Bioinformatics
doi.10.1093/bioinformatics/xxxxxx
Advance Access Publication Date: Day Month Year
Application note

Subject Section

scool: A new data storage format for single-cell Hi-C data

Joachim Wolff [1,*], Nezar Abdennur [2], Rolf Backofen [1,3], Björn Grüning [1]

[1] Bioinformatics Group, Department of Computer Science, University of Freiburg, Georges-Köhler-Allee 106, 79110 Freiburg, Germany
[2] Institute for Medical Engineering and Science, Massachusetts Institute of Technology, Cambridge, MA, 02139, USA
[3] Signalling Research Centres BIOSS and CIBSS, University of Freiburg, Schänzlestr. 18, 79104 Freiburg, Germany

*To whom correspondence should be addressed.

Associate Editor: XXXXXXX

Received on XXXXX; revised on XXXXX; accepted on XXXXX

Abstract

Motivation: Single-cell Hi-C research currently lacks an efficient, easy to use and shareable data storage format. Recent studies have used a variety of sub-optimal solutions: publishing raw data only, text based interaction matrices, or reusing established Hi-C storage formats for single interaction matrices. These approaches are storage and pre-processing intensive, require long labour time and are often error-prone.
Results: The single-cell cooler file format (*scool*) provides an efficient, user-friendly and storage-saving approach for single-cell Hi-C data. It is a flavour of the established cooler format and guarantees stable API support.
Availability: The single-cell cooler format is part of the cooler file format as of API version 0.8.9. It is available via pip, conda and github: `https://github.com/mirnylab/cooler`.
Contact: wolffj@informatik.uni-freiburg.de
Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 Introduction

The storage, processing and analysis of single-cell Hi-C data face several challenges. First, the pre-processing overhead for single-cell Hi-C is both storage-intensive and time-consuming. For example, reproducing the results of the Nagano *et al.* (2017) single-cell Hi-C study requires downloading, demultiplexing and mapping more than 1.1 TB of compressed raw FASTQ data and creating thousands of interaction matrices. Second, manually handling so many files is unwieldy and prone to error. For example, some studies (Nagano *et al.* (2013); Stevens *et al.* (2017); Ramani *et al.* (2017)) have published their pre-processed data as text-based files. Depending on the resolution, these files potentially store millions to billions of features in an uncompressed text file without fast random or partial access. By contrast, studies like Gassler *et al.* (2017) published pre-processed *cool* files (Abdennur and Mirny (2019)) for each cell and at multiple resolutions. However, due to redundancy in data storage and the complexitiy of handling a proliferation of files, this one-matrix-per-file approach has limited scalability and makes reproducible analysis challenging.

Here, we present the single-cell cooler format, a "flavour" of the cooler file format (Abdennur and Mirny (2019)), that stores multiple single-cell sparse Hi-C interaction matrices at a common resolution in a single HDF5 (Koziol and Robinson (2018)) file, allowing portable, space-efficient and fast access to single-cell interaction data. It uses the recommended extension *.scool*.

2 Methods

We adopt the basic structure of the cooler format to create a collection of single-cell interaction matrices having common dimensions (see Figure 1 A and B). Internally, all single-cell interaction matrices are stored under a group `/cells` and each matrix is identified by a unique cell ID and has the structure of a standard cooler *data collection* (Figure 1 A), allowing it to be read independently and transparently with the regular cooler API (see Listing 2). However, to eliminate redundancy, data structures that are shared between all cells are implemented as HDF5 hard-links pointing to the data that is shared between the cells, which is stored in the root group (Figure 1 B). These include the index-associated genomic coordinates of the Hi-C contacts: `/bins/chrom`, `/bins/start`, `/bins/end`, and the general information about the stored chromosomes: `/chroms`.

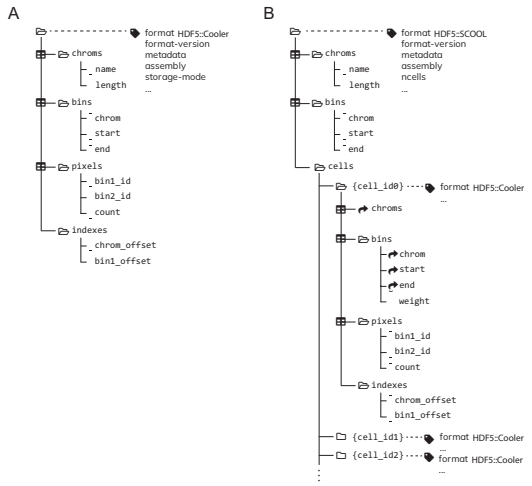

Fig. 1. (A) The structure of the cooler file format from Abdennur and Mirny (2019). (B) The structure of the single-cell cooler file format as a flavour of the cooler format. Hard linked groups and arrays are denoted with the curved arrow icon.

These shared data structures provide significant space reduction when consolidating contact maps from a multitude of cells into a single file as opposed to using a large collection of separate cooler files. As a matrix format, an scool file stores binned contact data conforming to a specific genomic segmentation. While binning naturally leads to a loss of information and comparing data sets can be difficult when bin sizes are not compatible, single-cell cooler files can be binned at any resolution and even lossless contact maps can be produced using 1-bp resolution, if desired.

2.1 Metadata

The single-cell cooler format stores specific metadata HDF5 attributes at the root level of the file: the `format` string `HDF5::SCOOL`, the `format-version`, whether the `bin-type` is fixed or variable, the `bin-size`, the genome `assembly`, the number of stored cells `ncells` and the optional field `metadata` for quality information or other user metadata.

2.2 Creation

To create a single-cell cooler file, the API can be used by calling the function *cooler.create_scool* and providing a file name, a dictionary of *bins* with the unique cell name as key (or a global common bin table, see Supplementary Material 1) and a dictionary mapping unique cell names to pixel information (Listing 1).

```
import cooler
bins_dict = {'cell1' : bins1, 'cell2' : bins2}
pixel_dict = {'cell1' : pixels1, 'cell2' : pixels2}
cooler.create_scool(cool_uri=file_name, bins=bins_dict,
      cell_name_pixels_dict=pixel_dict)
```

Listing 1. Python API example to create a scool file

2.3 Access

The interaction matrices in a single-cell cooler file can be listed with *cooler.fileops.list_coolers*. The interaction matrix of one cell can be retrieved using the resource syntax:

```
if cooler.fileops.is_scool_file(file_path):
    matrices_list = cooler.fileops.list_scool_cells(
         file_path)
    for cell in matrices_list:
        clr = cooler.Cooler(file_path + "::" + cell)
```

Listing 2. Python API example to read cells of a scool file

3 Results

The single-cell Hi-C data provided by Nagano *et al.* (2017) as raw FASTQ files has a compressed size of more than 1 TB. After demultiplexing, mapping and matrix creation several terabytes are consumed. At 10 kb resolution, 3882 individual cool files have a size of 3 GB, which is reduced to 1.9 GB using scool. At 1 MB, the cool files require 350 MB and the scool 267 MB. Gassler *et al.* (2017) provides 144 individual cool files at different resolutions. The storage reduction provided by scool are 2300 to 116 MB at 1 kb; 348 to 65 MB at 10 kb; 120 to 28 MB at 40 kb; and 63 to 26 MB at 100 kb. Compression ratios (see Supplementary Material Table 2) depend on the density and the resolution of the data. Generally, there is a greater overhead of storing a full bin table for each cell the fewer reads relative to the number of possible interactions and the higher the resolution. For example, the density for the 10 kb single-cell Hi-C data from Gassler *et al.* (2017) is up to 0.0004, while for Nagano *et al.* (2017) it is up to 0.0012. Accordingly, the scool / cool compression ratio for Gassler *et al.* (2017) (0.193), is better than that for Nagano *et al.* (2017) (0.633). See the Supplementary Material for more read coverages, densities and compression rates with respect to text and cooler files.

4 Conclusion

The single-cell cooler format makes it possible to store thousands of state-of-the-art single-cell Hi-C matrices in a single file with minimal redundancy. By storing all matrices in a space-efficient way, the reproducibility of single-cell Hi-C analyses is better achievable and the data is more accessible to a broader range of researchers. A portable container format prevents the complexity of managing thousands of files or needing to download and process large amounts of raw data from scratch. The embedding into the cooler API guarantees a fast and reliable access to the individual single-cell matrices and facilitates the use of parallel computing to improve analysis performance. The scool format is ideal for single-cell Hi-C data analysis software and is supported by scHiCExplorer (Wolff *et al.* (2020)).

Funding

We acknowledge funding from the German Science Foundation [CRC992 'Medical Epigenetics' to R.B. and B.G.]. German Federal Ministry of Education and Research [031 A538A de.NBI-RBC awarded to R.B.; 031 L0101C de.NBI-epi awarded to B.G.]. R.B. was supported by the German Research Foundation (DFG) under Germany's Excellence Strategy [CIBSS-EXC-2189-Project ID 390939984].

References

Abdennur, N. and Mirny, L. A. (2019). Cooler: scalable storage for Hi-C data and other genomically labeled arrays. *Bioinformatics*, **36**(1), 311–316.

Gassler, J. *et al.* (2017). A mechanism of cohesin-dependent loop extrusion organizes zygotic genome architecture. *The EMBO journal*, **36**(24), 3600–3618.

Koziol, Q. and Robinson, D. (2018). HDF5. [Computer Software] https://dx.doi.org/10.11578/dc.20180330.1.

Nagano, T. *et al.* (2013). Single-cell hi-c reveals cell-to-cell variability in chromosome structure. *Nature*, **502**(7469), 59.

Nagano, T. *et al.* (2017). Cell-cycle dynamics of chromosomal organization at single-cell resolution. *Nature*, **547**(7661), 61.

Ramani, V. *et al.* (2017). Massively multiplex single-cell hi-c. *Nature methods*, **14**(3), 263–266.

Stevens, T. J. *et al.* (2017). 3d structures of individual mammalian genomes studied by single-cell hi-c. *Nature*, **544**(7648), 59–64.

Wolff, J. *et al.* (2020). Galaxy HiCExplorer 3: a web server for reproducible Hi-C, capture Hi-C and single-cell Hi-C data analysis, quality control and visualization. *Nucleic Acids Research*, **48**(W1), W177–W184.