

Stepfinder Manual

Jacob Kerssemakers

March 2018

Contents

Stepfinder Manual.....	1
Introduction.....	1
Notes on Development version.....	1
Package Contents	1
Step-by-Step-Use of the Stepfinder	2
Basic Use.....	2
Data In/Out.....	2
Run settings	3
Advanced options.....	4
Auxiliary code	4

Introduction

This manual describes content and workings of a ‘Step finder’ code. This code is an improved version of the original algorithm described in: **‘Assembly dynamics of microtubules at molecular resolution’**, Jacob W. J. Kerssemakers, E. Laura Munteanu, Liedewij Laan, Tim L. Noetzel, Marcel E. Janson and Marileen Dogterom, Nature. 2006 Aug 10;442(7103):709-12)

Notes on Development version

March 2018: “advanced settings” and “fittings” tab still crash

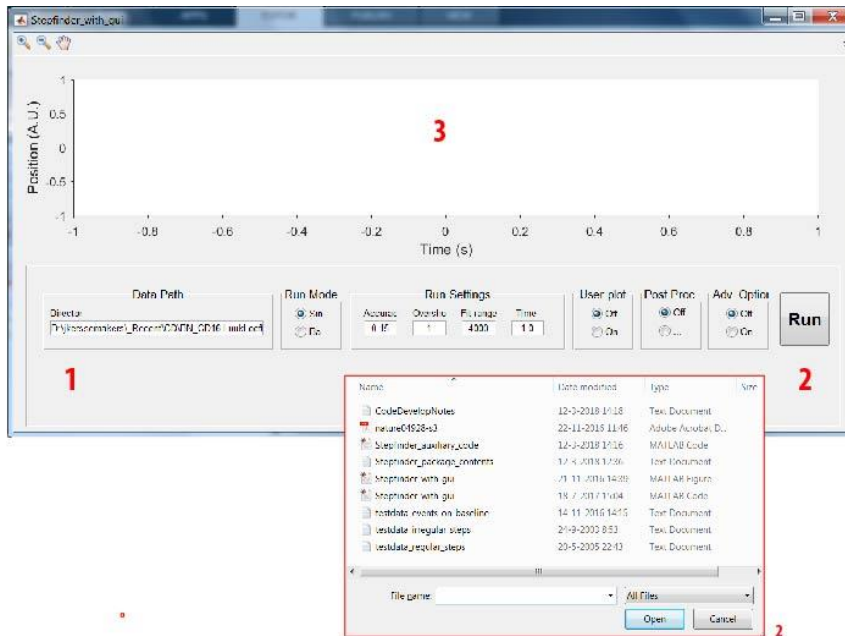
Package Contents

- Stepfinder_with_gui.m/ .fig: MATLAB code and gui of the stepfinder program
- testdata_events_on_baseline.txt; testdata_regular_steps.txt; testdata_irregular_steps.txt’: various types of testdata showing the workings and input format of the code
- Stepfinder_auxiliary_code.m : code containing extra functions
- nature04928-s3.pdf: pdf file with theory as described in the original paper:
- Kerssemakers et al, Nature. 2006 Aug 10;442(7103):709-12)

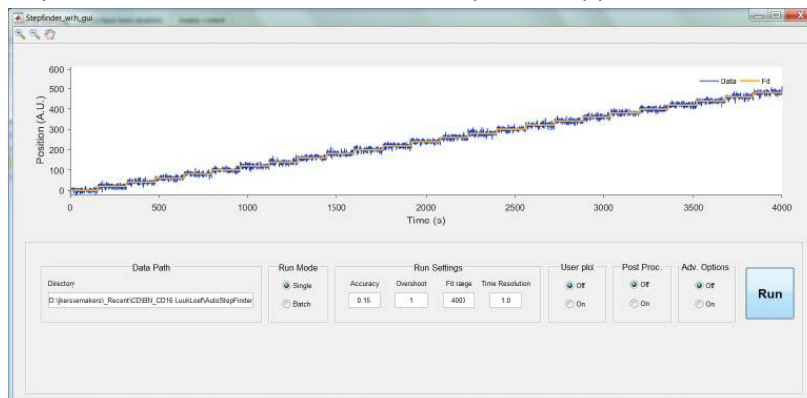
Step-by-Step-Use of the Stepfinder

Basic Use

Open the 'Stepfinder_with_gui' in MATLAB and run it. A User interface will appear with the main run settings set to their default values.



1. Here, set the path towards input data (default is the location of the code and testfiles)
2. Press 'Run' ; select a data file. For the demo, choose "testdata_regular_steps" . A representation of the data and the stepfit will appear here.



Data In/Out

Input Data should be 2-column (Time, Signal). Note: the time units will NOT be used for processing or output; These are user set by the button 'time resolution'.

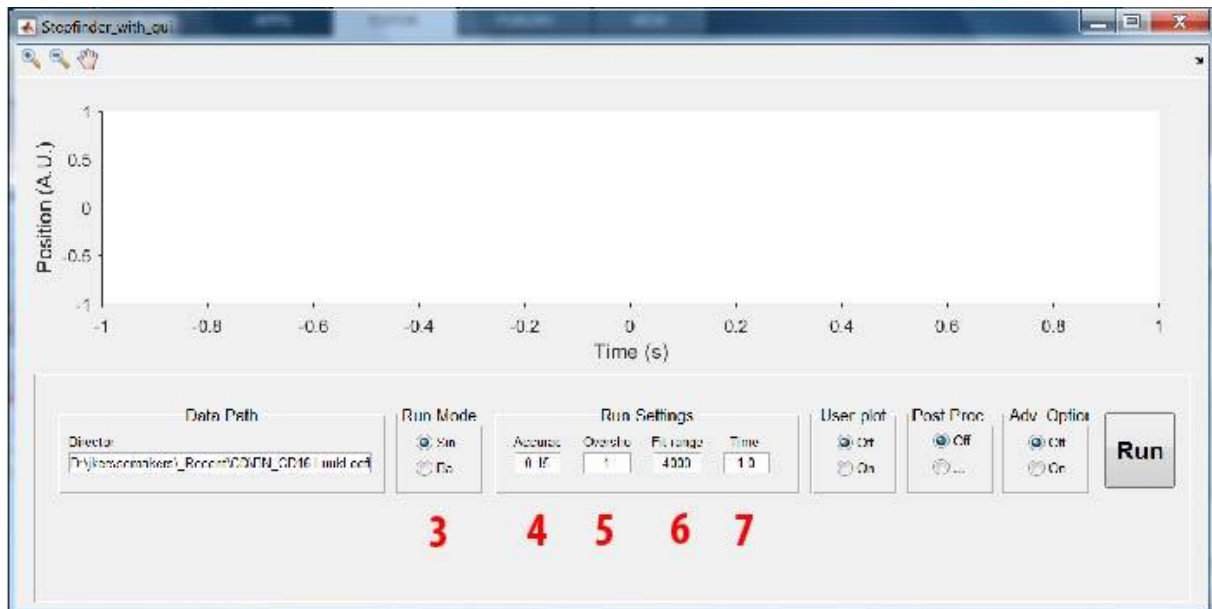
Output data is auto-saved in a local directory; here "testdata_regular_steps_Fitting_Result". It contains:

- 1) testdata_regular_steps_fits: columns original time, original data, step fit as shown in the GUI screen

- 2) testdata_regular_steps_properties: columns with properties of individual steps.
 - a. IndexStep: the data index of the step (defined as the rightmost point of the preceding plateau)
 - b. TimeStep: the time associated with this point, taken as (number of datapoints times time-resolution)
 - c. LevelBefore: the average of the plateau preceding the step
 - d. LevelAfter: the average of the plateau following the step
 - e. StepSize: the stepsize, defined as the value difference between the adjacent plateaus
 - f. DwellTimeStepBefore: time interval covered by the preceding plateau
 - g. DwellTimeStepAfter: time interval covered by the following plateau
 - h. StepError: the error associated with the step size

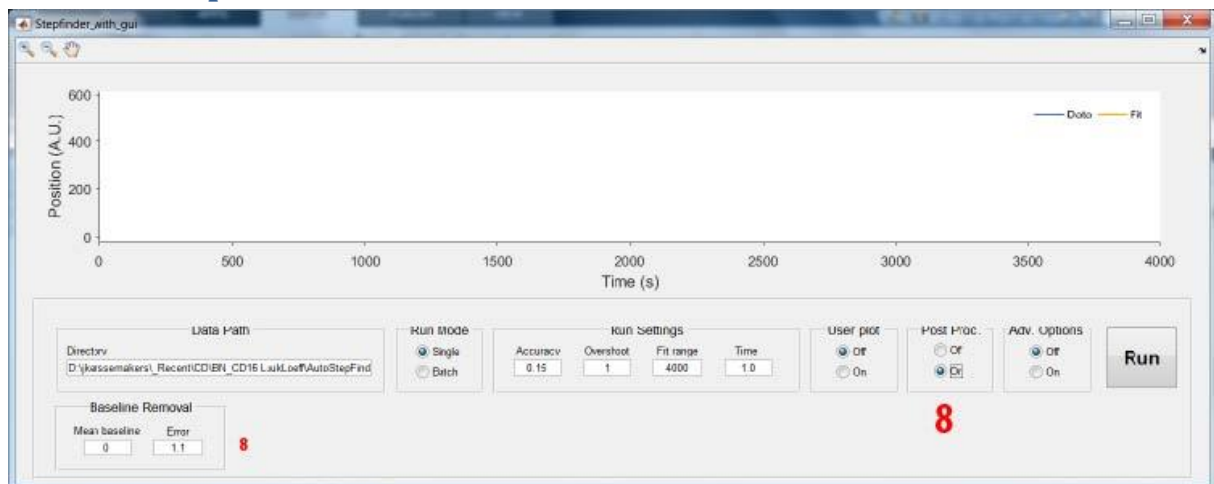
- 3) testdata_regular_steps_SCurve: columns with the ‘S-curves’ produced during the fitting procedure (see reference text).

Run settings

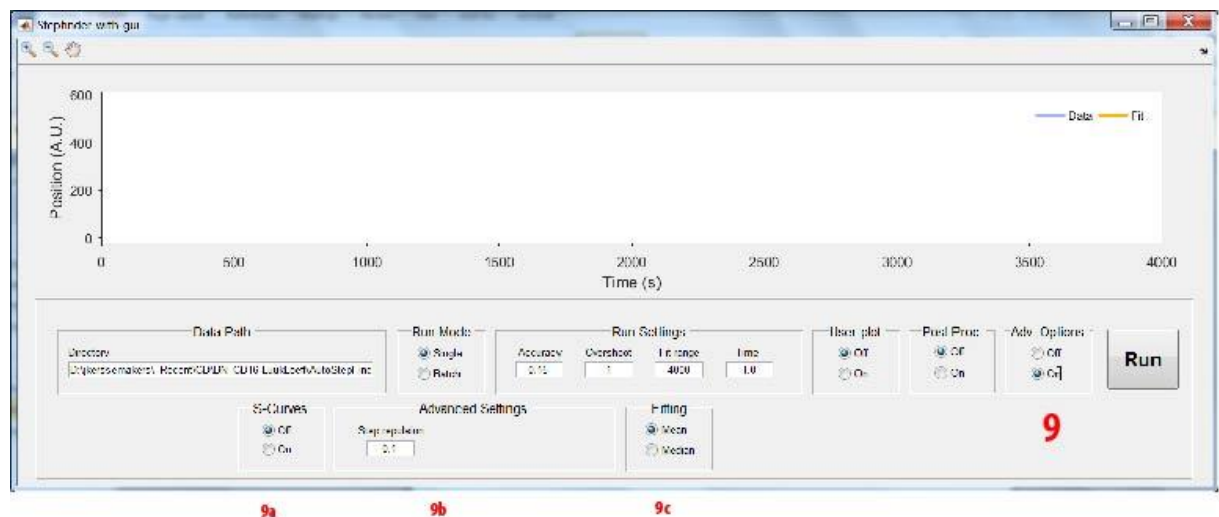


3. When setting this button to ‘batch mode’, the user can select a directory containing multiple text files, which the code will analyze in batch style.
4. “Accuracy” determines how far the code will fit steps into the noise by setting the “S-value” limits; see reference text for a more detailed explanation
5. “Overshoot” : forcing a factor ...more steps than the fitting algorithm itself found as the optimal number. This setting normally be left at 1, unless the user has other criteria for determining the optimal step number (such a known, fixed number of steps in the data).
6. “Fit range” : this sets the maximum number of steps to be fit; use it for large datasets with limited step numbers, to save computing time.
7. “Time” this sets the time resolution, i.e. the time interval between each data point. This will be used for time data in the output, the original time axis is ignored.

Advanced options



8. Pressing “Post Proc.” Shows a tab for removing a baseline from the fit in post-processing



9. Pressing ‘Advanced ’ options shwos tabs for advanced use. It is assumed the user has an understanding of the workings of the algorithm.
- Show the ‘S-curves’ after fitting.
 - ‘Step repulsion’ puts a limit on very short plateaus during fitting. Normally, it should not be changed.
 - Optionally, one can determine plateaus for the final fit using the median of plateaus, not the average. The fit procedure itself always uses averages.

Auxiliary code

Stepfinder_auxiliary_code.m contains extra functions for custom Matlab use. It is assumed the user can edit in Matalab and has a good understanding of the workings of the algorithm.

- StepfinderCoreCopy with sub-functions [line11-250]: this section contains the “core part” of the stepfinder code. It can be run separately in single-run mode and in demo mode using

the included 'BottomUpTraceBuilder' function, that will produce a trace containing nested steps at widely different sizes.

- (StepInjection)

References

1. Thesis Luuk_Loeff Chapter 6: thesis chapter describing the current version of the algorithm
2. **Assembly dynamics of microtubules at molecular resolution'**, Jacob W. J. Kerssemakers, E. Laura Munteanu, Liedewij Laan, Tim L. Noetzel, Marcel E. Janson and Marileen Dogterom, Nature. 2006 Aug 10;442(7103):709-12)