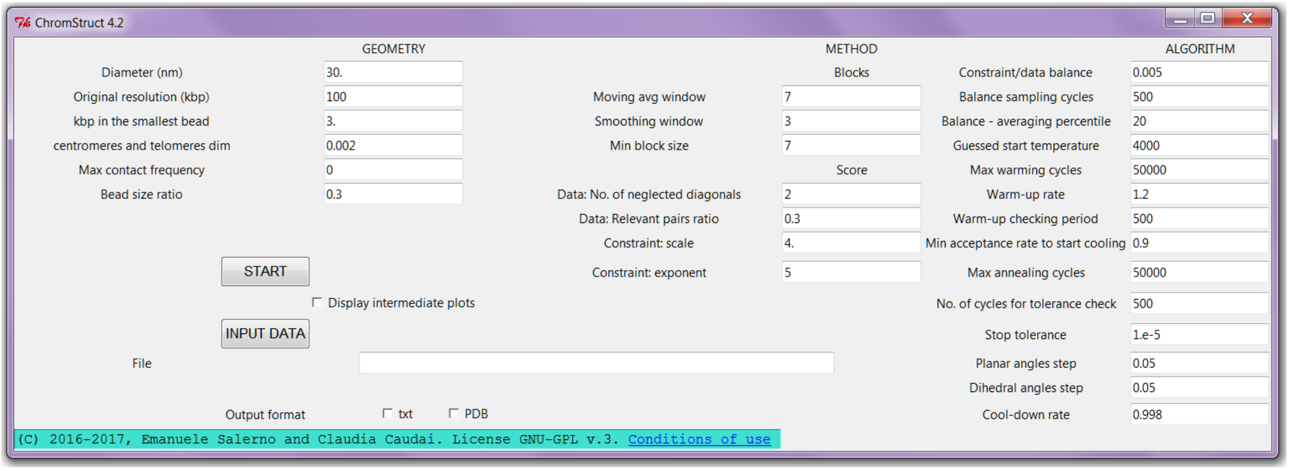
CHROMSTRUCT 4.2 folder: README

This folder contains 3 files:

- `ChromStruct_4.2_GUI.py` (GUI version of CHROMSTRUCT 4.2)
- `Plot_Energy_Chain_2.0.py` (commandline code to display CHROMSTRUCT results)
- `README.pdf` (this file)

The code files are all self-contained and only need a Python interpreter to run.

The ChromStruct GUI appears as shown below. All the parameters can be edited before starting the program.



GEOMETRY
Diameter (nm) 30.
Original resolution (kbp) 100
kbp in the smallest bead 3.
centromeres and telomeres dim 0.002
Max contact frequency 0
Bead size ratio 0.3

□ Display intermediate plots
File
Output format □ txt □ PDB

METHOD
Blocks
Moving avg window 7
Smoothing window 3
Min block size 7
Score
Data: No. of neglected diagonals 2
Data: Relevant pairs ratio 0.3
Constraint: scale 4.
Constraint: exponent 5

ALGORITHM
Constraint/data balance 0.005
Balance sampling cycles 500
Balance - averaging percentile 20
Guessed start temperature 4000
Max warming cycles 50000
Warm-up rate 1.2
Warm-up checking period 500
Min acceptance rate to start cooling 0.9
Max annealing cycles 50000
No. of cycles for tolerance check 500
Stop tolerance 1.e-5
Planar angles step 0.05
Dihedral angles step 0.05
Cool-down rate 0.998

`(C) 2016-2017, Emanuele Salerno and Claudia Caudai. License GNU-GPL v.3.` Conditions of use

Three groups of quantities are displayed: the first includes geometrical features, the second sets up the TAD extraction and the score function, and the third is only related to the simulated annealing algorithm. We chose to make all the parameters available, but in normal use only a few of them need to be tuned:

- GEOMETRY - Original resolution: this is the genomic resolution of the data to be treated.
  GEOMETRY - Max contact frequency: if set to zero (the default), its value is computed as the maximum of the data matrix. In some cases, for example when different segments of the same chain are being estimated separately, it could be appropriate to set it to some fixed, user-defined value.
- GEOMETRY - Bead size ratio: this allows the user to tune the flexibility of the output chain. The adequacy of its choice can be evaluated *a posteriori*, by considering the biological plausibility of the output.
- METHOD - Blocks - Min block size: this can prevent the program from working with too small submatrices. Use cautiously.

<u>Changing the other parameters can influence the performance of the algorithm in a complicated way. We recommend to be careful when acting on them. To fully understand their significance, please refer to the commented source code and to the table at the end of this text.</u>

The GUI code can be run from the interactive python dialog or from the console window, by invoking

```
python ChromStruct_4.2_GUI.py
```

After setting the parameters, type the data file name in the File field, or press the INPUT DATA button to choose the data file from the file system; its complete path then appears in the File field. Then, to start the algorithm, press the START button. If Display intermediate plots is checked, for each block, the program displays the plots of the score function values during the annealing, the number of accepted versus proposed updates during the annealing, and the estimated 3D structure of the subchain mapped onto the current data block. To continue execution, the user must first close all the graphical windows.

If the data file is

```
filename.suffix
```

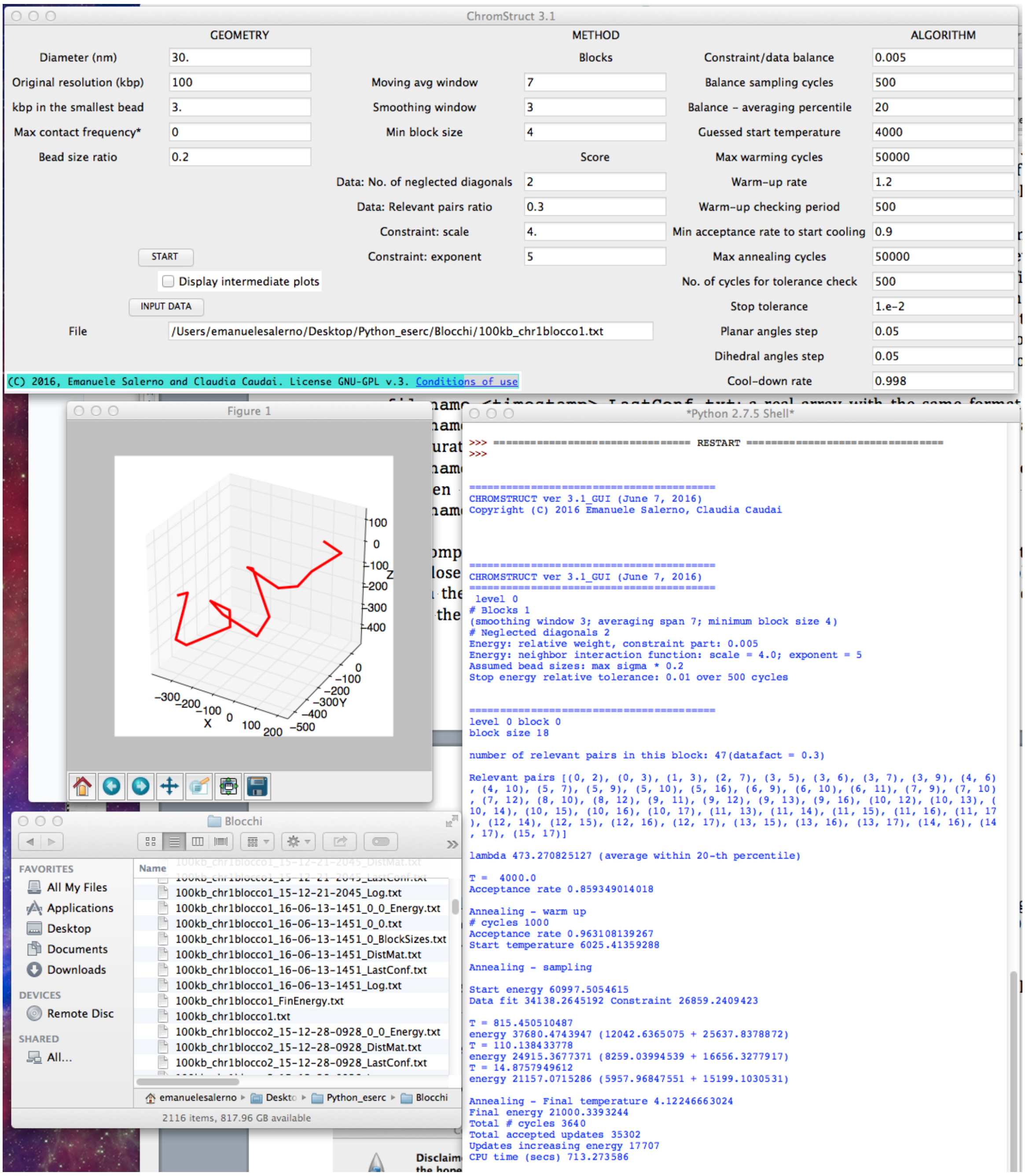
the code produces a number of files with these names:

- `filename_<timestamp>_<level>_BlockSizes.txt`: a list with as many entries as blocks detected at resolution level `<level>`. `<level>` is coded as an integer from 0 to *number of detected levels – 1*. `<timestamp>`, a character string formatted as <`yy-mm-dd-hhmm`>, is referred to the date and time when the algorithm starts, and is used to identify the files coming from the same data file and the same run.
- `filename_<timestamp>_Log.txt`: a self-explanatory logfile.
- `filename_<timestamp>_<level>_<block>_Energy.txt`: a real array with 3 columns and as many rows as accepted annealing updates of the configuration of block `<block>` (coded as an integer from 0 to *number of detected blocks at level* `<level>` *– 1*) at the resolution level `<level>`. The first real in each row is the data fit part of the score function, the second is the constraint part, and the third is the total score. If the related checkbox in the GUI is active, this is plotted as soon as the iteration at each block and level is complete. It can also be plotted by `Plot_Energy_Chain_2.0.py`.
- `filename_<timestamp>_<level>_<block>.txt`: a real array with 4 columns and as many rows as three times the number of beads in block `<block>` at level `<level>`. The first of each three rows contains the coordinates (in nm) of the first endpoint of a bead; the second contains the coordinated of the centroid, and the third contains the coordinates of the second endpoint. Each row is completed with the estimated size (in nm) of the related bead. If the related checkbox in the GUI is active, this structure is plotted as soon as block `<block>` at level `<level>` has been computed. It can also be plotted by `Plot_Energy_Chain_2.0.py`.
- `filename_<timestamp>_LastConf.txt`: a real array with the same format as `filename_<timestamp>_<level>_<block>.txt`, with the final 3D chain configuration. This is plotted at the end of the procedure. It can also be plotted by `Plot_Energy_Chain_2.0.py`.
- `filename_<timestamp>_DistMat.txt`: a real array with the mutual distances between bead centroids, computed from the final estimated structure in `filename_<timestamp>_LastConf.txt`.

The code also prints the logfile information in the console window (score values and related annealing temperature once in every 1000 cycles). To close the program after the final plot, close the plot window and then the graphical interface. To abort the program, press <ctrl>-c from the keyboard. `ChromStruct_4.2_GUI.py` displays the 3D coordinates of the bead centroids linked by a red ploygonal line. `Plot_Energy_Chain_2.0.py`, conversely, links the centroids with a smooth curve, obtained by cubic spline interpolation.

The two Output format checkboxes, txt and PDB, offer the possibility of choosing the format of the final estimated chain: if none or txt is selected, the only file `filename_<timestamp>_LastConf.txt` is stored. If `PDB` is selected, the file `filename_<timestamp>_LastConf.pdb` is stored, which can then be visualized by the standard software packages accepting the PDB format.

The picture below is a screenshot taken during a computation of CHROMSTRUCT version 3.1 (externally, no substantial difference exists with version 4.2): Top: the CHROMSTRUCT GUI; Bottom - right: the Python console window; left: the 3D plot of the reconstructed structure, and the directory list with the data file and the output files, with `<timestamp>` = `16-06-13-1451`.



ChromStruct 3.1

GEOMETRY

Diameter (nm) 30.
Original resolution (kbp) 100
kbp in the smallest bead 3.
Max contact frequency* 0
Bead size ratio 0.2

METHOD

Blocks

Moving avg window 7
Smoothing window 3
Min block size 4

Score

Data: No. of neglected diagonals 2
Data: Relevant pairs ratio 0.3
Constraint: scale 4.
Constraint: exponent 5

ALGORITHM

Constraint/data balance 0.005
Balance sampling cycles 500
Balance – averaging percentile 20
Guessed start temperature 4000
Max warming cycles 50000
Warm–up rate 1.2
Warm–up checking period 500
Min acceptance rate to start cooling 0.9
Max annealing cycles 50000
No. of cycles for tolerance check 500
Stop tolerance 1.e–2
Planar angles step 0.05
Dihedral angles step 0.05
Cool–down rate 0.998

File /Users/emanuelesalerno/Desktop/Python_eserc/Blocchi/100kb_chr1blocco1.txt

(C) 2016, Emanuele Salerno and Claudia Caudai. License GNU-GPL v.3. Conditions of use

Python 2.7.5 Shell

```
>>> =============================== RESTART ===============================
>>>


========================================
CHROMSTRUCT ver 3.1_GUI (June 7, 2016)
Copyright (C) 2016 Emanuele Salerno, Claudia Caudai



========================================
CHROMSTRUCT ver 3.1_GUI (June 7, 2016)
========================================
 level 0
# Blocks 1
(smoothing window 3; averaging span 7; minimum block size 4)
# Neglected diagonals 2
Energy: relative weight, constraint part: 0.005
Energy: neighbor interaction function: scale = 4.0; exponent = 5
Assumed bead sizes: max sigma * 0.2
Stop energy relative tolerance: 0.01 over 500 cycles


========================================
level 0 block 0
block size 18

number of relevant pairs in this block: 47(datafact = 0.3)

Relevant pairs [(0, 2), (0, 3), (1, 3), (2, 7), (3, 5), (3, 6), (3, 7), (3, 9), (4, 6)
, (4, 10), (5, 7), (5, 9), (5, 10), (5, 16), (6, 9), (6, 10), (6, 11), (7, 9), (7, 10)
, (7, 12), (8, 10), (8, 12), (9, 11), (9, 12), (9, 13), (9, 16), (10, 12), (10, 13), (
10, 14), (10, 15), (10, 16), (10, 17), (11, 13), (11, 14), (11, 15), (11, 16), (11, 17
), (12, 14), (12, 15), (12, 16), (12, 17), (13, 15), (13, 16), (13, 17), (14, 16), (14
, 17), (15, 17)]

lambda 473.270825127 (average within 20-th percentile)

T =  4000.0
Acceptance rate 0.859349014018

Annealing - warm up
# cycles 1000
Acceptance rate 0.963108139267
Start temperature 6025.41359288

Annealing - sampling

Start energy 60997.5054615
Data fit 34138.2645192 Constraint 26859.2409423

T = 815.450510487
energy 37680.4743947 (12042.6365075 + 25637.8378872)
T = 110.138433778
energy 24915.3677371 (8259.03994539 + 16656.3277917)
T = 14.8757949612
energy 21157.0715286 (5957.96847551 + 15199.1030531)

Annealing - Final temperature 4.12246663024
Final energy 21000.3393244
Total # cycles 3640
Total accepted updates 35302
Updates increasing energy 17707
CPU time (secs) 713.273586
```

Blocchi

100kb_chr1blocco1_15-12-21-2045_Log.txt
100kb_chr1blocco1_16-06-13-1451_0_0_Energy.txt
100kb_chr1blocco1_16-06-13-1451_0_0.txt
100kb_chr1blocco1_16-06-13-1451_0_BlockSizes.txt
100kb_chr1blocco1_16-06-13-1451_DistMat.txt
100kb_chr1blocco1_16-06-13-1451_LastConf.txt
100kb_chr1blocco1_16-06-13-1451_Log.txt
100kb_chr1blocco1_FinEnergy.txt
100kb_chr1blocco1.txt
100kb_chr1blocco2_15-12-28-0928_0_0_Energy.txt
100kb_chr1blocco2_15-12-28-0928_DistMat.txt
100kb_chr1blocco2_15-12-28-0928_LastConf.txt

emanuelesalerno ▸ Deskto ▸ Python_eserc ▸ Blocchi

2116 items, 817.96 GB available

Table of input parameters

As denoted in GUI	Related variable	Description
GEOMETRY		
Diameter (nm)	DIA	Assumed diameter of the chromatin fiber
Original resolution (kbp)	RIS	Genomic size of a single locus in the original data matrix
kbp in the smallest bead	NB	Genomic length of a DNA chain with physical length DIA
centromeres and telomeres dim	crate	Compactness rate for centromeres and telomeres.
Max contact frequency	NMAX	Maximum entry in the data matrix. The default, zero, lets che code compute the value from the input data; in particular cases, for example when a chain is estimated in separate segments, the user may want to set a unique value. This is used to assign approximate sizes to the smallest-scale beads
Bead size ratio	extrate	Fraction of the largest principal component of a subchain centroids coordinates used to assign an approximate size to the equivalent coarser-scale bead
METHOD		
Moving avg window	span	Size of the triangular submatrix used to compute the moving average of the contact frequencies off the main diagonal of the data matrix
Smoothing window	window	Size of the window used to smooth the moving average function
Min block size	minsize	Minimum size accepted for any diagonal block extracted from the data matrix
Data: No. of neglected diagonals	diagneg	Number of subdiagonals (including the main diagonal) in the data matrix to be excluded from the population of the in-contact pairs set
Data: Relavant pairs ratio	datafact	Contact frequency percentile to be exceeded by any bead pair to be included in the in-contact pairs set
Constraint: scale	scale	Scale factor *c* in the constraint part of the score function (see reference [3])
Constraint: exponent	exponent	Exponent *b* in the constraint part of the score function. It must be an odd integer (see reference [3])
ALGORITHM		
Constraint/data balance	regulenergy	A factor to balance equally the influence of data and prior knowledge in all the blocks and all scales. It sets the appropriate value for λ in all the annealing cycles. It is easily determined by trial and error. If it is not very different from its defaul value, it is not particularly critical (see reference [3])
Balance sampling cycles	avgenergy	Number of random configurations used to determine λ statistically. Normally, it does not need to be tuned.
Balance -averaging percentile	percenergy	Score percentile used to select the random samples used to compute λ
Guessed start temperature	Tmax	Initial temperature set to start the annealing cycles. A too small or too large value have just the effect of slowing down the estimation
Max warming cycles	itwarm	Maximum number of cycles performed to evaluate the appropriate start temperature. The actual number needed is always much smaller than the default value. If the default itwarm is reached, consider looking for something wrong with the data or the parameters
Warm-up rate	incrtemp	Parameter used to increase the temperature until the actual annealing can start. Normally, it does not need to be tuned

Warm-up checking period	checkwarm	Number of periods used to check whether the right start temperature for annealing has been reached. This does not need to be tuned
Min acceptance rate to start cooling	muwarm	Minimum ratio between the accepted and proposed transitions to be reached to start annealing. The default 0.9 does not need to be altered
Max annealing cycles	itmax	Maximum allowed number of annealing cycles. The actual number needed to satisfy the stop criterion is always much smaller than the default value.
No. of cycles for tolerance check	itstop	Cardinality of the consecutive accepted solutions set with final scores within the stop tolerance (see reference [3]). If not very different from the default, this is not a critical parameter
Stop tolerance	stoptol	Stop tolerance (see row above)
Planar angles step	RANDPLA	Maximum random increment to be assigned to the planar angle between any two adjacent beads
Dihedral angles step	RANDDIE	Maximum random increment to be assigned to the dihedral angle between any two adjacent beads
Cool-down rate	decrtemp	Parameter used to decrease the temperature during the annealing cycles: T(n)=decrtemp*T(n-1). It is not safe to make this parameter much smaller than its default

References

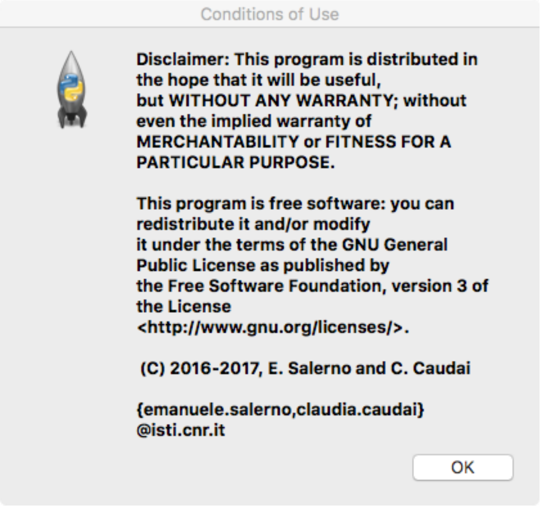
[1] C. Caudai, E. Salerno, M. Zoppè, A. Tonazzini, "Inferring 3D chromatin structure using a multiscale approach based on quaternions", *BMC Bioinformatics*, Vol. 16, 234, 2015, DOI: 10.1186/s12859-015-0667-0.

[2] C. Caudai, E. Salerno, M. Zoppè, A. Tonazzini, "Estimation of the Spatial Chromatin Structure Based on a Multiresolution Bead-Chain Model", *IEEE-ACM Trans. Comp. Biol. Bioinf.*, 2018, to appear, DOI: 10.1109/TCBB.2018.2791439.

[3] C. Caudai, E. Salerno, M. Zoppè, I. Merelli, A. Tonazzini, "ChromStruct 4: A Python Code to Estimate the Chromatin Structure from Hi-C Data", *IEEE-ACM Trans. Comp. Biol. Bioinf.*, 2017, to appear, DOI: 10.1109/TCBB.2018.2838669.

Conditions of use

Please refer to GNU-GPL version 3 or later: <http://www.gnu.org/licenses/gpl-3.0.html>



Disclaimer: This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, version 3 of the License <http://www.gnu.org/licenses/>.

(C) 2016-2017, E. Salerno and C. Caudai

{emanuele.salerno,claudia.caudai}@isti.cnr.it