Advantage of Using Spherical over Cartesian Coordinates in the Chromosome Territories 3D Modeling

Magdalena A. Tkacz[1] [0000−0002−3931−0474]
Kornel Chromiński[1] [0000−0001−9875−5812]

Faculty of Science and Technology,
Będzińska 39, 41-200 Sosnowiec
University of Silesia in Katowice, Poland
magdalena.tkacz@us.edu.pl
kornel.chrominski@us.edu.pl

Abstract. This paper shows results of chromosome territory modeling in two cases: when the implementation of the algorithm was based on Cartesian coordinates and when implementation was made with Spherical coordinates. In the article, the summary of measurements of computational times of simulation of chromatin decondensation process (which led to constitute the chromosome territory) was presented. Initially, when implementation was made with the use of Cartesian Coordinates, simulation takes a lot of time to create a model (mean 746.7[sec] with the median 569.1[sec]) and additionally requires restarts of the algorithm, also often exceeds acceptable (given a priori) time for the computational experiment. Because of that, authors attempted changing the coordinate system to Spherical Coordinates (in a few previous projects it leads to improving the efficiency of implementation). After changing the way that 3D point is represented in 3D space the time required to make a successful model reduced to the mean 25.3[sec] with a median 18.5[s] (alongside with lowering the number of necessary algorithm restarts) which gives a significant difference in the efficiency of model's creation. Therefore we showed, that a more efficient way for implementation was the usage of spherical coordinates.

Keywords: spherical coordinates · euclidean coordinates · 3D · chromatin decondensation · geometry · chromosome territories · modeling

1 Introduction

Computational power gives very powerful support in the life sciences today. A lot of experiments can be done – they are cheaper to conduct, their parameters can be easily modified. They are also in most cases reproducible and ethical (no wronging living creatures).

According [1] the term *modeling* is defined as *"to design or imitate forms: make a pattern"* or *"producing a representation or simulation"* and *model* is

ICCS Camera Ready Version 2020
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-50417-5_49

defined as *"a system of postulates, data, and inferences presented as a mathematical description of an entity or state of affairs"*. In fact, in a case of *in-silico* experiments more precious would be "computational model" and "computational modeling", but in this paper, it will be referred to in a shorter form. Sometimes term "modeling" is used in the context of "running computational model" – but in this paper, it will be referred to as *"simulation"*. For many disciplines creating a model is important – it allows to re-scale (extend or reduce) object, slow down or speed up modeled process, examine almost any aspect of object or process (separating parameters or taking into account a given quantity of parameters). With the use of computers, it is also possible to make visualizations and animations.

This paper describes some aspects of the modeling process that occurs in all organisms – precisely speaking occurs in almost every living cell. This process also occurs just right now – in my body while I'm writing this text, as well as in your – when you read this. This is the process of transferring genetic material, *DNA*– during cell division. This process is difficult to examine – we can only observe living cells during a relatively short time. Another difficulty here is its microscale – to observe it we have to use microscopes. And, besides of a scale – when we want to examine the interior of the cell – we have to destroy (and kill) it ... There are attempts to create an "artificial" [2] (or "synthetic" [3]) cell, but this is not an easy task. To face this up, using *"divide and conquer"* strategy there are attempts to create models of certain cell components and processes. This paper shows some new knowledge that we discover while trying to model chromosome territories (*CT*'s) being a final result of modeling and simulation chromatin decondensation (*CD*) process and documents some problems (and the way we took to solve them) to make the working model.

1.1 Motivation

Some time ago we are asked if we can help in the creation of a probabilistic model of *CT*'s (in short – CT's are the distinct 3D space occupied by each chromosome after cell division, see also Sec. 1.2). We agreed and something that we supposed to be a project for a few months of work, becomes the true mine of many different problems to be solved.

The first one we focused on, was the problem of creating appropriate model of chromatin and the model of the chromatin decondensation process (to be able to implement and simulate this process) in a phase just right after cell division.

1.2 Background

In eukaryotic cells, genetic material is not stored in a well-known form of a helix because *DNA* strand is too long (and too vulnerable to damage). It is stored as a complex of *DNA* strand and proteins – altogether called *chromatin* which is being rolled-up in a very sophisticated way [5]. This allows taking much less space and store *DNA* untangled. Probably it also helps in preventing random breaks and changes in *DNA* sequences. Researches concerning chromatin organization are important because of its influence on gene transcription [6].

There are levels of chromatin organization (depending on the level of packing) ([4], [7]). The two extreme levels of packing are *condensed* and *decondensed* ones [11]. The one – somewhere in between extreme ones, that we are interested in, is called *euchromatin*. This level of organization is often referred to as ”beads on a strand” (see Fig. 1).

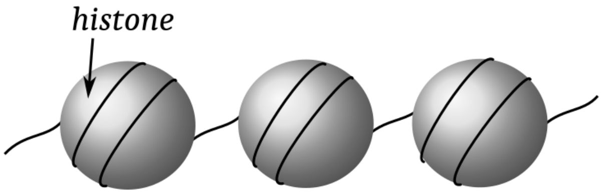

Fig. 1. Euchromatin – beads on strand

The level of chromatin condensation depends on different factors. It can be cell state (during cell cycle) but it is also known that it can be controlled by epigenetic modifications [8] or biological process [10]. The risk of *DNA* damage [9] or modification varies depending on the chromatin condensation level.

During the cell division, chromatin fibers condense into structures called chromosomes. In the period between two subsequent divisions, called interphase, chromosomes decondense and occupy 3-D areas within the nucleus. Those distinct areas – called „chromosome territories” (*CT*'s) – are regarded as a major feature of nuclear structure ([12], [22]). Chromosome territories can be visualized directly using *in-situ* hybridization with fluorescently labeled *DNA* probes that paint specifically individual chromosomes ([18], [20]). Researches concerning *CT*'s are: studying the relationship between the internal architecture of the cell nucleus and crucial intranuclear processes such as regulation of gene expression, gene transcription or *DNA* repair ([17], [19], [21]). Those studies are related to spatial arrangement, dynamics (motion tracking)[13], frequency of genomic translocations ([14]) and even global regulation of the genome [15]. Possibility of making experiments *in-silico*would speed up and make some of the experiments easier and cheaper.

2 Euchromatin Model and Chromatin Decondensation Process Modeling

The euchromatin was the starting point to model chromatin structure for us: we decided to model chromatin (and arms of chromosomes) as a sequence of

ICCS Camera Ready Version 2020
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-50417-5_49

tangent spheres (Fig. 2) – visually very similar to euchromatin (see Fig.1). Because euchromatin is observed as ”beads on a strand” and beads (sometimes also called ”domains”) are its basic structural units, we decided to make a single sphere our basic part of the chromatin chain component (and the basic structural units building up CTs). This allows also to make our model scalable – by changing the size of the sphere we can easily change the level of chromatin packing. A sphere can be also easily rendered as graphical primitive in most graphical libraries which were very important to guarantee the possibility of further CT's visualization. Our modeling process was very closely related to geometrical, visible objects, because it was very important, that the final models could be visualized – to allow visual comparison with real images from confocal microscopy.

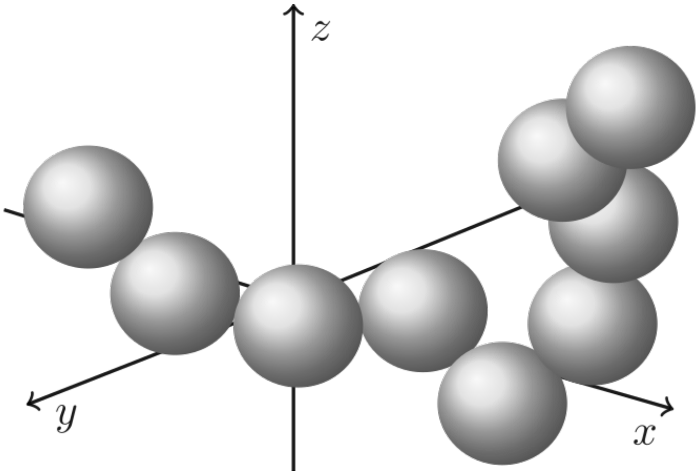

Fig. 2. Euchromatin model as a tangent spheres

We also decide to model the decondensation process by adding tangent spheres around existing ones. This effects in gradually expanding volume of the initial strand of spheres. The process continues until the stop condition was met (volume or size of decondensed chromatin).

The computational problem was as follows: starting from the initial (condensed) chromatin model (in a form ”beads-on-strand”), consisting of a sequence of mutually tangent spheres find coordinates for next N spheres (where N denotes the size –number of beads of chromatin after decondensation). Geometrically it is a problem of finding (x, y, z) being the center of a new sphere with the condition of being tangent to the previous one and not in collision in any other (previously generated).

Our first goal was to make a fully probabilistic model – that means that we do not add additional conditions like the position of centromeres, telomeres,

DOI: 10.1007/978-3-030-50417-5_49

nucleoplasm stickiness and so on (extending model and making it more "real data-driven" are in our current field of interest and research). The modeled process of decondensation can be somehow regarded as a Markov process – the subsequent state $i+1$ of decondensation strictly depends on the previous one i.

The very basic component of our model was a sphere $S((x,y,z),r)$. This notation should be read as a sphere S with a center in the point that has (x,y,z) coordinates and a radius with the length of $r,(r \geq 0)$. The ordered chain of spheres – makes our model of a chromosome, a set of indexed spheres makes a model of CT.

The very general algorithm for CT's modeling is presented in Alg.1. Line 4 and 5 reflect creating initial chromatin strand, line 6 simulation of decondensation. Altogether, they led to the generation of the model of the certain CT.

```
Algorithm 1: CT modeling algorithm (general version)
  Data: size of nucleus, size of nucleolus, the number of chromosomes, initial
        chromosome arms length, chromosome length after decondensation
  Result: model of chromosome distribution in the nucleus
1 begin
2     generate positions of nucleolus;
3     foreach chromosome do
4         generate positions of centromere;
5         generate initial arms of chromatin;
6         simulate chromatin decondensation by adding additional spheres
            "around" existing ones;
7     end
8 end
```

The last step of the algorithm (line 6) proved to be the most demanding and challenging, which is described in the next section.

3 Experiments and Results

In the following section, we document the way we take to successfully made the probabilistic model of CT's.

3.1 Modeling Chromatin Decondensation with *CC*

At first, we used the *Cartesian coordinates* (CC). First, the algorithm generates coordinates for the sphere that are denoted as the *centromere*, and next add to it the next ones until it reaches the (given a-priori) length of arms for certain chromosome. Having model of the entire chromosome algorithm draw a *id* of one of the present spheres $S_i((x_i,y_i,z_i),r)$ (from those composed the chromosome) and then draws "candidate coordinates": x_{i+1}, y_{i+1} and z_{i+1} for the center for the next sphere. The new coordinates are to be from limited range – not too far current sphere's (as they should be tangent).

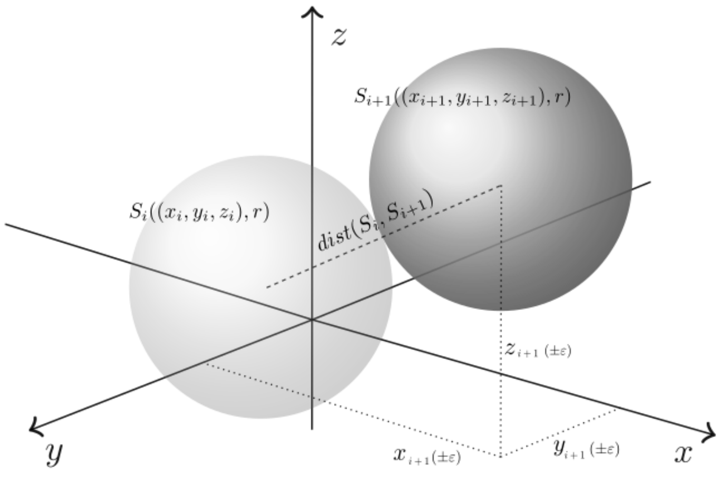
z
$S_{i+1}((x_{i+1}, y_{i+1}, z_{i+1}), r)$
$S_i((x_i, y_i, z_i), r)$
$dist(S_i, S_{i+1})$
$z_{i+1}\,(\pm\varepsilon)$
y
$x_{i+1}(\pm\varepsilon)$
$y_{i+1}\,(\pm\varepsilon)$
x

Fig. 3. Way of determining the location of S_{i+1} sphere using *CC*

To allow small flexibility, the ε value to the drawn coordinates was introduced. When we had coordinates drawn, the distance $dist(S_i, S_{i+1})$ was calculated to check whether a new sphere can be added. The distance was computed by calculating ordinary Euclidean distance.

$$(S_i, S_{i+1}) = \sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2 + (z_{i+1} - z_i)^2} \tag{1}$$

If $dist(S_i, S_{i+1})$ was appropriate, the conditions to not collide with existing elements were checked. If all conditions are met – new sphere were added (for details see [26]).

There were no problems with the generation of the initial chromatin strand as a sequence of spheres (chromosome). The problem emerges when we tried to simulate the decondensation of chromatin: generation of a model takes a lot of time, and we noticed that sometimes simulation was unsuccessful. We discovered (after log analysis) that the algorithm got stuck trying to find coordinates for S_{i+1}. So, we added additional function that triggers restart of algorithm after 500 unsuccessful attempts for placing S_{i+1} sphere (see Alg. 2 lines 12-14). If S_{i+1} cannot be placed – algorithm starts over and searches possibility to add S_{i+1}, but for another sphere forming chromosome.

The pseudocode for this version of the algorithm is shown in Alg.2. In the first step it generates the "candidate coordinates" for S_{i+1} center (Alg. 2 lines

3-6). Thanks to ε a possibility that new sphere could be a little too far, or too close the previous S_i. The fine-tuning is made by an additional function that checks the distance from the previous sphere (Alg. 2 lines 7-8). Additional code for stuck detection that triggers restarting computations are in (Alg. 2 lines 12-14).

Algorithm 2: Simulating chromatin decondensation with *CC*

```
   Data: sphere rad ($r$), coordinates of previous sphere, $\varepsilon = 0.001$
   Result: new sphere
 1 begin
 2     try=0;
 3     do
 4         generate co-ordinates as
 5         $new\_sphere\_x = previous\_sphere\_x \pm random(0, 2 \cdot r + 2 \cdot \varepsilon)$
 6         $new\_sphere\_y = previous\_sphere_y \pm random(0, 2 \cdot r + 2 \cdot \varepsilon)$
 7         $new\_sphere\_z = previous\_sphere_z \pm random(0, 2 \cdot r + 2 \cdot \varepsilon)$;
 8         check distance from previous sphere
 9         $sqrt((new\_sphere\_x - previous\_sphere\_x)^2 + (new\_sphere\_y -$
             $previous\_sphere\_y)^2 + (new\_sphere\_z - previous\_sphere_z)^2) <$
             $(2 \cdot r_d \pm \varepsilon)$;
10         is inside nucleus;
11         is outside nucleolus;
12         if $try == limit\_of\_try$ then
13             reset model generation
14         end
15         foreach $existing\ sphere$ do
16             new sphere have no collision;
17         end
18     while $sphere\ generate\ correct$;
19 end
```

This makes the simulation of *CD* process long and inefficient, and the result was disappointed: the algorithm got stuck relatively often. The measured number of necessary restarts to complete model creation is shown in Tab.1.

Table 1. The number of restarts during simulating *CD* using *CC*

modeling run no.	***number of restarts***	***ineffective searches***
1	23	11500
2	16	8000
3	20	10000
4	22	11000
5	11	5500

In one model creation, about 650 spheres should be placed as the tangent ones, so it was easy to asses the number of inefficient searches – they are presented in the last column of Tab.1.

Tab.2 showed the time needed to generate one *CT* model. Time was measured in seconds, basic statistics were also given, measurements were made on 40 generated models.

Table 2. Time of modeling *CT*'s with *CC* used in simulation of *CD* [in seconds, measurements from 40 models creation]

Time of simulating chromatin decondensation using CC	
mean (time)[s]	746,747
standard deviation [s]	438,362
median	569,198
min value [s]	303,53
max value [s]	1794,582
Q1	416,536
Q3	828,267

That was not a satisfactory result. We had to rethink the way we implement the decondensation of chromatin. We decided to try to add – at first sight – additional computations: shifting (change location) of the center of coordinate systems. Then we were able to use the notion of the neighborhood with a fixed radius (inspired by a topology) and use spherical coordinates (see Fig.4).

We were aware of the fact that shifting the coordinate system takes additional time – but the solution with *CC* works so bad, that we hope that this approach will work a little better. The result of this change beats our expectations – which is described and documented in the next sections.

3.2 Modeling Chromatin Decondensation Process Using *SC*

We decided to try *Spherical Coordinates* (*SC*) [27] instead of *CC* (for those, who are not familiar with different coordinate systems we recommend to take a look at [28], [29]). When we wanted to add sphere S_{i+1} to the certain one S_i, we first made a shift of the center of the coordinate system in such a way, that the center of coordinate system was situated in the middle of S_i sphere (see Fig.4).

This let us search for the S_{i+1} by drawing two angles and using just one parameter: $2r$.

After switching to the *SC*, we got rid of the problem of looping the simulation during attempts of finding the location for the S_{i+1}. Therefore, the function that restarts *CT* model creation could be removed.

We made measurements – time necessary to generate *CT* models (equivalent to the time of *CD* simulation) with shifting coordinate system and using spherical coordinates is presented in Tab.3.

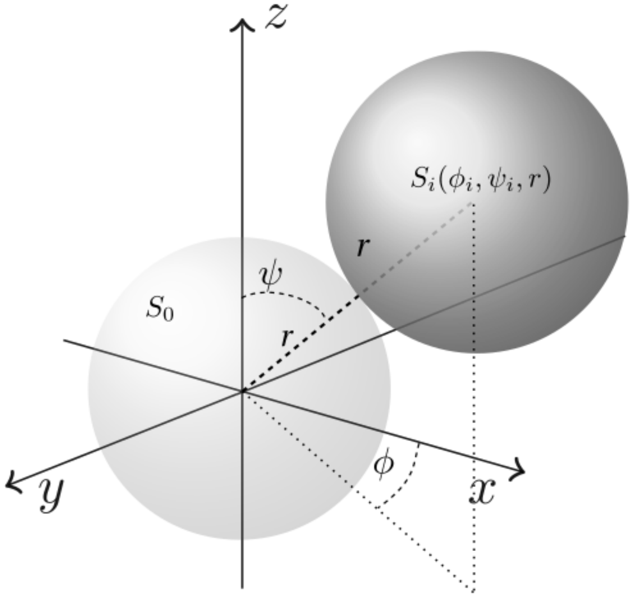

Fig. 4. Way of determining the location of S_{i+1} sphere using *SC*

Algorithm 3: Simulating chromatin decondensation process with *SC*

```
Data: sphere rad ($r$), coordinates of previous sphere
Result: new sphere
begin
    do
        generate coordinates as
        $\psi = random(0, \pi)$
        $\phi = random(0, 2 \cdot \pi)$
        $new\_sphere\_x = previous\_sphere\_x + 2 \cdot r \cdot cos(\psi) \cdot sin(\phi)$
          $new\_sphere\_y = previous\_sphere\_y + 2 \cdot r \cdot sin(\psi) \cdot sin(\phi)$
          $new\_sphere\_z = previous\_sphere\_z + 2 \cdot r \cdot cos(\phi)$;
        is inside nucleus;
        is outside nucleolus;
        foreach existing sphere do
            new sphere have no collision;
        end
    while sphere generate correct;
end
```

Time of creating *CT* models decreases significantly in comparison to the use of *CC*. This had a direct and significant impact on the time of the model creation.

Table 3. Time of simulating *CD* process with the use of *SC*[in seconds, measured on 40 models]

Time of CT model creation with SC	
mean (time)[s]	25,381
standard deviation [s]	19,758
median	18,467
min value [s]	6,380
max value [s]	91,275
Q1	11,107
Q3	30,701

3.3 Comparison of Computational Time of *CT* modeling with *CC* and *SC*

To follow the rigor for scientific publications (despite very clear difference between times showed in Tab.2 and Tab.3) we made an analysis, presented in this section. For the purpose of visual comparison of the times of *CT* model creation we prepared a boxplot (see Fig. 5) for general view.

In Fig. 5 the difference, in general, is easy to notice. There is even no single element of the chart (neither whiskers nor dots (outliers)) that overlaps each other.

It is easy to notice a huge difference between computing time (and its stability) in both cases.

For the record we made statistical test – the result is presented in Tab.4. We calculated the value of the t-test, to confirm that the difference in creation times of model (*CC* and *SC*) is statistically significant (p-value below 0.05 means that the difference is statistically significant).

Table 4. Statistics for two sample t-test (modeling time with *CC* and with *SC*)

Statistics for two sample t test – modeling with CC and with SC	
t	-10.393
df	40.201
p-value	2.95E-13

This proves the statistical significance between modeling time in described two methods.

4 Conclusions and Future Works

Based on presented in this paper results we can conclude that when you model in 3D space, using Spherical coordinates may lead to a more efficient implementation of the algorithm, even when you have to shift the center of the

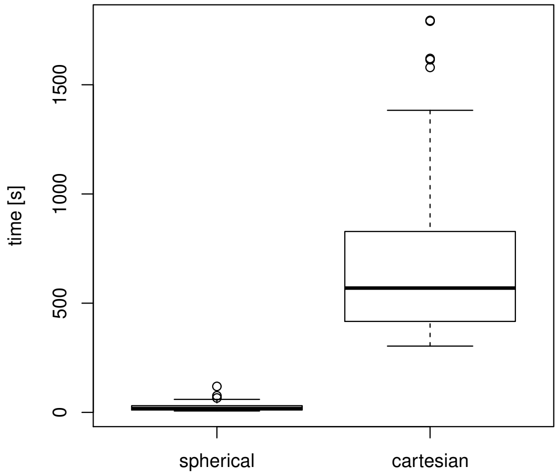

Fig. 5. Time [seconds] of simulation chromatin decondensation – consolidated comparison of *CC* and *SC* coordinates used [from generation of 40 models]

coordinate systems. The solution when using Euclidean distance in the Cartesian coordinate system in implementation was much more time-consuming. What is more important – it often does not finish modeling process in an acceptable time (sometimes we have to break simulation after 3 weeks of computing on a computer with 16Gb RAM and i5 processor), if it finishes at all (do not got stuck).

As future work, knowing that using a spherical coordinate system is helpful we want to examine the effectiveness of quaternion-based implementation as a way to represent coordinates in 3D space. We also want to check in a more detailed way, what has an impact: only changing the center of the coordinate system, only changing the way of point representation – or both.

Because it is not the first time when we noticed significant change (in plus) after using Spherical (or hyperspherical – in more dimensions) Coordinates instead of the Cartesian ones, we plan (after finishing actual projects with deadlines) design and conduct a separate experiment. We want to investigate in a more methodological and ordered way to answer the question: why Spherical coordinates give better results in computational implementations?

Our case study also shows that it is possible that geometrical and visual thinking while modeling in 3D space can be helpful. With the "pure algebraic" thinking (based on the calculation on coordinates) finding the idea – to search

DOI: 10.1007/978-3-030-50417-5_49

in the neighborhood, shifting the center of the coordinate system and next using direction (angles) and fixed distance – would be more difficult (if even possible).

References

1. https://www.merriam-webster.com/dictionary/model(Last access: 13.04.2020)
2. E-cell Project webpage https://www.e-cell.org/ (Last access 06.02.2020)
3. Synthetic Cell Project, http://www.syntheticcell.eu/ (Last access 6.02.2020)
4. Daniel R. Larson, Tom Misteli: The genome—seeing it clearly now. https://science.sciencemag.org/content/357/6349/354.full (Last access 13.04.2020)
5. Kieran J. O'Donnell1, Vivette Glover: Maternal Prenatal Stress and the Developmental Origins of Mental Health. The Epigenome and Developmental Origins of Health and Disease, Page: 103-126 (2016), doi: 10.1016/b978-0-12-801383-0.00007-4
6. Moumné L., Betuing S., Caboche J: Multiple Aspects of Gene Dysregulation in Huntington's Disease. Frontiers in neurology. 4. 127. 10.3389/fneur.2013.00127, (2013).
7. An Jansen, Kevin J. Verstrepen: Nucleosome Positioning in Saccharomyces cerevisiae. Microbiology and Molecular Biology Reviews Jun 2011, 75 (2) 301-320; DOI: 10.1128/MMBR.00046-10
8. Martin, R. M., Cardoso, M. C. (2010). Chromatin condensation modulates access and binding of nuclear proteins. FASEB journal : official publication of the Federation of American Societies for Experimental Biology, 24(4), 1066–1072. doi:10.1096/fj.08-128959
9. Burgess, Rebecca C; Burman, Bharat; Kruhlak, Michael J; Misteli, Tom: Activation of DNA damage response signaling by condensed chromatin. Cell reports, ISSN: 2211-1247, Vol: 9, Issue: 5, Page: 1703-1717. (2014)
10. Zhigang Lu, Chuanmao Zhang, Zhonghe Zhai: Nucleoplasmin regulates chromatin condensation during apoptosis. Proceedings of the National Academy of Sciences Feb 2005, 102 (8) 2778-2783; DOI: 10.1073/pnas.0405374102
11. Wolfram Antonin, Heinz Neumann: Chromosome condensation and decondensation during mitosis. Current Opinion in Cell Biology, Volume 40 (2016), Pages 15-22, ISSN 0955-0674, https://doi.org/10.1016/j.ceb.2016.01.013. http://www.sciencedirect.com/science/article/pii/S0955067416300059(Last access 13.04.2020)
12. Cremer, Thomas, and Marion Cremer. "Chromosome territories." Cold Spring Harbor perspectives in biology vol. 2,3 (2010): a003889. doi:10.1101/cshperspect.a003889
13. How do chromosome territory dynamics affect gene redistribution? https://www.mechanobio.info/genome-regulation/how-do-chromosome-territory-dynamics-affect-gene-redistribution/(Last access 13.04.2020)
14. Leah F Rosin, Olivia Crocker, Randi L Isenhart, Son C Nguyen, Zhuxuan Xu, Eric F Joyce: Chromosome territory formation attenuates the translocation potential of cells. eLife 2019;8:e49553 doi: 10.7554/eLife.49553
15. Fritz, AJ, Sehgal, N, Pliss, A, Xu, J, Berezney, R. Chromosome territories and the global regulation of the genome. Genes Chromosomes Cancer. 2019; 58: 407– 426. https://doi.org/10.1002/gcc.22732

16. Albiez H, Cremer M, Tiberi C, et al. 2006. Chromatin spheres and the interchromatin compartment form structurally defined and functionally interacting nuclear networks. Chromosome Res 14, 707-733.
17. Cremer M, Kupper K, Wagler B, et al. 2003. Inheritance of gene density-related higher order chromatin arrangements in normal and tumor cell nuclei. J Cell Biol 162, 809-820.
18. Cremer T, Cremer C. 2001. Chromosome territories, nuclear architecture and gene regulation in mammalian cells. Nat Rev Genet 2, 292-301.
19. Fraser P, Bickmore W. 2007. Nuclear organization of the genome and the potential for gene regulation. Nature 447, 413-417.
20. Pinkel D, Landegent J, Collins C, et al. 1988. Fluorescence in situ hybridization with human chromosome-specific libraries: detection of trisomy 21 and translocations of chromosome 4. Proc Natl Acad Sci U S A 85, 9138-9142.
21. Zhang Y, McCord RP, Ho YJ, Lajoie BR, Hildebrand DG, Simon AC, Becker MS, Alt FW, Dekker J. 2012. Spatial organization of the mouse genome and its role in recurrent chromosomal translocations. Cell 148, 908-921.
22. Meaburn K., Misteli T.:Cell biology - Chromosome territories. Nature. 445. 379-781. 10.1038/445379a. (2017)
23. Cremer T., Cremer M., Dietzel S., Müller S., Solovei I., Fakan S: Chromosome territories – a functional nuclear landscape. Current opinion in cell biology. 18. 307-16. (2006). 10.1016/j.ceb.2006.04.007.
24. Misteli, T. Chromosome territories: The arrangement of chromosomes in the nucleus. Nature Education 1(1):167, (2008) https://www.nature.com/scitable/topicpage/chromosome-territories-the-arrangement-of-chromosomes-in-3025/(Last access 13.04.2020)
25. What are chromosomes and chromosome territories? https://www.mechanobio.info/genome-regulation/what-are-chromosomes-and-chromosome-territories/(Last access 13.04.2020)
26. Tkacz, M. A., Chromiński K., Idziak-Helmcke D., Robaszkiewicz E., Hasterok R.: Chromosome Territory Modeler and Viewer **2**(5), https://doi.org/10.1371/journal.pone.0160303 (2016)(Last access 13.04.2020)
27. Wolfram Alpha http://mathworld.wolfram.com/SphericalCoordinates.html. (Last access 4.02.2020)
28. Nykamp DQ: Spherical coordinates. From Math Insight. http://mathinsight.org/spherical_coordinates (Last access 4.02.2020)
29. Lum C.: Cartesian, Polar, Cylindrical, and Spherical Coordinates https://youtu.be/FLQXW6G9P8I (Last access 4.02.2020)

ICCS Camera Ready Version 2020
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-50417-5_49