FACULTY OF MATHEMATICS AND INFORMATION SCIENCE
WARSAW UNIVERSITY OF TECHNOLOGY

"Modelling from Scratch with OpenMM"

Project report

RESEARCH WORKSHOPS -
BIOINFORMATICS WORKSHOPS

Krzysztof Tkaczyk, Filip Szlingiert, Michał Zajączkowski

13 June 2024

1 Abstract

Modern bio-informatics focused on the problem of modelling human genome. Recently discovered algorithms and developed computers give scientists chance to create more and more advanced models of chromatin that are very similar to the reality. This paper presents the result of the work on modelling chromatin using popular tools.

2 Introduction

Chromatin chain is a structure build of DNA fibre and proteins. Special types of proteins hold the very dense and complex shape of this chain. Modern researches shown that the proper organisation of chromatin plays the crucial role in some biological processes, such as replication of DNA.
The organisation of the DNA is well known. The DNA, as a whole, can be described as a fibre. It is packed into a tiny nucleus inside cell. Most mammalian cells have a single ovoid or spherically shaped nucleus with a diameter of 5 to 20 μm, making it the largest cellular organelle[1].But the DNA fibre can be approximately 2 meters long if straightened [2]. For this reason, it has to be folded in the specific way in three dimensions.
We can observe that this folding divides the chromatin into some regions. The largest are chromosomes, which are further are divided into A and B compartments. Then every compartment consists of many topologically associated domains (TADs). Finally, the TADs are the 'macro' scale result of the process of forming groups of loops. Loops are formed during loop extrusion. The most important factor in this process is the cohesin protein.
The goal was to model this structure basing on real world data. This project involved several tasks, including data analysis, model preparation, simulation execution, and result visualisation.

3 Building model

The simulation was based on the OpenMM[3] library for Python. It was created specifically for the task of modelling physical structures. Also, initial code from the EasyOpenMM[4] repository on GitHub was used.

3.1 First steps

As the familiarising oneself with software documentation and real world data is crucial in every project, it was done in the first order. The official OpenMM documentation was thoroughly read to understand the library's capabilities and functionalities, focusing on the user guide and API references. Next step was to understand the given initial code of the simulation. The study drove to the conclusion that model was to have following steps:

1. definition of the initial structure
2. definition of the forcefield and addition of proper forces
3. setting proper integrator
4. minimisation of the energy
5. visualisation and analysis of results

[1] https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4600468/
[2] https://en.wikipedia.org/wiki/DNA
[3] https://openmm.org/
[4] https://github.com/BlackPianoCat/EasyOpenMM/tree/main

3.2 Initial Structure

Definition of the initial structure is very important step of the simulation. Because imperfections of the simulation choosing improper initial structure might cause blunders in results or even errors during simulation run. For this reason, various functions generating initial structures were prepared and tested. These included: straight line, random walk (two types: one was always making unitary step and second fully random), 2d hilbert curve, 3d hilbert curve, helix, sphere, cube (two types: one was allowing only points on walls and second returning points from the whole volume). They were tested using initial simulation code. Results are shown on the Picture 1.

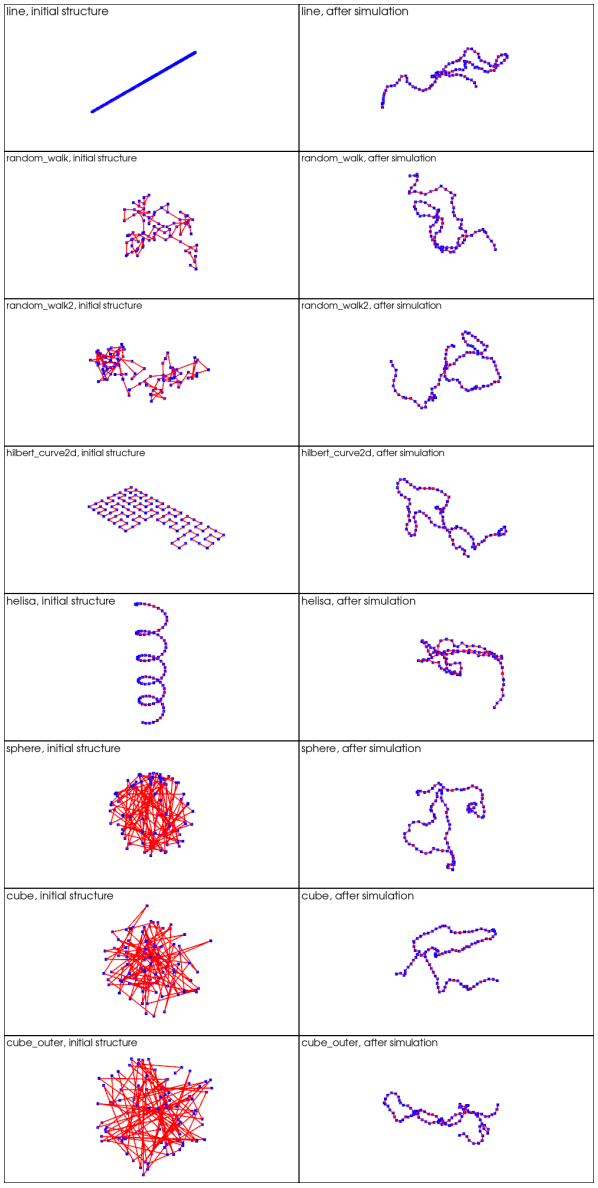

Figure 1: Generated initial structures with the results of the initial simulation presented side by side.

3.3 Forcefield

The basic information about particles (beads) where loaded from .xml file conducted with initial code. It defined mass of the one bead and some other less physical features that were mostly needed by OpenMM. In this step, also the real world data was used. The simulation was aimed on modelling the structure examined during ChIA-PET experiment for the GM12878 cell. Code was based on the information about loops in the chromatin. This information was read from .bedpe file[5] downloaded from ENCODE. There were three additional forces defined: bond force, angle force and Lennard-Jones force. Bond forces were responsible for creating the chain on beads. It also defined which pairs should be closer in result forming groups. Information about those loops was taken from the data. The role of angle forces was different. They were making the chain more rigid by connecting following three particles to make an angle. Lennard-Jones force contributed a potential energy into simulation. It caused the chain to be more self-avoiding. It was weak repulsive force.

3.4 Integrator

As an integrator we chose *LangevinIntegrator* which was also used in EasyOpenMM repository and adjusted it to fit our needs.

3.5 Minimisation of the energy

We achieved energy minimisation in similar way to how it was done in EasyOpenMM project. Steps included:

- initialising simulation
- setting up two reporters
- setting initial position of beads
- performing energy minimisation with a specific tolerance
- modifying topology file

3.6 Visualisation

We also prepared notebook with visualisation of modelling results, compared to state of initial structures using *pyvista* package. Images included in this report were generated using that file. Readers can also run simulation and generate 3D interactive output on their own simply by running *visualisation.ipynb* file from provided code.

[5] https://www.encodeproject.org/experiments/ENCSR184YZV/

4 Results

The obtained model produced results that seem rational. As we didn't have any objective metric to measure the effectiveness, we couldn't draw definitive conclusions. The fact that, regardless of the initial structure, most of the time we got a pretty similar outcome - one denser part with one looser tail - can lead to conclusions that our model works properly.

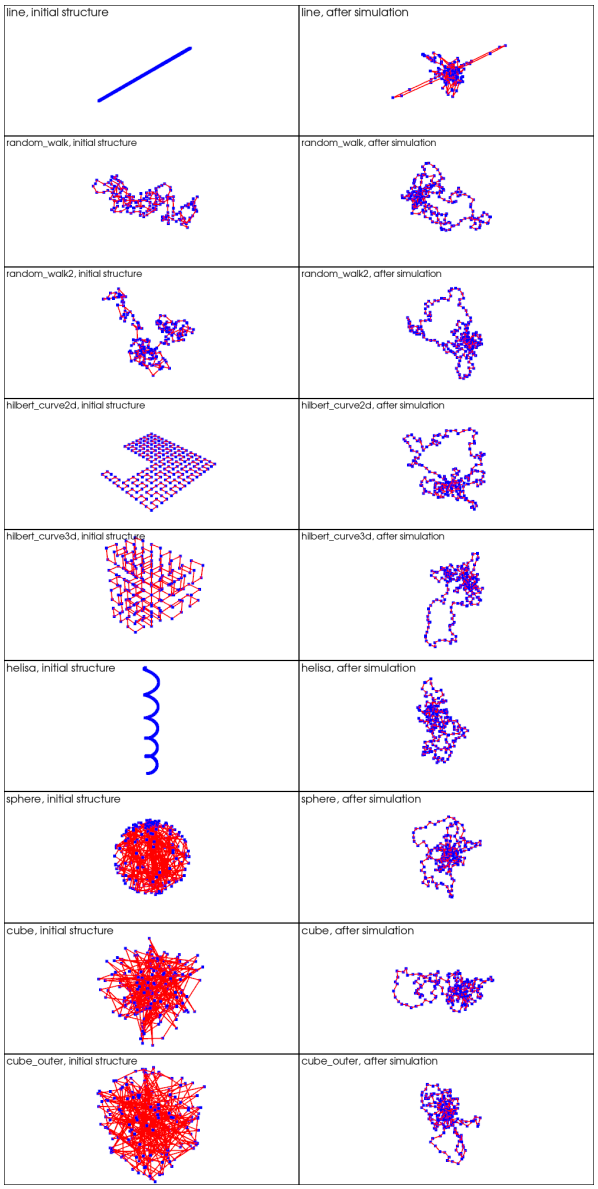

Figure 2: Generated initial structures with the results of the final simulation presented side by side.

5 Contributions

- Filip Szlingiert - adding Lennard Jonnes force, score implementation, bug fixes, code cleaning, report, presentation
- Krzysztof Tkaczyk - initial structures, visualisation, model class, notebooks, report, presentation
- Michał Zajączkowski - adding angle and bond forces, 3D hilbert curve, data prepossessing, bug fixes, report, presentation