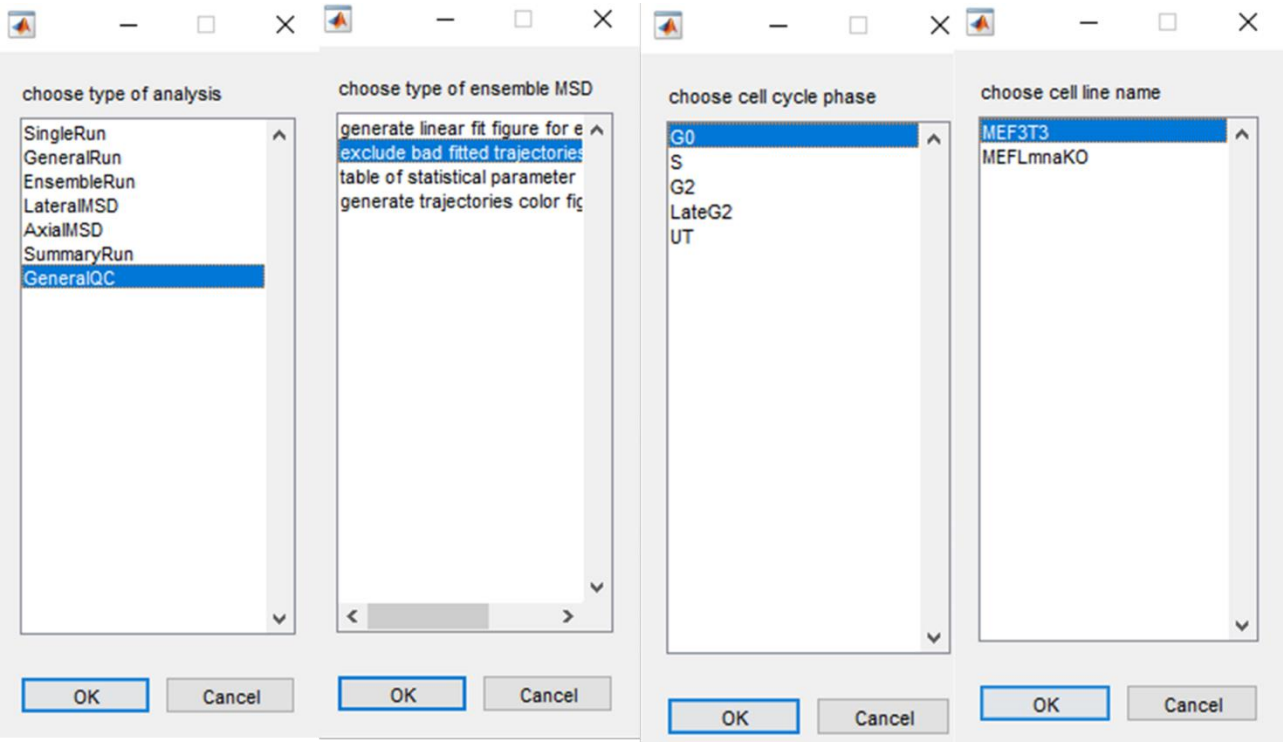
Step 2QC1: RunMode: `'GeneralQC'`
At this point user may run QC analysis which generate linear fit figures for all the trajectories in single experiment, in order to manually detect bad fitted trajectories and exclude them from analysis. Run file: “CCCDO_startup.m” then run function: “Main.m”. Answer the questions of the user dialog boxes:



Step 2QC2: RunMode: `'GeneralQC'`
After running Step 2QC1, user should update the file: "`UserParameter.m`" and specify the trajectories to be excluded as follow:

```
%%%%%%%%%% define here trj indexes to be excluded %%%%%%%%%
                ExcludeTrj=struct();
                ExcludeTrj.Cell_19 =[6];
                ExcludeTrj.Cell_26 =[5];
                ExcludeTrj.Cell_27 =[10];
                ExcludeTrj.Cell_33 =[6];
```

Then run Main.m to remove these trajectories from analysis; answer the dialog boxes questions:

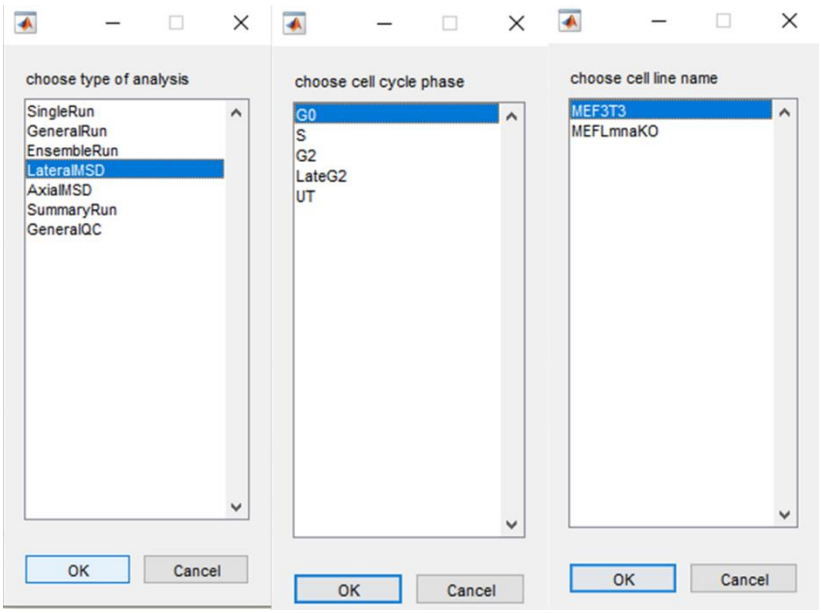


Step 2QC3: RunMode: `'GeneralRun'`

When QC procedures are done. Run again general analysis.

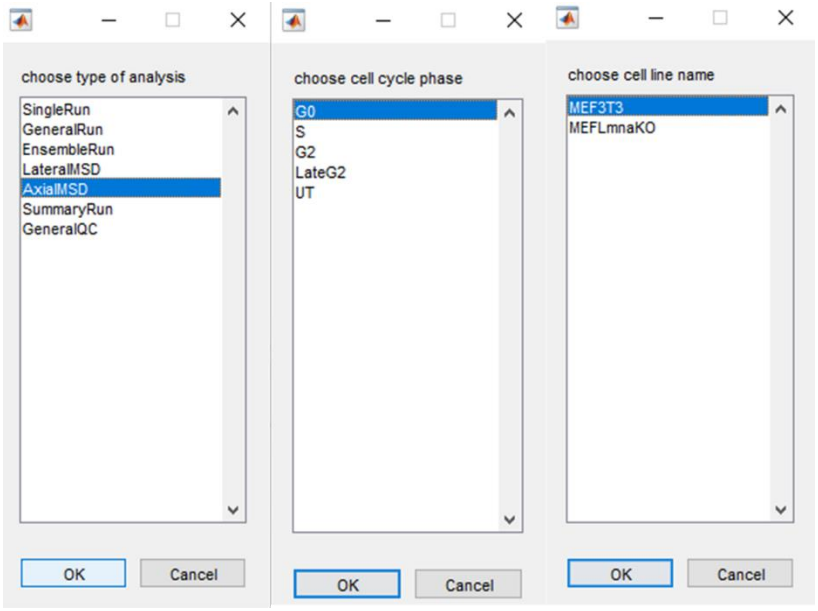
Step 3a: RunMode: `'LateralMSD'`

This function summarizes all trajectories data from single experiment, using only XY coordinates. Run file: “CCCDO_startup.m” then run function: “Main.m”. Answer the questions of the user dialog boxes:

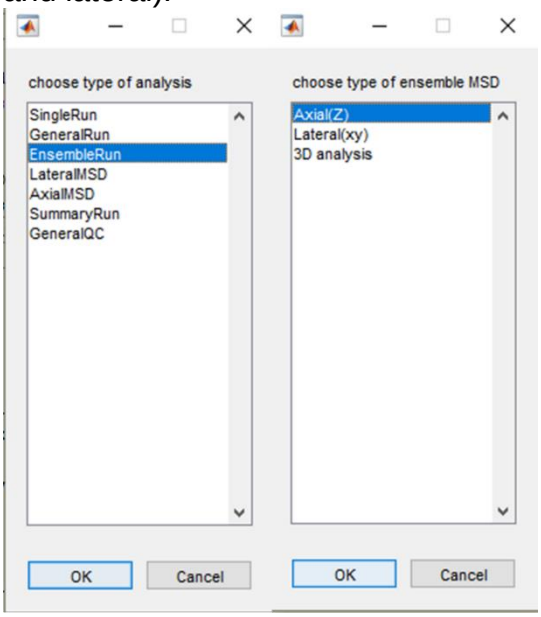


Step 3b: RunMode: `'AxialMSD'`

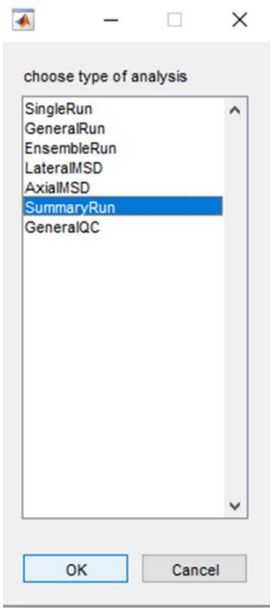
This function summarizes all trajectories data from single experiment, using only Z coordinates. Run file: “CCCDO_startup.m” then run function: “Main.m”. Answer the questions of the user dialog boxes:



Step 4: RunMode: `'EnsembleRun'`
calculates anomalous exponent and diffusion coefficient.
compare between cell cycle phase and between cell lines (MEF 3T3 and MEFLmnaKO)
user shall run this section 3 times for each type of ensemble MSD analysis (3D, axial and lateral).



Step 5: RunMode: `'SummaryRun'`
This function calculates the " constraining volume"- the mean volume "visited" by a random telomere in each phase of the cell cycle. This value represents the constraint level of the specific cell cycle phase.



Step 6: RunMode: `'GeneralQC'`
Final step is statistical tests for all analyses. This run will calculate the results significance using two-sample t-test, under the assumption that different variance is allowed. In addition, a table similar to supplementary table S2 in our article will be printed in MATLAB command line.