Technická univerzita v Košiciach
Fakulta elektrotechniky a informatiky

Genome structure visualization

Bakalárska práca

2021 **Bc. Oleksandr Korotetskyi**

Technická univerzita v Košiciach
Fakulta elektrotechniky a informatiky

Genome structure visualization

Bakalárska práca

Študijný program:	Informatika
Študijný odbor:	9.2.1. Informatika
Školiace pracovisko:	Katedra počítačov a informatiky (KPI)
Školiteľ:	Jan Genci
Konzultant:	

Košice 2021 **Bc. Oleksandr Korotetskyi**

Abstrakt v SJ

V tejto bakalárskej práci analyzujem všeobecnú štruktúru genómu rôznych organizmov (eukaryoty, prokaryoty a vírusy) s cieľom porozumieť rozdielom a navrhnúť možné riešenia ich vizualizácie. Aby som pochopil súčasný stav problému, popisujem a porovnávam niektoré populárne existujúce programy, ktoré sú schopné vizualizovať vlastnosti genómu.V ďalšom kroku vykonávam 2D vizualizáciu a analýzu genómu SARS-CoV-2 pomocou niektorých existujúcich a mnou vyvinutých techník. Potom sa výsledky dosiahnuté počas vizualizácie overia a skripty, ktoré som na ich získanie použil, sa skomponujú do samostatnej aplikácie.

Kľúčové slová v SJ

Programovanie, bioinformatika, vizualizácia údajov, genóm, covid-19

Abstrakt v AJ

In this bachelor thesis I analyze the general genome structure of different organisms (eukaryotes, prokaryotes and viruses) in order to understand the differences and come up with possible solutions for their visualization. Also, in order to understand the current state of a problematic, I describe and compare some of popular existing programs that are capable of visualizing genome properties.As the next step, I perform 2D visualization and analysis of SARS-CoV-2 genome using some existing techniques and one developed by me. Later, the results achieved during the visualization are verified and the scripts that were used to obtain them being composed into a stand-alone application.

Kľúčové slová v AJ

Programming, bioinformatics, data visualization, genome, covid-19

Bibliografická citácia

KOROTETSKYI, Oleksandr. *Genome structure visualization* . Košice: Technická univerzita v Košiciach, Fakulta elektrotechniky a informatiky, 2021. **??**s. Vedúci práce: Jan Genci

Tu vložte zadávací list pomocou príkazu
`\thesisspec{cesta/k/suboru/so/zadavacim.listom}`
v preambule dokumentu.
Kópiu zadávacieho listu skenujte čiernobielo (v odtieňoch sivej) na 200 až 300 DPI! Nezabudnite do jednej práce vložiť originál zadávacieho listu!

Čestné vyhlásenie

Vyhlasujem, že som záverečnú prácu vypracoval(a) samostatne s použitím uvedenej odbornej literatúry.

Košice, 13.5.2021 .....

Vlastnoručný podpis

Poďakovanie

Na tomto mieste by som rád poďakoval svojmu vedúcemu práce za jeho čas a odborné vedenie počas riešenia mojej záverečnej práce.

Rovnako by som sa rád poďakoval svojim rodičom a priateľom, najmä *Adamovi Galuškovi*, *Nikite Solomatinovi* a *Sultanu Shaimardanovi* za ich podporu a povzbudzovanie počas celého môjho štúdia.

V neposlednom rade by som sa rád poďakoval spoločnosti *RedBull* a *Ozzy Osbornovi* za za energiu pri napísaní tejto práce.

Obsah

Zoznam obrázkov

Zoznam tabuliek

Úvod

The order of DNA sequence and its variations are the very aspect which dictates the developmental processes of an organism, determines susceptibility to various diseases and uniquely identifies each creature. This area has always been on the periphery of the interests of scientific society, since the discovery in 1869 by Swiss-born biochemist Fredrich Miescher. For instance, The Human Genome Project (HGP) which started on October 1, 1990 and completed in April 2003 was one of the greatest feats of exploration in history of science. It was aimed at reading all the DNA sequences of our species, Homo sapiens. All in all, HGP introduced us the ability to read nature's complete genetic blueprint for building a human being. However, despite the successful completion of the project, a number of unknown DNA properties is still exists and demands the profound studying.

The most of the people do not know what bioinformatics is and this thesis is an attempt of diving in it.

The COVID-19 pandemic intoduced new challenges for the humanity, and bioinformatics particularly: after the sequencing every genome should be analyzed properly in order to obtain better understanding of its features. Often, during the genome analysis different visualization techniques are used in order to portray data in an easy-to-understand manner. Therefore, the visualization of SARS-CoV-2 genome structure and properties deserves special attention.

The first chapter of this thesis is aimed at analyzing the general genome structure of different organisms (eukaryotes, prokaryotes and viruses) in order to understand how to visualize it. Also, this chapter contains the overview of existing solutions for genome data representing.

The second chapter of this thesis contains the analysis of files that are used to store SARS-CoV-2 data, analysis and implementaton of different 2D visualiaton techniques to the coronavirus genome. Moreover, this thesis suggests a new approach of how genome data can be visualized in order to see the smallest differences in genomes without profound investigation. In addition, the second chapter also describes the composition of the software that is made of the scripts that

were used to visualize SARS-CoV-2 genome.

Although several DNA processing tools exist, the problem of representing different genome properties which might vary at various species, concerning either the number of particular genes or complete chromosomes (if they are present), remains still actual. Moreover, the processing of the genome and its visualization demand an efficient approach, concerning the size of data and computational capabilities of an average computer. This work aims at representing some key genome properties in such a way.

Formulacia ulohy

In this bachelor thesis I would like to analyze the genome structure of organisms, estimate the existing solutions to the genome data representing. The very aim is to perform SARS-CoV-2 genome analysis, visualization and comprasion using different techniques and to compare the results with the existing solutions. Another goal is to compose all the scripts that were used for the analysis and visualization into a stand-alone aplication.

1 Analytická časť

This chapter covers analysis of genome structure of different organisms, or, in other words, biological background which is a necessity for creating any bioinformatics software. Moreover, it particularly describes existing solutions for genome data representing. Each of the mentioned topics is described in the subsection corresponding to it.

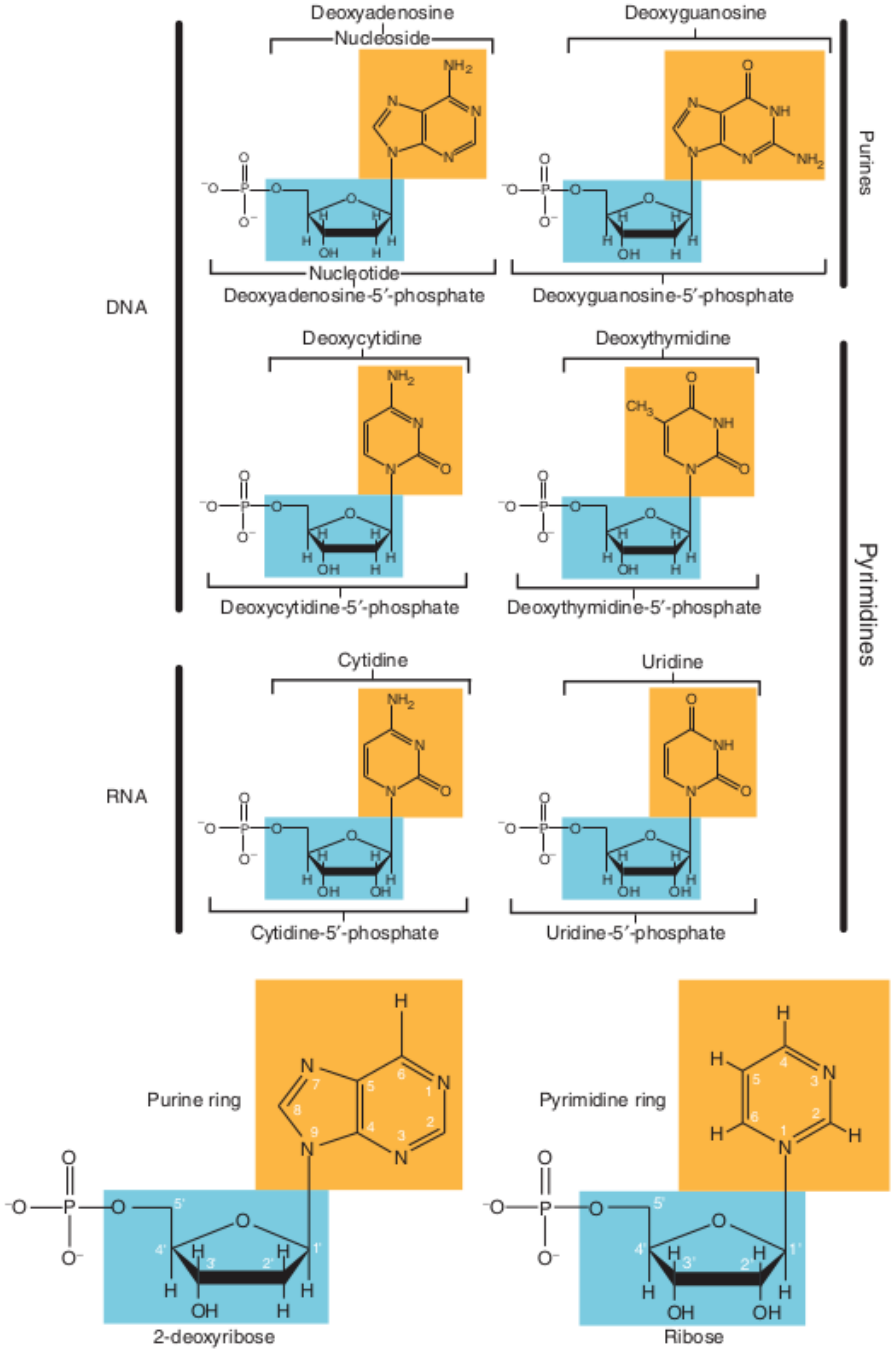
1.1 General Genome Structure

In most eukaryotic and prokaryotic organisms the hereditary material is either linear double-stranded DNA (deoxyribonucleic acid) molecules or a circular double-stranded DNA molecule. However, some extracellular life forms, might use RNA (ribonucleic acid) as the building block for their genome. For instance, viruses have a genome composed of either single-stranded DNA, double-stranded DNA or RNA, depending on the type of a virus. Therefore, a genome itself, is the complete content of genetic information in an organism, or in other words, all the unique DNA (RNA) sequences the organism possesses.

1.1.1 Nucleotides: the basic subunit of genome

Both of DNA and RNA are polymeric molecules, that are composed of linear chains of various combinations of four different subunits, called nucleotides. The nucleotide itself is the basic unit of the DNA and RNA molecules, the monomer, which, however, could be found in the cell not only as the bearer of the genetic information, but also as a carrier of energy used to power enzymatic reactions [1]. A five-carbon-atom sugar, a phosphate group and a nitrogenous base are three distinct components which, combined together, make up the quite complex nucleotide molecule. The combination of sugar and base is called a nucleoside, while the phosphate-sugar-base is termed a nucleotide.

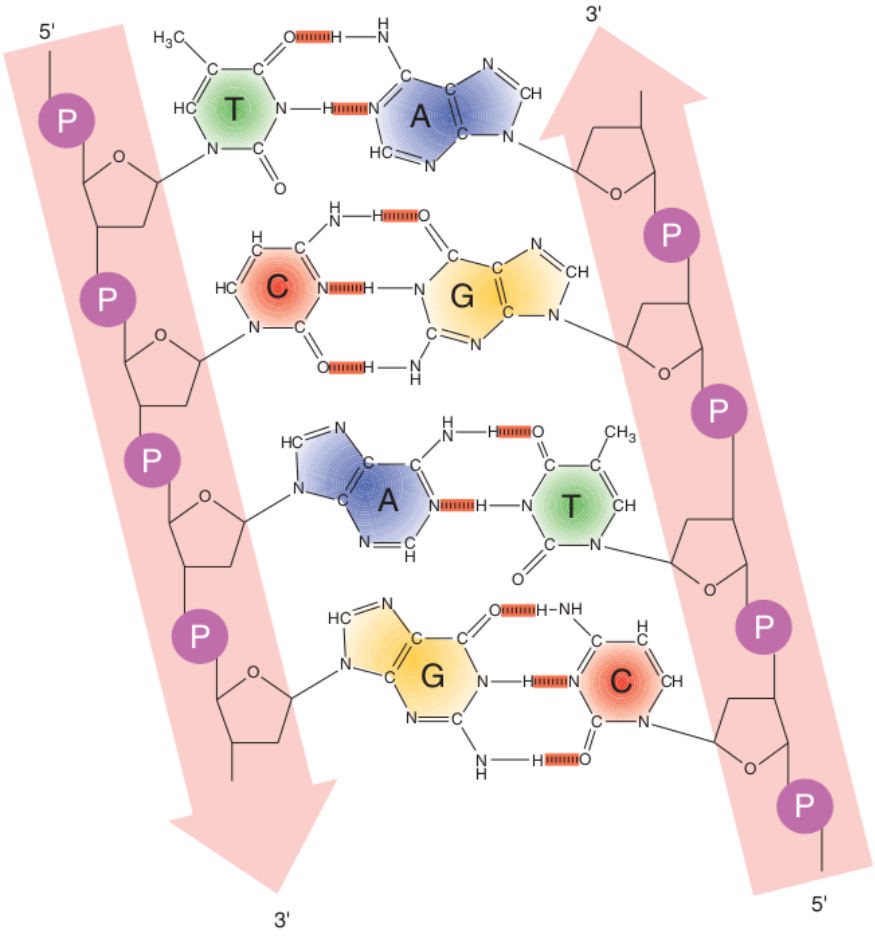
Dinucleotide, trinucleotide and polynucleotide are the terms corresponding to two, three or many nucleotides connected with each other respectively.



Obr. 1.1: The structures of the pyrimidines and purines found in DNA and RNA. The sugar groups are highlighted in blue and the nitrogenous bases are highlighted in orange. The atoms of the sugar are numbered from 1 to 5. The atoms of the purine ring are numbered from 1 to 9, while those of the pyrimidine ring are numbered from 1 to 6.

A nucleotide can be either a double-ringed purine or a single-ringed pyrimidine. Guanine (G) and adenine (A) are the common purines for both of DNA and RNA; the pyrimidine called cytosine (C) is also present in both nucleic acids.

However, the pyrimidine uracil (U) is limited only to RNA, being replaced with thymine (T) in DNA. There are merely two base-pair combinations that are permissible – A base-paired with T (U) and C base-paired with G. It happens due to the geometries of the nucleotide bases and relative positions of atoms which participate in the connection [2]. This property makes two sequences of polynucleotides in helix complement.



Obr. 1.2: DNA base pairing and complementation. The two chains of the helix, arrowed in the 5′ to 3′ direction, are antiparallel. The bases on one strand of the helix are complementary to those on the opposite strand, A always base pairs with T and G always base pairs with C

Discrete nucleotides are attached to each other through sugar–phosphate bonds that connect the phosphate group on the 5′ carbon of one nucleotide with the hydroxyl group on the 3′ carbon of another nucleotide. The base pairing between adenine and thymine (uracil) involves two hydrogen bonds, while between cytosine and guanine involves three hydrogen bonds.

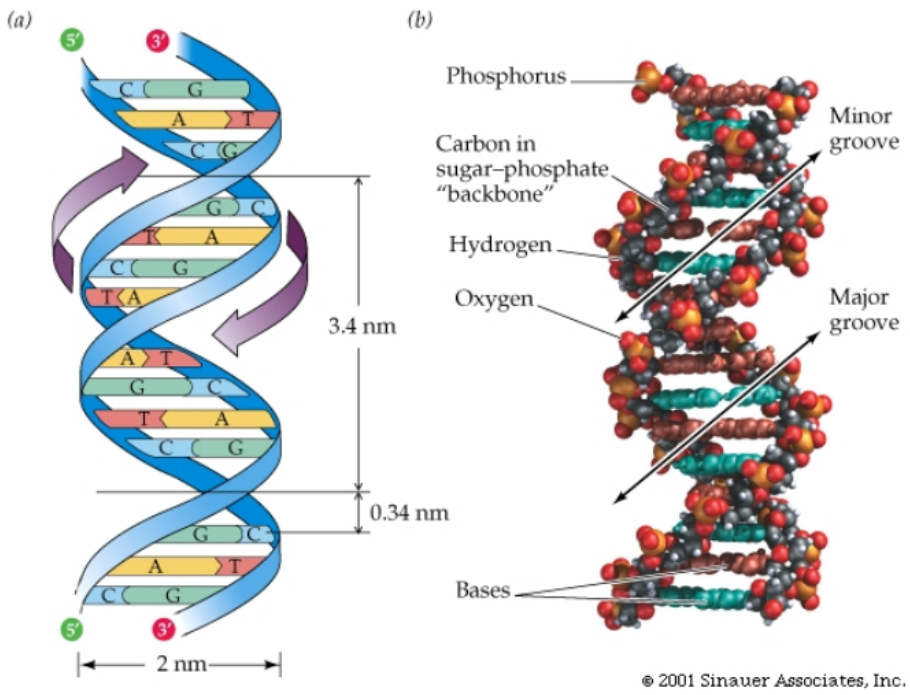
1.1.2 Nucleodic acid spatial stucture

As the three-dimensional structure of a nucleotide is not completely rigid, it is possible for DNA to have various spatial architectures: A-form, B-form, Z-form and the circular one. The position of the base relatively to the five-carbon-atom sugar can be changed by a rotation around the N-glycosidic bond and, in this way, significantly affect the three dimensional configuration of the molecule and helix consequently.

Tabuľka 1.1: DNA double helix

Features of the different conformations of the DNA double helix			
Feature	B-DNA	A-DNA	Z-DNA
Type of helix	Right-handed	Right-handed	Left-handed
Number of base pairs per turn	10	11	12
Distance between base pairs (nm)	0.34	0.29	0.37
Distance per complete turn (nm)	3.4	3.2	4.5
Diameter (nm)	2.37	2.55	1.84
Major groove	Wide, deep	Narrow, deep	Flat
Minor groove	Narrow, shallow	Wide shallow	Narrow, deep

Moreover, although usually single-stranded, some RNA sequences have the ability to form a double helix. However, double helix RNAs are rare and do not appear to participate in some genome related processes in the eukaryotic and prokaryotic organisms. Since circular DNA may exist in several forms including single-stranded c-DNA, intact double-stranded c-DNA (closed circles with both strands covalently linked), nicked ds-c-DNA (only one strand covalently linked) and in the form of "concatenated circles" their properties are not described in the attached table.



Obr. 1.3: Double stranded DNA helix.

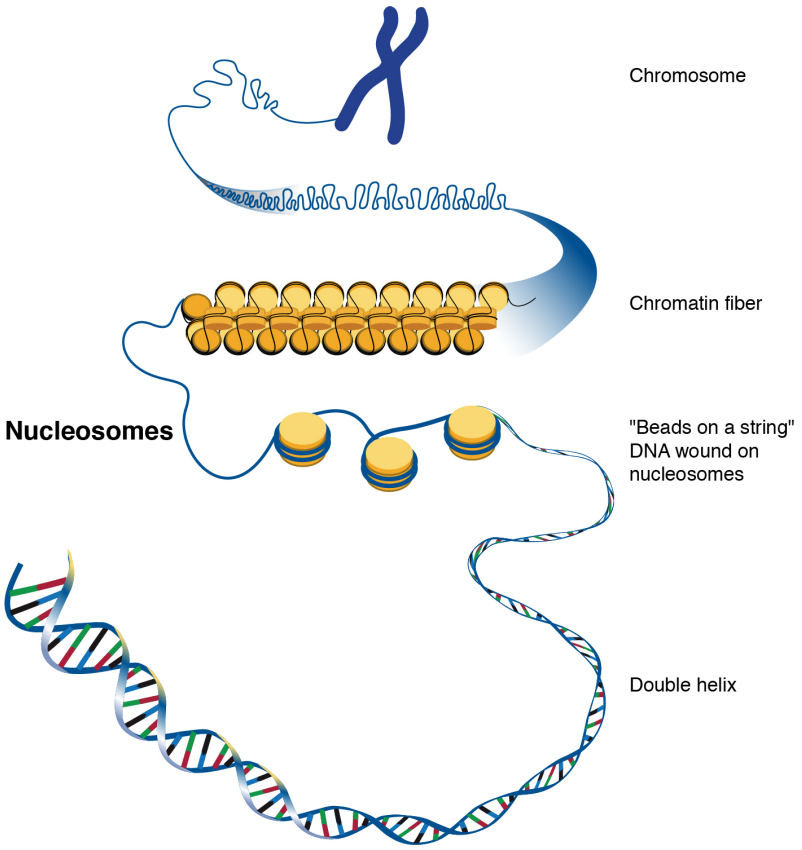
1.1.3 Eukaryotic genome organization

Eukaryotes are organisms whose cells have a nucleus enclosed within a nuclear envelope [3]. In eukaryotic cells nucleic acid is situated in a membrane-bound organelle called the nucleus. The nuclear genome is split into a set of linear double-helix DNA molecules, each contained in a chromosome. A chromosome itself is a linear string of DNA wrapped around associated proteins that give the connected nucleic acid bases a structure.

No exceptions to this pattern are known: eukaryotes that have been studied posess DNA molecules that are always linear and have at least two chromosomes. [4] The only variability at this level of organization of eukaryotic genome is coherent with the number of chromosomes. Moreover, it appears, that biological features of an organism have no dependence on the number of chromosomes.

The ends of eukaryotic chromosomes are also the ends of linear duplex DNA and as such require a special structure to ensure that they are maintained. The reason for this is connected with the way in which double-stranded DNA is replicated [5]. In the absence of a method for completing the ends, chromosomes would become shorter after each cell division.

Telomeres are specialized nucleic acid sequences whose role is to protect the ends of chromosomes. In most eukaryotes the telomere consists of a short repeat of TTAGGG many hundreds of units long, but the repeated segment might differ



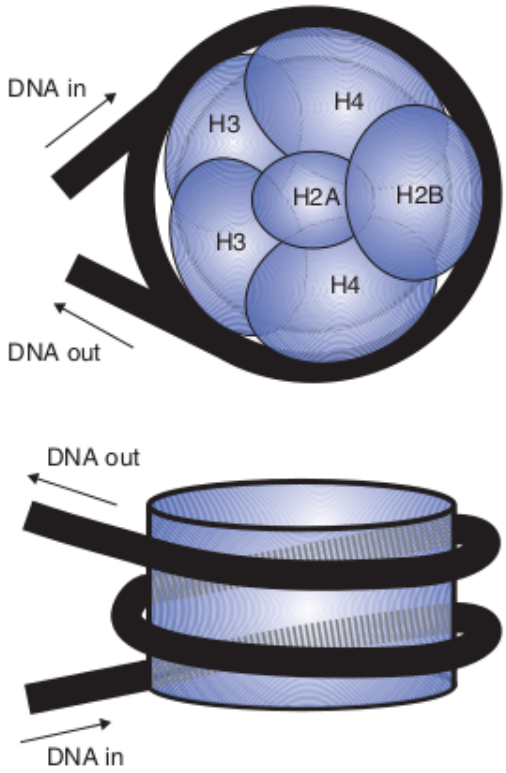
Obr. 1.4: Eukatyotic genome organization.

among species. Also, these repeats vary considerably in length between species, however each species maintains a fixed average telomere length in its germline.

Despite the size of a nucleus (5-10 um), an overall length of DNA in the human cell is approximately 2.1m and can be packed inside the cell because of the method the nucleic acid is stored.

The genetic material in viruses and bacteria consists of strings of DNA or RNA almost devoid of proteins. However, in eukaryotes, a substantial quantity of protein is associated with the DNA to form chromatin. At the lowest level, the DNA is organized by wrapping DNA strands around he proteins called histones, that contain a large amount of positively charged amino acids arginine and lysine. Those amino acids, and histones in general, play the crucial structural role, making it possible to bind the negative charged phosphate groups of the DNA nucleotides.

Averagely, the DNA rolled around the histones consists of 140-150 base pair, dependently on the species. Such a complex of DNA and histones is termed a nucleosome. These nucleosomes can be further coiled into increasingly larger coils



Obr. 1.5: The nucleosome structure. H2A, H2B, H3 and H4 represent different types of histones.

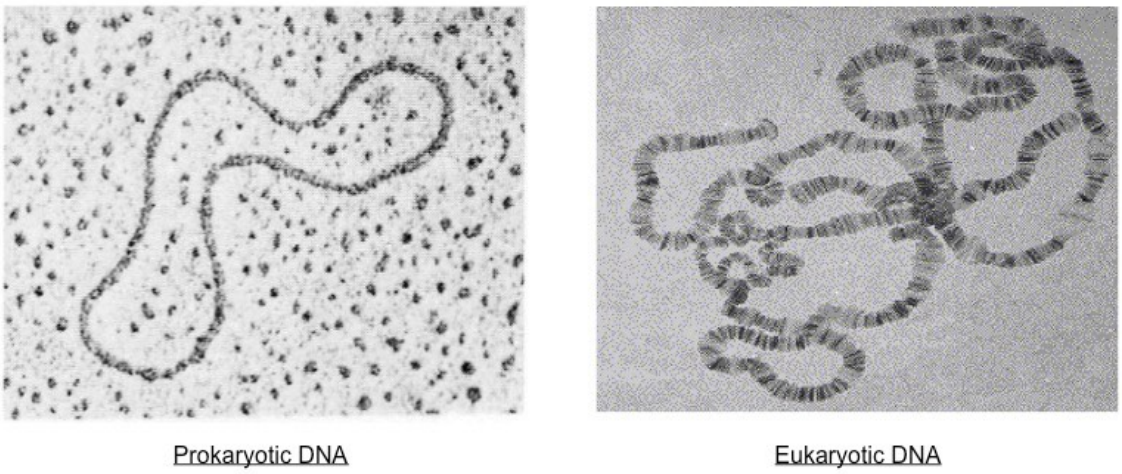
up until forming chromosomes. However, tight coiling of DNA limits cells ability to access DNA and to process it. Instead of being constantly coiled, the nucleic acid is usually found in a state called chromatin where some segments of acid are tightly reeled (heterochromatin), while other segments are entirely open (euchromatin). Euchromatin DNA is is highly accessible by the molecular complexes used by the cell and therefore is easier to manipulate with.

The amount and extent of packing are determined by a sell, to control which sections of the genome can be expressed and which cannot. It affects cellular function and appears to be the predominant cause of differentiating cells type, while having the same DNA.

1.1.4 Prokaryotic genome organization

A prokaryote is a cellular organism that lacks an envelope-enclosed nucleus. Prokaryotic genomes are very different from eukaryotic ones, in particular with regard to the physical organization of the genome within the cell. Although the word "chromosome" is used to describe the DNA–protein structures present in prokaryotic cells, this is a misnomer as this structure does not resemble an or-

dinary eukaryotic chromosome. In a typical prokaryote the genome is contained in a single, circular DNA molecule, localized within the nucleoid — the lightly staining region of the otherwise featureless prokaryotic cell.



Obr. 1.6: Comprasion of eukaryotic and prokaryotic DNAs

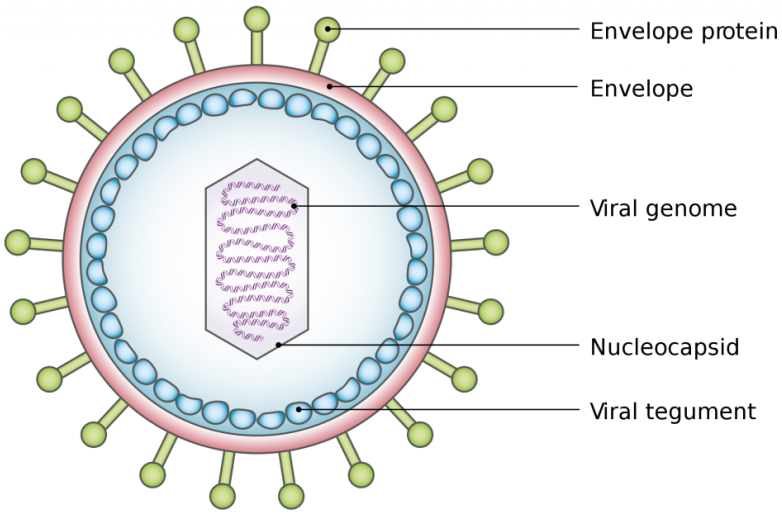
Most of what is known about the organization of DNA in the nucleoid comes from studies of E. coli. The first feature to be recognized was that the circular E. coli genome is supercoiled. Supercoiling occurs when additional turns are introduced into the DNA double helix (positive supercoiling) or if turns are removed (negative supercoiling). With a linear molecule, the torsional stress introduced by over- or underwinding is immediately released by rotation of the ends of the DNA molecule, but a circular molecule, having no ends, cannot reduce the strain in this way. Instead the circular molecule responds by winding around itself to form a more compact structure. Supercoiling is therefore an ideal way to package a circular molecule into a small space. Evidence that supercoiling is involved in packaging the circular E.coli genome was first obtained in the 1970s from examination of isolated nucleoids, and subsequently confirmed as a feature of DNA in living cells in 1981. In E. coli, the supercoiling is thought to be generated and controlled by two enzymes, DNA gyrase and DNA topoisomerase I.

Despite the conventional prokaryotic genome structure, an increasing number of linear genomes are being found. Linear molecules have free ends, which must be distinguishable from DNA breaks, so these chromosomes require terminal structures equivalent to the telomeres of eukaryotic chromosomes. In Borrelia burgdorferi and Agrobacterium tumefaciens, the real chromosome ends are distinguishable because a covalent linkage is formed between the 5‘ and 3‘ ends of the polynucleotides in the DNA double helix, and in Streptomyces coelicolor the

ends appear to be marked by special binding proteins.

1.1.5 Viral genome organization

To begin with, viruses are out-of-cell forms of life. It means, that their lifecycle and structure are generally less complicated comparely to others. Viruses can be extremely simple in design, consisting of nucleic acid surrounded by a protein coat known as a capsid. The capsid is composed of smaller protein components referred to as capsomers. The capsid+genome combination is called a nucleocapsid.



Obr. 1.7: Virus Characteristics.

Viruses can also possess additional components, with the most common being an additional membranous layer that surrounds the nucleocapsid, called an envelope. The envelope is actually acquired from the nuclear or plasma membrane of the infected host cell, and then modified with viral proteins called peplomers. Some viruses contain viral enzymes that are necessary for infection of a host cell and coded for within the viral genome. A complete virus, with all the components needed for host cell infection, is referred to as a virion.

While cells contain double-stranded DNA for their genome, viruses are not limited to this form. Moreover, as it was mentioned at the very beginning, apart from the dsDNA (double stranded DNA) viruses, there are also viruses with single-stranded DNA (ssDNA), double-stranded RNA (dsRNA), and single-stranded RNA (ssRNA). In this last category, the ssRNA can either be positive-sense (+ssRNA, meaning it can transcribe a message, like mRNA) or it can be negative-sense (-ssRNA, indicating that it is complementary to mRNA). Some viruses even start

with one form of nucleic acid in the nucleocapsid and then convert it to a different form during replication. In addition, these can be multipartite, meaning they consist of several RNA molecules.

In general, DNA viruses tend to be larger in size than RNA viruses and the single stranded genomes are smaller than those that are double-stranded. It is hypothesized that single-stranded virus are smaller because that type of molecule is more fragile than the double stranded molecule. This is generally true for both ssDNA and ssRNA viruses. Among the double-stranded genomes, these can either have 'small' or 'large' genomes. One major difference between the two genomes is the mechanism of DNA replication. Small genomes use host polymerase activities, whereas large genomes encode an own DNA polymerase.

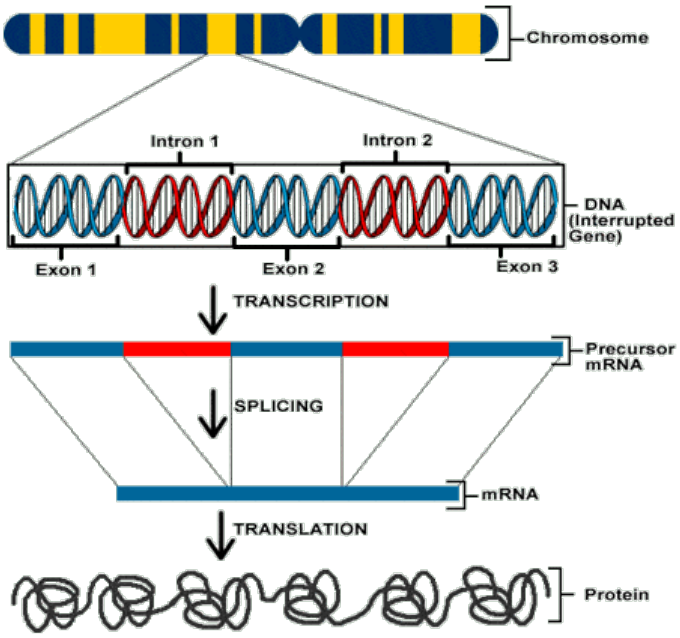
1.1.6 Genes: location and general structure

Gene is a sequence of nucleotides in DNA or RNA that encodes the synthesis of a gene product, either RNA or protein that have distinctive features. At present the nature of all of these specific features is not fully understandable, and sequence inspection is therefore not a foolproof way of locating genes[4].

Apart from the usual genes, pseudogenes are also present in different genomes. Pseudogenes are sequences of genomic DNA with such similarity to normal genes that they are regarded as non-functional copies or close relatives of genes[5]. They are formed in two ways:

- Classical duplicated pseudogenes are formed when genes that are tandemly duplicated accumulate mutations such that one of the genes becomes non-functional. These mutations may prevent transcription and/or translation.
- Processed pseudogenes are formed by the accumulation of mutations in a gene that has been retrotransposed to a new location. They are characterized by an absence of introns that are present in the parental gene.

Genes that code for proteins comprise open reading frames (ORFs) consisting of a series of codons (trinucleotides) that specify the amino acid sequence of the protein that the gene codes for. The ORF begins with an initiation codon — usually (but not always) ATG — and ends with a termination codon: TAA, TAG, or TGA. Searching a DNA sequence for ORFs that begin with an ATG and end with a termination triplet is therefore one way of looking for genes. The analysis is complicated by the fact that each DNA sequence has six reading frames, three in one direction and three in the reverse direction on the complementary strand.

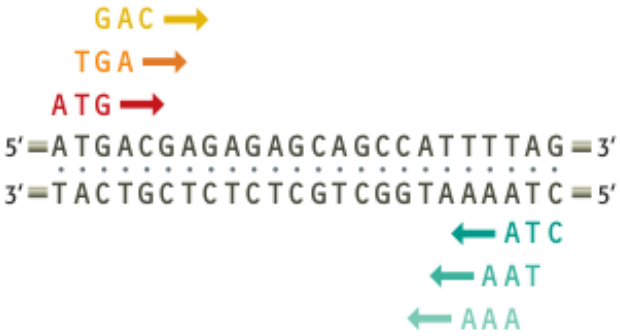


Obr. 1.8: Intron and exon organization during DNA-coherent processes at the cell's lifecycle.

The key to the success of ORF scanning is the frequency with which termination codons appear in the DNA sequence. If the DNA has a random sequence and a GC content of 50% then each of the three termination codons — TAA, TAG, and TGA — will appear, on average, once every 64 bp. If the GC content is greater than 50% then the termination codons, being AT – rich, will occur less frequently, but one will still be expected every 100–200 bp. This means that random DNA should not show many ORFs longer than 50 codons in length. Most genes, on the other hand, are longer than 50 codons: the average lengths are 317 codons for Escherichia coli, 483 codons for Saccharomyces cerevisiae, and approximately 450 codons for humans [6]. ORF scanning, in its simplest form, therefore takes a figure of, say, 100 codons as the shortest length of a putative gene and records positive hits for all ORFs longer than this.

With bacterial genomes, simple ORF scanning is an effective way of locating most of the genes in a DNA sequence. The real genes in the sequence cannot be mistaken because they are much longer than 50 codons in length. With bacteria the analysis is further simplified by the fact that the genes are very closely spaced and hence there is relatively little intergenic DNA in the genome (only 11% for E. coli). The most of bacterial genes do not overlap.

Although ORF scans work well for bacterial genomes, they are less effective for locating genes in DNA sequences from higher eukaryotes. This is partly be-



Obr. 1.9: Both strands are read in the 5′3′direction. Each strand has three reading frames, depending on which nucleotide is chosen as the starting position.

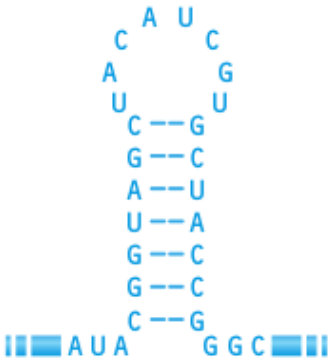
cause there is substantially more space between the real genes in a eukaryotic genome (for example, approximately 62% of the human genome is intergenic), increasing the chances of finding spurious ORFs. But the main problem with the human genome and the genomes of higher eukaryotes in general is that their genes are often split by introns (non-coding regions of gene), and so do not appear as continuous ORFs in the DNA sequence. Many exons (coding regions of gene) are shorter than 100 codons, some consisting of fewer than 50 codons, and continuing the reading frame into an intron usually leads to a termination sequence that appears to close the ORF. In other words, the genes of a higher eukaryote do not appear in the genome sequence as long ORFs, and simple ORF scanning cannot locate them.

In addition, since some viruses (mainly eukaryotic ones) have intron-exon stuctures in their genomes [7], ORF scanning is not the irrefragable method of locating genes among them.

ORF scanning is appropriate for protein-coding genes, but genes for functional RNAs such as rRNA and tRNA do not comprise open reading frames. Functional RNA molecules do, however, have their own distinctive features, which can be used to aid their discovery in a genome sequence.

The most important of these features is the ability to fold into a secondary structure, such as the cloverleaf adopted by tRNA molecules. These secondary structures are held together by base pairing not between two separate polynucleotides, as in the DNA double helix, but between different parts of the same polynucleotide — intramolecular base pairing.

In order for intramolecular base pairs to form, the nucleotide sequences in the two parts of the molecule must be complementary, and to produce a complex



Obr. 1.10: A typical RNA intramolecular base pairing structure.

structure such as the cloverleaf, the components of these pairs of complementary sequences must be arranged in a characteristic order within the RNA sequence. These features provide a wealth of information that can be used to locate tRNA genes in a genome sequence.

As well as tRNAs, rRNAs and some of the small functional RNAs also adopt secondary structures that have sufficient complexity to enable their genes to be identified without too much difficulty [8]. Other functional RNA genes are less easy to locate because the RNAs take up structures that involve relatively little base pairing or the base pairing is not in a regular pattern.

1.2 Existing Solutions For Genome Data Representing

With the rapid development of next-generation sequencing technologies, hundreds of thousands of genomes have been sequenced. All the sequence data as well as the annotations are collected in the genome databases and are publicly available through web portals such as the NCBI genome portal (http://www.ncbi.nlm.nih.gov/
and the EBI genome database website (http://www.ebi.ac.uk/Databases/genomes.html).

By systematic integration of genome sequences together with annotations generated through much heterogeneous data, genome browser provides a unique platform for molecular biologists to browse, search, retrieve and analyze these genomic data efficiently and conveniently. With a graphical interface, genome browser helps users to extract and summarize information intuitively from huge amount of raw complex data.

In general, genome browser can be divided into web-based browsers and standalone applications. Web-based genome browsers which usually are more usable in promoting biological research due to their data quality, flexible accessibility and high performance

- Firstly, dedicated organizations often collect and integrate high-quality annotation data into web-based genome browsers, providing plentiful up-to-date information for the community.
- Secondly, users can access them anywhere with a standard web browser, avoiding any additional effort of setting up local environment for application installation and data preparation.
- Thirdly, web-based genome browsers are usually installed on high performance servers and can support more complex and larger scale data types and applications.

1.2.1 Web-based genome browsers

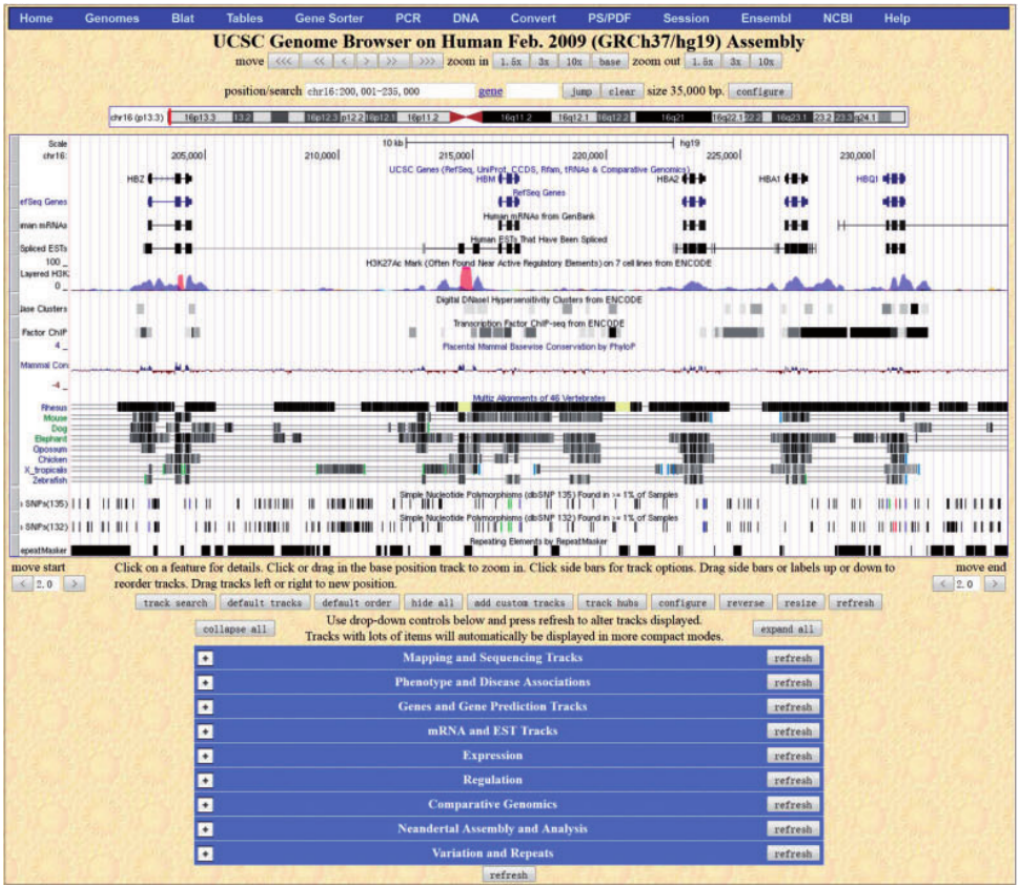
Currently, there are two types of web-based genome browsers.

The first type is the multiple-species genome browsers such as the Ensembl project [9], the UCSC genome database [10] and the NCBI Map viewer website [11]. These genome browsers promote cross-species comparative analysis. Most of them contain abundant annotations, covering gene model, transcript evidence, expression profiles, regulatory data, genomic conversation, etc. Each set of precomputed annotation data is called a track in genome browsers. The essence of a genome browser is to pile up multiple tracks under the same genomic coordinate along the Y-axis, thus users could easily examine the consistency or difference of the annotation data and make their judgments of the features of the genomic region.

The other type is the species-specific genome browsers which mainly focus on one model organism and may have more annotations for a particular species. Powered by the Generic Model Organism Database (GMOD) project (http://gmod.org/), dozens of open-source software tools are collected for creating and managing genome biological databases. The GBrowse framework [12] is one of the most popular tools in the GMOD project. Table 2 lists several mainstream web-based genome browsers, including Ensembl, the UCSC genome browser and the GBrowse, which are accessed by a large number of users worldwide.

1.2.2 Functionalities and features

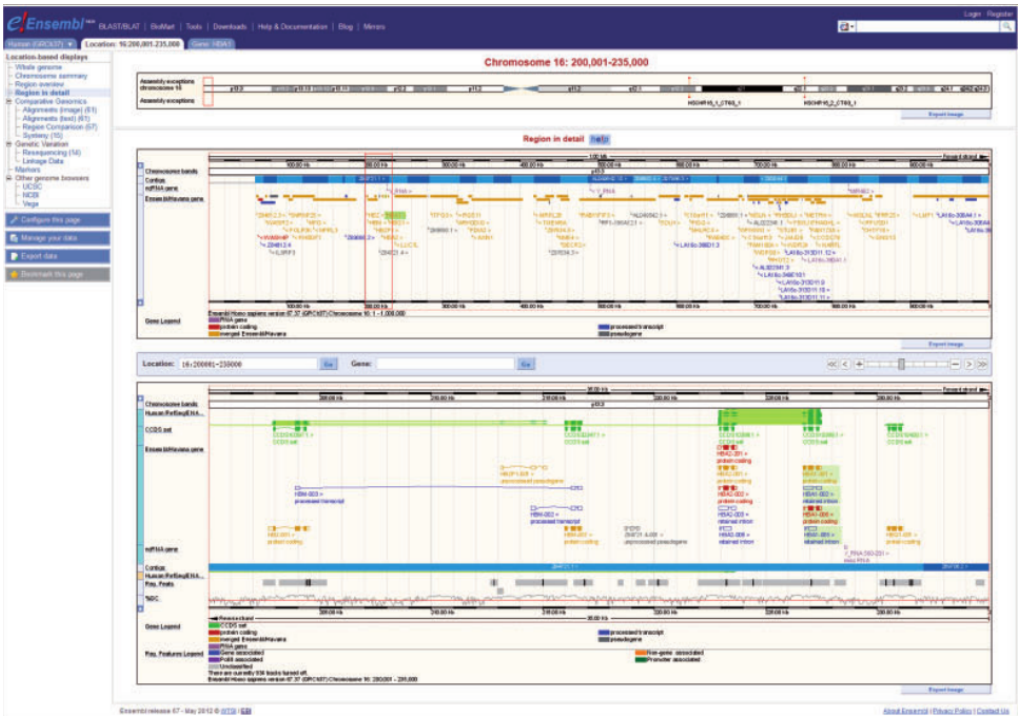
A web-based genome browser often provides a centralized or a set of databases to store different types of annotation data obtained from several organizations. The challenge for the general genome browsers is how to display this information properly for different genomic scales. Massive amounts of information need to be incorporated into the picture when a large genomic area is requested, which could overburden the server and the network. Furthermore, too many heavy and complicated details also disturb the user attention.



Obr. 1.11: The main user interface of the UCSC genome browser showing the default tracks in default order for the human alpha-globin gene cluster.

Being the one of the big players in genomic data visualization, The UCSC genome browser tries to solve the problem by providing multiple views for a track [5]. Every track can be viewed in different modes such as dense, or fully expanded or can be hidden. The user can go deeper on the dense track 25 to open it in full mode. There are many scales possible for the track display. The lowest is a single chromosome and the highest scale is the sequence of base pairs. The dense view of some tracks could be displayed to hide complicated details when

zooming out to a large area of a chromosome, so that the user has a broad picture of the selected chromosome region.



Obr. 1.12: The user interface of the Ensembl genome browser with default setting of the annotation tracks, showing the alpha-globin gene cluster. The graphical annotations are displayed in the main body divided in three sections from top to bottom.

The Ensembl genome browser provides the same user interface for each organism. The main body of the interface contains two panels. The left panel lists the main menu for location-based displays at different levels from whole genome, chromosome summary to region overview and region in detail. And links to comparative genomics, genetic variation as well as sequence markers are also provided. The main panel is arranged in three sections from top to bottom, providing different scales for users to analyze the genome. In addition to the location view, Ensembl provides separate pages to display various types of information, organized in a tabbed structure.

It is useful to have an overview of a large area of the chromosome and look into several small regions for details simultaneously. Putting paralogous genes together in one page could promote comparative analysis greatly. For example, NCBI sequence viewer supports users to view different regions inside the same chromosome, providing flexible multiple-panel-based navigating approach with different color cursors indicating the corresponding genomic locations. Meanwhile, ABrowse supports users to visualize multiple genome regions of different

Tabuľka 1.2: Main functions of the mainstream genome browsers.

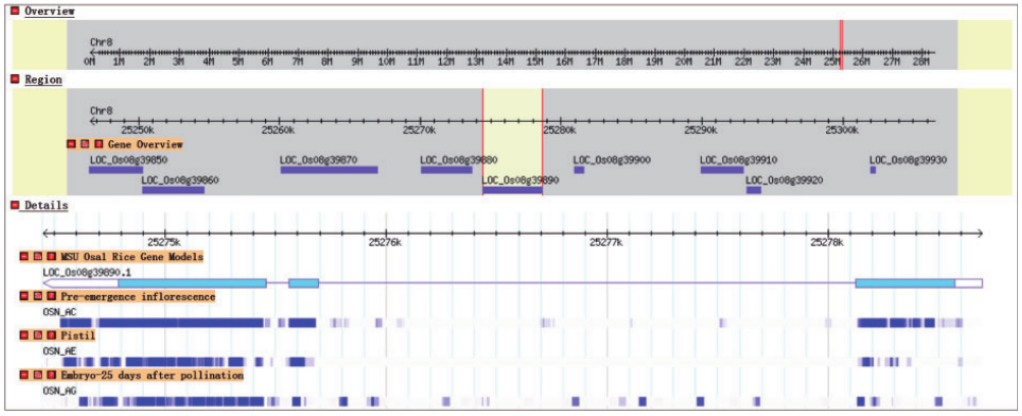
Features	UCSC	Ensemble	GBrowse
Visualization			
Annotation navigation	Page-based browsing, enabling dragging	Page-based browsing	Map-like browsing within a limited region
Multiple in-page windows	-	-	-
Data retrieval and analysis			
Query system	Table Browser	BioMart	-
User-oriented analysis	Direct data submission	-	Plugin tools
Machine-oriented interface	-	Through BioMart	Through BioDAS
Customization			
Upload user tracks	+	+	+
User-contributed contents	Session-based data restoring	Personal annotation, bookmark and group mechanism	-

chromosomes/genomes in separated in-page windows [11], while the separated windows are not fully operable as the main browsing canvas.

As was mentioned above, there are species-specific genome browsers are available online. The bright examples of such tools are the MSU rice genome browser and the Rice-Map genome browser [13].

In the MSU rice genome browser, users can search the OsSPL14 gene by specifying 'SPL14' in the search box. Powered by the GBrowse platform, the MSU rice genome browser provides annotation views with different scales, including chromosome overview, regional view and detailed view. The large-scale view provides a broader picture for users to inspect the upstream and downstream annotation conveniently. In the detailed annotation canvas, more than 82 annotation tracks are provided, covering gene model, transcript evidence, expression profiling, sequence alignment, genetic marker, SNP, RNA-Seq coverage and other genomic features. In addition to the basic gene model information, users may

inspect this gene in different development stages through various RNA-Seq expression data.



Obr. 1.13: The user interface of the MSU rice genome browser. The chromosome overview is displayed at the top, the regional view is shown at the middle and the bottom section is the detailed view for four annotation tracks.

In the Rice-Map genome browser different annotation tracks are organized in a map-like visualization canvas, with the name of opened tracks listed in the right panel. Besides basic gene annotation, there are rich annotation for cross genome alignments and conservation values, offering important clues to investigate this gene in other plants.

Obr. 1.14: The Rice-Map genome browser. The detailed information for individual entries is shown in the right panel, interpreting the data resource, entry location, sequence and function etc.

Since genome browsers are capable of providing the user with a plenty of different biology-specific information, the definitions of those terms are not provided by this thesis.

2 Syntetická časť

This chapter is focused on the applying of some widespread 2D genome visualization techniques to the SARS-CoV-2 genome. Since the topic is extremely complex, only some of existing methods are introduced. The comprehensive overview of those methods, their analysis and implementation are covered at the corresponding sections.

The second part of this chapter is aimed at the composition of functional but simple software which is capable of visualizing SARS-CoV-2 genome using previously described techniques and libraries.

2.1 FASTA, GFF and GBK formats

To begin with, the understanding of how the genomic data is being stored is the key to the proper visualization. Generally, the possible solutions for visualization can be divided in two separate categories: those, which use the very genome sequence and those, which use genome annotations [14].

The first category operates upon the raw DNA (RNA) sequence and is usually used to search for different patterns, tandem repeats, point mutations or to visually compare the genomes of related species. The raw data either it is DNA (RNA) or aminoacid sequence is usually stored in FASTA format (filename extensions .fasta, .fa, .fna) [15]. The source code below demonstrates the structure of FASTA file containing the SARS-CoV-2 DNA sequence.

```
>NC_045512.2 |Coronavirus 2 isolate Wuhan-Hu-1, complete genome
ATTAAAGGTTTATACCTTCCCAGGTAACAAACCAACCAACTTTCGATCTCTTGTAGATCT
GTTCTCTAAACGAACTTTAAAATCTGTGTGGCTGTCACTCGGCTGCATGCTTAGTGCACT
CACGCAGTATAATTAATAACTAATTACTGTCGTTGACAGGACACGAGTAACTCGTCTATC
```

Zdrojový kód 2.1: First 180 nucleotides from SARS-CoV-2 genome sequence in FASTA format. The description line, which begins with '>', contains information about the sequence. Following the initial line is the actual sequence itself in standard one-letter character string.

Meanwhile, the second one uses preprocessed and well-studied data [16], which is obtained from the raw sequence. Genome annotations contain locations of coding regions of genome and therefore they can be useful in fields coherent with genetics, synthesis of proteins, inheritance, etc.

```
##gff-version 3
#!gff-spec-version 1.21
#!processor NCBI annotwriter
#!genome-build ASM985889v3
#!genome-build-accession NCBI_Assembly:GCF_009858895.2
##sequence-region NC_045512.2 1 29903
##species https://www.ncbi.nlm.nih.gov/Taxonomy/Browser/wwwtax.cgi?id=2697049
NC_045512.2 RefSeq  region  1   29903   .   +   .   ID=NC_045512.2:1..29903;Dbxref=taxon:2697049;c
NC_045512.2 RefSeq  five_prime_UTR  1   265 .   +   .   ID=id-NC_045512.2:1..265;gbkey=5'UTR
NC_045512.2 RefSeq  gene    266 21555   .   +   .   ID=gene-GU280_gp01;Dbxref=GeneID:43740578;Name=
NC_045512.2 RefSeq  CDS 266 13468   .   +   0   ID=cds-YP_009724389.1;Parent=gene-GU280_gp01;Dbxre
NC_045512.2 RefSeq  CDS 13468   21555   .   +   0   ID=cds-YP_009724389.1;Parent=gene-GU280_gp01;D
NC_045512.2 RefSeq  mature_protein_region_of_CDS    266 805 .   +   .   ID=id-YP_009724389.1:1..18
NC_045512.2 RefSeq  mature_protein_region_of_CDS    806 2719    .   +   .   ID=id-YP_009724389.1:1
NC_045512.2 RefSeq  mature_protein_region_of_CDS    2720    8554    .   +   .   ID=id-YP_009724389.
NC_045512.2 RefSeq  mature_protein_region_of_CDS    8555    10054   .   +   .   ID=id-YP_009724389.
NC_045512.2 RefSeq  mature_protein_region_of_CDS    10055   10972   .   +   .   ID=id-YP_009724389.
NC_045512.2 RefSeq  mature_protein_region_of_CDS    10973   11842   .   +   .   ID=id-YP_009724389.
NC_045512.2 RefSeq  mature_protein_region_of_CDS    11843   12091   .   +   .   ID=id-YP_009724389.
NC_045512.2 RefSeq  mature_protein_region_of_CDS    12092   12685   .   +   .   ID=id-YP_009724389.
NC_045512.2 RefSeq  mature_protein_region_of_CDS    12686   13024   .   +   .   ID=id-YP_009724389.
NC_045512.2 RefSeq  mature_protein_region_of_CDS    13025   13441   .   +   .   ID=id-YP_009724389.
NC_045512.2 RefSeq  mature_protein_region_of_CDS    13442   13468   .   +   .   ID=id-YP_009724389.
NC_045512.2 RefSeq  mature_protein_region_of_CDS    13468   16236   .   +   .   ID=id-YP_009724389.
NC_045512.2 RefSeq  mature_protein_region_of_CDS    16237   18039   .   +   .   ID=id-YP_009724389.
NC_045512.2 RefSeq  mature_protein_region_of_CDS    18040   19620   .   +   .   ID=id-YP_009724389.
NC_045512.2 RefSeq  mature_protein_region_of_CDS    19621   20658   .   +   .   ID=id-YP_009724389.
NC_045512.2 RefSeq  mature_protein_region_of_CDS    20659   21552   .   +   .   ID=id-YP_009724389.
NC_045512.2 RefSeq  CDS 266 13483   .   +   0   ID=cds-YP_009725295.1;Parent=gene-GU280_gp01;Dbxre
NC_045512.2 RefSeq  mature_protein_region_of_CDS    266 805 .   +   .   ID=id-YP_009725295.1:1..18
NC_045512.2 RefSeq  mature_protein_region_of_CDS    806 2719    .   +   .   ID=id-YP_009725295.1:1
NC_045512.2 RefSeq  mature_protein_region_of_CDS    2720    8554    .   +   .   ID=id-YP_009725295.
NC_045512.2 RefSeq  mature_protein_region_of_CDS    8555    10054   .   +   .   ID=id-YP_009725295.
NC_045512.2 RefSeq  mature_protein_region_of_CDS    10055   10972   .   +   .   ID=id-YP_009725295.
NC_045512.2 RefSeq  mature_protein_region_of_CDS    10973   11842   .   +   .   ID=id-YP_009725295.
NC_045512.2 RefSeq  mature_protein_region_of_CDS    11843   12091   .   +   .   ID=id-YP_009725295.
NC_045512.2 RefSeq  mature_protein_region_of_CDS    12092   12685   .   +   .   ID=id-YP_009725295.
NC_045512.2 RefSeq  mature_protein_region_of_CDS    12686   13024   .   +   .   ID=id-YP_009725295.
NC_045512.2 RefSeq  mature_protein_region_of_CDS    13025   13441   .   +   .   ID=id-YP_009725295.
NC_045512.2 RefSeq  mature_protein_region_of_CDS    13442   13480   .   +   .   ID=id-YP_009725295.
NC_045512.2 RefSeq  stem_loop   13476   13503   .   +   .   ID=id-GU280_gp01;Dbxref=GeneID:4374057
NC_045512.2 RefSeq  stem_loop   13488   13542   .   +   .   ID=id-GU280_gp01-2;Dbxref=GeneID:43740
NC_045512.2 RefSeq  gene    21563   25384   .   +   .   ID=gene-GU280_gp02;Dbxref=GeneID:43740568;
NC_045512.2 RefSeq  CDS 21563   25384   .   +   0   ID=cds-YP_009724390.1;Parent=gene-GU280_gp02;D
```

Obr. 2.1: First 45 lines of SARS-CoV-2 genome annotation file. The rightmost part of the annotation is truncated on the image.

Each annotation of genome has 9 required fields:

1. Sequence ID
2. Source
   - Describes the algorithm or the procedure that generated this feature. Typically Genescane or Genebank, respectively
3. Feature Type
   - Describes what the feature is (mRNA, domain, exon, etc.)
4. Feature start

5. Feature end
6. Score
    - Values for sequence similarity or predictions
7. Strand (+ or -)
8. Phase
    - Indicates where the feature begins with reference to the reading frame
9. Atributes
    - A semicolon-separated list of tag-value pairs, providing additional information about each feature

This data is usually stored in GFF (General Feature Format) files. Filename extensions are .gff, .gff2. and .gff3 [17].

Apart from the GFF annotation and FASTA sequence files, the GBK (Genbank format) files are also widely used. The Genbank format allows for the storage of information in addition to a DNA/protein sequence. The screen grab shows various details, the first section includes the entry's LOCUS, DEFINITION, ACCESSION and VERSION and denoted by ORIGIN, the final detail is the actual sequence. These five elements are the essential parts of the GenBank format.

The non-essential parts of the entry contain what is commonly known as metadata, and can include more detailed information about the organism, cross-references to other databases, and even a list of publications in which this entry is featured in. The FEATURES part of the entry describes important characteristics of the entry's sequence such as presence of coding sequences, proteins, etc.

2.2 SARS-CoV-2 genome analysis and visualization

To perform the very analysis, BioPython and DNA features viewer packages will be used.

To visualze the SARS-CoV-2 genome the previously described genome data files are required. Both the FASTA and GBK files can be obtained at the NCBI webpage (https://www.ncbi.nlm.nih.gov/nuccore/MN908947).

```
LOCUS       MT350282              29903 bp    RNA     linear   VRL 17-APR-2020
DEFINITION  Severe acute respiratory syndrome coronavirus 2 isolate
            SARS-CoV-2/human/BRA/SP02cc/2020, complete genome.
ACCESSION   MT350282
VERSION     MT350282.1
KEYWORDS    .
SOURCE      Severe acute respiratory syndrome coronavirus 2 (SARS-CoV2)
  ORGANISM  Severe acute respiratory syndrome coronavirus 2
            Viruses; Riboviria; Nidovirales; Cornidovirineae; Coronaviridae;
            Orthocoronavirinae; Betacoronavirus; Sarbecovirus.
REFERENCE   1  (bases 1 to 29903)
  AUTHORS   Malta,F., Amgarten,D., de Oliveira,D.B.L., Araujo,D.B.,
            Machado,R.R.G., Santana,R.A.F., Mangueira,C.L.P., Durigon,E.L. and
            Pinho,J.R.R.
  TITLE     Direct Submission
  JOURNAL   Submitted (16-APR-2020) Clinical Laboratory, Hospital Israelita
            Albert Einstein, Av. Albert Einstein, 627/701, Sao Paulo, SP
            05652-000, Brazil
COMMENT     ##Assembly-Data-START##
            Assembly Method       :: Spades v. 3.11; Geneious v. 2020.1
            Coverage              :: 6001x
            Sequencing Technology :: Illumina
            ##Assembly-Data-END##
FEATURES             Location/Qualifiers
     source          1..29903
                     /organism="Severe acute respiratory syndrome coronavirus
                     2"
                     /mol_type="genomic RNA"
                     /isolate="SARS-CoV-2/human/BRA/SP02cc/2020"
                     /host="Homo sapiens"
                     /db_xref="taxon:2697049"
                     /lab_host="VERO E6 cells"
                     /country="Brazil"
                     /collection_date="2020-03-18"
     gene            266..21555
                     /gene="ORF1ab"
     CDS             join(266..13468,13468..21555)
                     /gene="ORF1ab"
                     /ribosomal_slippage
                     /codon_start=1
                     /product="ORF1ab polyprotein"
                     /protein_id="QJA41639.1"
                     /translation="MESLVPGFNEKTHVQLSLPVLQVRDVLVRGFGDSVEEVLSEARQ
                     HLKDGTCGLVEVEKGVLPQLEQPYVFIKRSDARTAPHGHVMVELVAELEGIQYGRSGE
                     TLGVLVPHVGEIPVAYRKVLLRKNGNKGAGGHSYGADLKSFDLGDELGTDPYEDFQEN
```

Obr. 2.2: First 45 lines of contents of GBK file corresponding to SARS-CoV-2 genome

2.2.1 Nucleotides distribution and GC-content

The analysis usually is stared by reading the DNA sequence:

Zdrojový kód 2.2: Python example

```
from Bio.SeqRecord import SeqRecord
from Bio import SeqIO
cov19 = SeqIO.read('MN908947.fna', "fasta")
```

One of the most essential genomic properties is G-C (or guanine-cytosine content) [18]. It is the percentage of nitrogenous bases in a DNA or RNA molecule that are either guanine (G) or cytosine (C). This measure indicates the proportion of G and C bases out of an implied four total bases, also including adenine and thymine in DNA and adenine and uracil in RNA.

GC-content may be given for a certain fragment of DNA or RNA or for an

entire genome. When it refers to a fragment, it may denote the GC-content of an individual gene or section of a gene (domain), a group of genes or gene clusters or a non-coding region [19].

GC-content is usually expressed as a percentage value, but sometimes as a ratio. GC-content percentage is calculated as:

$$\frac{G+C}{A+T+G+C} * 100\%$$

The distribution of the nucleotides (A,T,C,G) over the Covid19's DNA can be computed by the attached script.

Zdrojový kód 2.3: The script for computing the distribution of the nucleotides over the SARS-CoV-2 genome.

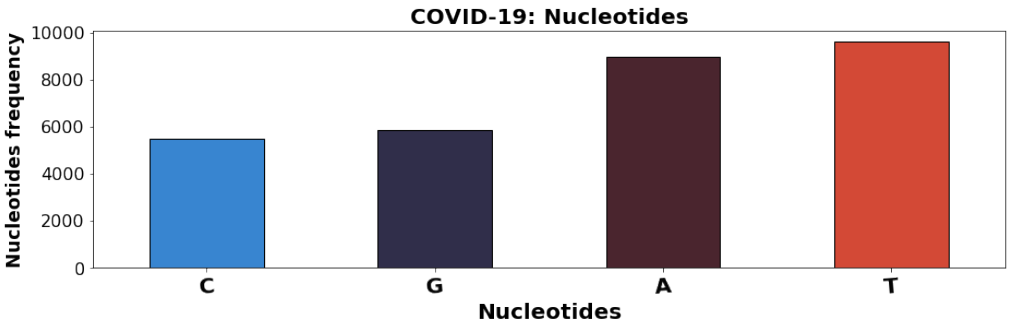
```
#Count the nucleotides frequency in the DNA
DNA = SARS_Cov_2_DNA
nucleotides = {}
for n in DNA:
    if n in nucleotides:
        nucleotides[n] += 1
    else:
        nucleotides[n] = 1

#Create a dataframe
nts = pd.DataFrame(data=nucleotides,
                           index=[0]).T.reset_index()
nts = nts.rename(columns={0: 'frequency',
                           'index': 'nucleotides'})
nts = nts.sort_values(by=['frequency'], ascending=True)
```

The first observation is that the frequency of the nucleotides A (8954) and T (9594) is higher than the frequency of C (5492) and G (5863). Therefore, the GC-content is 37.97%. In comprasion, the GC-content of eukaryotes, such as vertebrates, including humans, can be up to 60% [20].

2.2.2 Gates's method

The 2D methods are primarily based on the Cartesian coordinate system and the very representation is a set of dots or vectors corresponding to different genome properties; Gates's method is a typical instance of 2D visualization techniques that operates upon the raw genomic data. Therefore, the FASTA file is being processed during the visualization.

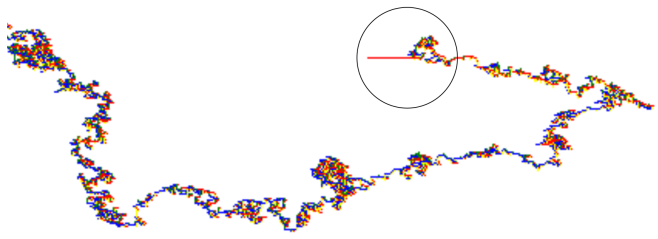


Obr. 2.3: Diagram showing the distribution of SARS-CoV-2 nucleotides.

It is understandable and was chosen as one of methods to implement because it was accidently reinvented by me during the thesis creation.

The four nucleic acid bases are assigned to the four axes of a 2D Cartesian coordinate system. A given sequence is plotted according to distribution of its bases in the corresponding direction; in computations for this note adenine (A) was assigned to the negative x-axis, cytosine (C) to the positive y-axis, guanine (G) to the positive x-axis and thymine (T) to the negative y-axis. The weighted average of the x- and y-coordinates of each point of a sequence of length N represents the centre of mass. The Euclidean distance between the origin and the centre of mass provides a quantitative graph descriptor, termed as the graph radius (gR).

This method can be used to search for similarities among related species and for searching patterns in the particular ones. For instance, after the visualization of SARS-CoV-2 genome sequence using the Gates's method, the sequence of 33 adenine (red) nucleotides is easily distinguishable over the whole genome.



Obr. 2.4: A part of SARS-CoV-2 genome representation using Gates's method. The long sequence of 33 adenine nucleotides is marked at black circle.

Another instance of finding patterns in DNA sequence using the Gates's method is an attempt of visualization nucleotide sequence of the first chromosome of Encephalitozoon Intestinalis, which is considered to posess the smallest eukaryotic genome. The sequence can be easily obtained using the NCBI resourse at the following address https://www.ncbi.nlm.nih.gov/nuccore/CP001942.

```
ACAGTGAACAATGCTAGGGAGAGCTGCCTATATGGAAGAGCCCTAATGTGTAAAATTAAT
TTTAGTAGTGCTATCCCCATGTGATTTTAATAGCTTCTTAGGAGAATGACAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAA
```

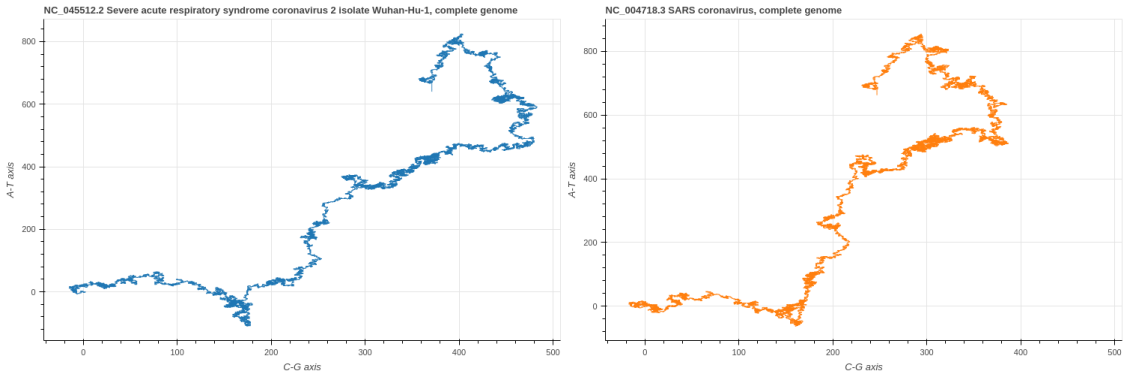
Obr. 2.5: Last 143 nucleotides of SARS-CoV-2 virus genome at the FASTA file with the sequenced genome. The long sequence of 33 adenine nucleotides is highlighted.

After the visualization of the mentioned chromosome, the first observation is that both ends of the sequence are almost identical. The only difference, except for point mutations, is that they are composed of complement nucleotides.

Obr. 2.6: Part of the first chromosome visualization of Encephalitozoon Intestinalis. The identical sequences are shown at the middle.

The closest relative of modern +ssRNA SARS-CoV-2 virus is SARS-CoV virus, that was the cause of 2004 SARS outbreak. The comprasion of SARS-CoV and SARS-CoV-2 genome sequences may be considered as the bright example of applying this method.

The very comrasion can be done using the Squiggle, a two-dimensional DNA sequence visualization library [21].



Obr. 2.7: Comprasion of SARS-CoV-2 (from the left) and SARS-CoV (from the right) genomes using Gates's method.

As it visible at the provided plot, both of the viruses resemble each other in the terms of the nucleotide sequence [22].

After processing the sequences of both viruses with the help of pairwise2 algorithm, the percentage of similarity between them is equal to 83.34%.

The PairWise algorithm is a variant of the Smith–Waterman algorithm best local alignment algorithm. These algorithms all belong to the class known as minimal string edit algorithms. It was chosen to compare sequences due to the way he aligns sequences. The main differences between PairWise and other alignment algorithm is that, besides normal penalties such as Gap Opening Penalty (GOP), Gap Extension Penalty (GEP) and Match, PairWise introduced two new penalties called Frame Opening Penalty (FOP) and Frame Extension Penalty (FEP) [23].

The very comprasion can be easily done using BioPython library, as demonstrated at the attached code.

Zdrojový kód 2.4: Pairwise2 algorithm using BioPython. COV1.seq and COV2.seq are the DNA sequences of SARS-CoV and SARS-CoV-2 viruses. Two arguments of the very alignment are provided to reduce the alignment comlexity and time.

```
from Bio import pairwise2

alm = pairwise2.align.globalxx(COV1.seq, COV2.seq,
            one_alignment_only=True, score_only=True)
print('Similarity (%):', alm / len(COV2.seq) * 100)
```

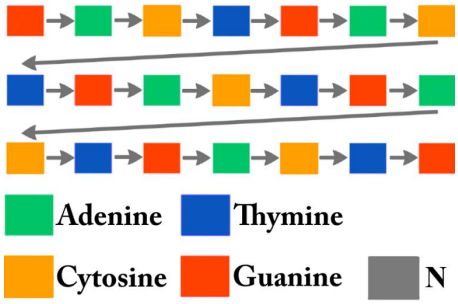
However, the most important disadvantage of Gates's method is degeneracy, meaning that a visualization is not necessarily unique. For example, TGAC is a square (up, right, down, and left), but so is GTCA (right, up, left, down).

2.2.3 2D Matrix method

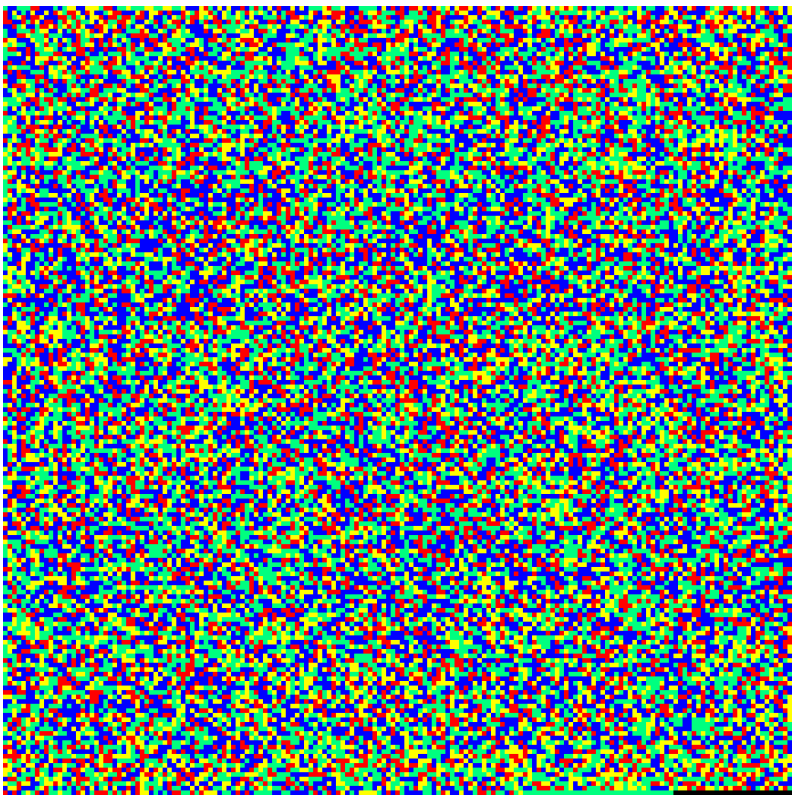
Another interesting method for DNA visualization, that operates upon the raw DNA squence is 2D matrix method. It's aim is to plot whole genome sequence in the square image of predefined size. Plotting is done from left corner to right corner untill the end of the line, and moving to the left side of the new one afterwards.

Each nucleotide is represented by a pixel (square) of particular color.

This method might be useful in finding tandem repeats [24] without the detailed examination by a machine since they might be visually detectable. Moreover, this method plots every genome in the picture of fixed size, that might be used to identify the very sequence uniquely.



Obr. 2.8: The plotting of DNA into the two-dimensional matrix.



Obr. 2.9: Visualization of SARS-CoV-2 genome using the 2D Matrix method. Since the most fluent genome sequence is composed of 29903 nucleotides and the shown matrix contains 29929 positions (173 on each side), the black squares at the right bottom corner represent empty space that was not used for visualization.

However, in the case of point mutations, or even significant ones, the difference in the plotted genome might be hardly distinguishable without any machine examination.

2.2.4 2D Matrix method improvement

This thesis suggests new visualization technique which is capable of visualizing each genome uniquely using the hash-function [25], which might be a solution to the menioned above disadvantage.

The main idea of the visualization technique is a recursive algorithm that partitions the image into smaller parts coloring each one depending on the pevious one. This prevents the random noise that might appear by breaking it into tiny bits and coloring each one independently. Instead, there are larger regions which maintain some continuity even though their constituent parts diverge.

The recursive algorithm is composed of three functions:

- Function, which achieves a hash of the nucleotide sequence using sha256 algorithm.
- Function, which recursively partitions the initial empty image in $1/8$ parts. It is done recursively 8 times to each of the initial partitions.
- Function, which colors each partition according to the hash and inserts each partition above the bigger one. The opacity parameter that is computed of each partition size and used to prevent the overlapping between colors of smaller partitions and bigger ones (that are colored before).

Listing 2.5 demonstrates the console output during the creation of image which is 512x512 pixels in size. The width and heigh values represent the sizes of partitioned images that are being colored, and the opacity value shows the opacity during each partition overlaps. The images generated to illustrate this method were using the opacity equation defined as:

$$opacity = 256 * (level/2)/2^{(level-1)}$$

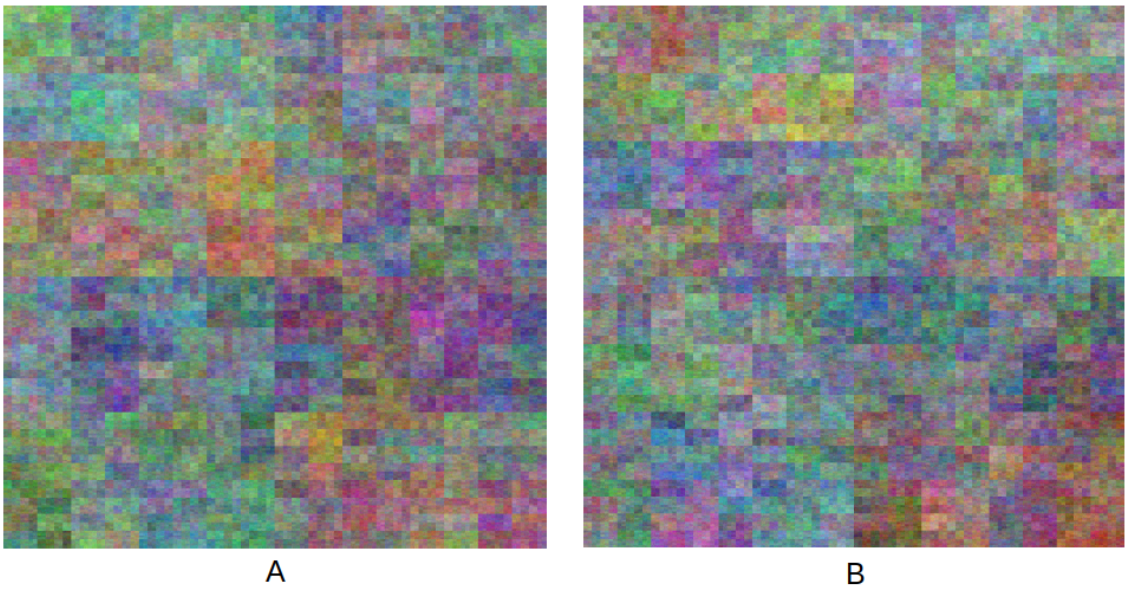
where level represents the current depth of recursion. However, this equation can be changed to intoduce another coloring schemes.

Zdrojový kód 2.5: Console output produsded during the visualization using the improved 2D Matrix method.

```
seed: 223dd4d5c80b98e69f9f8536afa54b...
width, height: 256.0, 256.0
opacity: 50%
width, height: 128.0, 64.0
opacity: 37%
```

```
width, height: 32.0, 32.0
opacity: 25%
width, height: 16.0, 8.0
opacity: 15%
width, height: 4.0, 4.0
opacity: 9%
width, height: 2.0, 1.0
opacity: 5%
```

As it is visible on the figure 2.10, this method lets different genomes be easily distinguished with the naked eye.



Obr. 2.10: Visualization of original SARS-CoV-2 genome [A] and the same genome with a point mutation [B] at the second nucleotide (T changed on G) using the improved 2D Matrix method.

Because of the recursive nature of the algorithm, the image can be generated at smaller or larger sizes and maintain the same level of detail.

However, among the main disadvantages are the small practical usage and a limit regarding the size of a genome. Currently, the method suports the visualization of genomes that are less than 262145 nucleotides long.

2.2.5 Aminoacids retrieval

For the next visualization the raw DNA sequence must be converted to the sequence of aminoacids. There are 61 codons (trinucleotides) for 20 amino acids, and each of them is "read"to specify a certain amino acid out of the 20 commonly found in proteins. Each aminoacid can be written as a letter of latin alphabet.

Therefore, the sequence of aminoacids is represented by the sequence of letters A-V.

One codon, AUG, specifies the amino acid methionine and also acts as a start codon to signal the start of protein construction. There are three more codons that do not specify amino acids: UAA, UAG, and UGA, tell the cell when a polypeptide is complete. All together, this collection of codon-amino acid relationships is called the genetic code, because it lets cells "decode" an mRNA into a chain of amino acids.

Before converting the DNA sequence into the aminoacid one, firstly, has to be transcribed into the mRNA molecule [26] with the help of transcribe() function. Luckily, with the translate() function, BioPython does translate the mRNA to amino acids chains. Chains are separated with a * which is the stop codon (UAA, UAG and UGA).

Zdrojový kód 2.6: Transcription and translation using BioPython

```
cov_DNA = covid19.seq
cov_mRNA = covid_DNA.transcribe()
cov_aa = covid_mRNA.translate()
```

SARS-CoV-2 genome contains 9967 aminoacids separated with stop codons * or, in other words, 775 amino acid chains. It's worth to mention that not all the amino acids sequences are proteins. Only the sequences with more than 20 amino acids code for functional proteins. The short amino acid sequences are oligopeptides and have other functionalities. The next step is to filter obtained sequences in such a way, that only long ones will remain to focus only on proteins.

After the removal of short proteins, only 5 of remaining ones satisfy the length condition (source code 2.6) and are provided at the table 2.1 .

The easiest way to verify results is to find the protein sequences already available in the databases that are the most similar to obtained protein sequences. The BLAST search was used for these purposes.

BLAST (basic local alignment search tool) is an algorithm and program for comparing primary biological sequence information, such as the amino-acid sequences of proteins or the nucleotides of DNA and/or RNA sequences. A BLAST search enables a researcher to compare a subject protein or nucleotide sequence (called a query) with a library or database of sequences, and identify library sequences that resemble the query sequence above a certain threshold.

Zdrojový kód 2.7: Filtering aminoacid sequences and storing them in a dataframe

Tabuľka 2.1: Obtained protein sequences of SARS-CoV-2 genome that are composed of more than 50 aminoacids.

	Aminoacid sequence	**Protein length**
1	CTIVFKRVCGVSAARLTPCGTGTSTDVVYRAFDIYND...	2701
2	ASAQRSQITLHINELMDLFMRIFTIGTVTLKQGEIKD...	290
3	TNMKIILFLALITLATCELYHYQECVRGTTVLLKEPC...	123
4	AQADEYELMYSFVSEETGTLIVNSVLLFLAFVVFLLV...	83
5	QQMFHLVDFQVTIAEILLIIMRTFKVSIWNLDYIINL...	63

```
Proteins = covid_aa.split('*')

#Remove porteins with less than than 50 amino acids
for i in Proteins[:]:
    if len(i) < 50:
        Proteins.remove(i)
#Store the protein sequences in a pandas dataframe

proteins=pd.DataFrame(Proteins)
proteins['amino acid sequence'] = proteinas[0].apply(str)
proteins['Protein length'] = proteinas[0].apply(len)
proteins.rename(columns={0: "sequence"}, inplace=True)
pro=proteins.drop('sequence', axis=1)
pro=pro.sort_values(by=['Protein length'], ascending=False)
```

After searching for the 83 aminoacid chain using BLAST, the results have shown that it has 100% similarity with Envelope small membrane protein that belongs to SARS-CoV-2 genome. The results of performed BLAST searches for other obtained proteins are attached at the table 2.2 . It is worth mentioning, that the highest similarities were found among other coronavirus species.

2.2.6 ORF identification and visualization

The next visualization operates upon the preprocessed data stored at the GenBank annotation file. As it was mentioned at the first chapter of this thesis, ORF scanning is not a full proof of finding gene locations since not every ORF is a gene beginning. However, the longer ORF is, the likely it is a part of gene. [27]

The identification of coding sequences (CDS) is an important step in the functional annotation of genes. A typical CDS starts with ATG and ends with a stop

Tabuľka 2.2: Results of comprasion between obtained protein sequences of SARS-CoV-2 using BLAST.

	Protein length	**DB:ID**	**Organism**	**Protein**	**Match**
1	2701	P0C6X7	Replicase polyprotein 1ab	Replicase polyprotein 1ab	96%
2	290	Q0Q474	Bat coronavirus 279/2005 (BtCoV)	Protein 3	75%
3	123	Q3I5J0	Bat coronavirus Rp3/2004	Protein 7a	89%
4	83	P0DTC4	Human SARS coronavirus (SARS-CoV-2)	Envelope small membrane protein	100%
5	63	Q3I5J1	Bat coronavirus Rp3/2004	Non-structural protein 6	69%

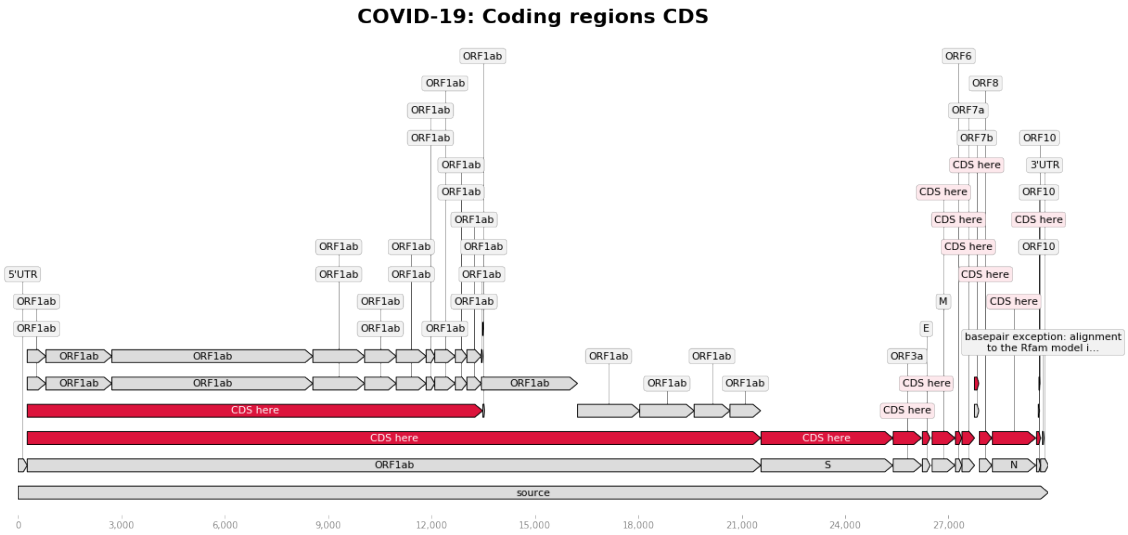
codon. CDS is a sequence of nucleotides that corresponds with the sequence of amino acids in a protein. Therefore, the amino acids analysis and retrieval (done in section 2.2.5) was necessary to identify coding regions of genome. The SARS-CoV-2 genome encodes as far as 50 non-structural, structural, and accessory proteins.

The source code 2.7 is an output a script for visualization that finds the ORFs in the SARS-CoV-2 genome using genome annotation file. The minimum protein length is set to 200 amino acids to retrieve only long ORFs.

Zdrojový kód 2.8: Positions of SARS-CoV-2 genome CDSs

```
PKGKMESLVPGFNEKTH...FAV - length 4409, strand 1, 253:13483
CTIVFKRVCGVSAARLT...VNN - length 2701, strand 1, 13449:21555
LEKTTELLFLVMFLLTT...HYT - length 1293, strand 1, 21502:25384
ASAQRSQITLHINELMD...VPL - length 290, strand 1, 25347:26220
SSGLNELNIILVFLFGT...LVQ - length 243, strand 1, 26459:27191
RSCCFRFHLNEQTKMSD...TQA - length 433, strand 1, 28231:29533
```

The Covid-19 genome has 6 ORFs with more than 200 amino acids. Figure 2.11 demonstrates all the ORFs and CDSs retrieved from the SARS-CoV-2 genome using BioPython.



Obr. 2.11: Coding regions of SARS-CoV-2 genome are highlighted in red among other ORFs. They include the ORF1ab, ORF3a, S protein, M protein and N protein. The visualization is done using BioPython

The results of comparing obtained sequences to the existing ones are performed using BLAST and listed at the table 2.3 .

Tabuľka 2.3: BLAST search results for SARS-CoV-2 ORFs.

	ORF length	**DB:ID**	**Protein**	**Organism**	**Match**
1	4409	P0C6U8	Replicase polyprotein 1a	Human SARS coronavirus (SARS-CoV)	80%
2	2701	P0C6X7	Replicase polyprotein 1ab	Human SARS coronavirus (SARS-CoV)	96%
3	1293	P59594	Spike glycoprotein	Human SARS coronavirus (SARS-CoV)	76%
4	290	Q0Q474	Protein 3	Bat coronavirus Rp3/2004	95%
5	243	Q0Q472	Membrane protein	Bat coronavirus Rp3/2004	92%
6	433	P59595	Nucleoprotein N	Human SARS coronavirus (SARS-CoV)	91%

2.3 Software composition

The software developed during this thesis appears to be a collection of scripts used to visualize SARS-CoV-2 genome in previously described methods. In general, unlike the web based genome browsers where the computations are done at the server side, the suggested software represent a simple stand-alone console application. No significant computations are being done and, therefore, the application do not satisfying of demanding system requirements.

The software is written mainly using the BioPython package in Python 3.8 . All the packages that are required are mentioned at the project documentation.

The main idea of a program is to let the user choose which information regarding the virus genome he wants to visualize. It consists of 9 modules which have own role at the very process of visualization:

1. **Main Module** is the core of the program. It is responsible for providing user with navigation within the program. It handles console input and output, suggests available methods of visualiation and retrieves details required for their performance.

2. **Sequence Collector** is responsible for downloading all the required sequences and annotation files from NCBI database.

3. **Statistics Generator** obtains the statistical data such as GC-content and nucleotide / aminoacid distribution. User is able to choose the region of genome which statistics should be collected and to choose whether he wants his data to be saved into particular file.

4. **Gates Visualization** performs a visualization using the Gates's method into a .png file. User is able to adjust color dependencies.

5. **2D Matrix** module plotes the selected genome into a 2D matrix into a .png file. The size of an output picture is computed automatically.

6. **2D IMatrix** module plotes a genome into a 2D matrix of the selected size using the hash function algorithm into a .png file. The bigger size, the more RAM memory is needed.

7. **Protein Plotter** generates protein sequences of a genome according to the nucleotide sequence. User is able to remove those, that do not satisfy length conditions and is able to chose whether to print the results or to store them into a .csv file.

8. **ORF Plotter** generates an image of SARS-CoV-2 ORFs and GC-content ratio within the genome.

9. **Comprasion** module performs a comprasion of selected genomes. The similarity percentage is obtained using the pairwise2 algorithm.

Modules do not able to interact with each other, but every of them can be called from the main one.

Each of the modules is described at the documentation and accompanied with the input and output examples. Moreover, every module contains different runtime tests which prevent unexpected program behaviour.

At the moment all the modules, except for ORF Plotter, support processing of various genomes. However, it is not recommended to use Protein Plotter with genomes that contain complex intron-exon structures due to their complexity. In such a case, results might differ significantly from the real characteristics of genome.

3 Vyhodnotenie

During the first part of this thesis I have analyzed the general genome structure of different organisms, DNA and RNA from the molecular, biological and informatical point of view.

The spatial structure of DNA and its differences were listed at table 1.1 .

The genomes of eukaryotes, prokaryotes and viruses were described in corresponding sections. The description of those genomes includes their unique properties, such as intron-exon structures mainly present in genomes of eukaryotes and gene clusters present in genomes of prokaryotes. The genes, their types, locations, functions and patterns that allow to find them (ORFs) were analyzed and properly described.

The next step was to classify existing solutions for genome data representation in order to achieve the understanding how the modern software works and what it can suggest to users worldwise. They can be classifed as *web-based* and *standalone applications*.

Moreover, two categories of existing genome browsers were precisely described:

- **Species-independent** solutions that are capable of visualizing any genome, including Ensemble genome browser, UCSC genome browser and GBrowse framework were analyzed and compared (table 1.2).

- **Species-specific** solutions that are aimed at visualization of particular species, including MSU rise genome browser and Rice-Map genome browser.

After, due to the extreme complexity of prokaryotic and eukaryotic genomes and due to the world pandemic, the SARS-CoV-2 virus genome was chosen to be visualized.

During the second part of this thesis, in order to understand how and where genome data is being stored, I have performed a comprehensive analysis of formats that are usually used to store genome-coherent information: FASTA file format, GFF and GenBank (GBK) file formats. The SARS-CoV-2 related files were

used as an example for the very analysis.

I came to a conclusion, that genome visualization can be done in two ways:

- The first way is to visualize raw data which is a nucleotide sequence for DNA and RNA or an amino acid sequence (using information in FASTA files).
- The second way is to visualize previously processed and well studied data, that contain gene locations, ORFs positions, etc. and which is stored in the genome annotation files (GFF and GenBank files).

After that, I have performed the analyzis and visualization of SARS-CoV-2 genome using the following methods.

Nucleotides distribution and GC-content analysis of FASTA file contents have shown, that SARS-CoV-2 genome is composed of 29903 nucleoties (basic units of genome). The distribution of them was the following: adenine (A) appeared at the genome sequence 8954 times, thymine (T) - 9594, cytosine (C) - 5492 and guanine (G) - 5863 times. By knowing this data, I have computed the GC-content property of a genome that was low and appeared to be 37.97%.

This property is important, because it shows that SARS-CoV-2 genome has small number of ORFs (genes), because they are usually begin at GC-rich regions of genome.

Gates's method visualization was chosen to be performed, since I have accidently rediscovered it while thinking of how genome can possibly be visualized. It also processes the FASTA file. After performing this visualization, I have noticed that strange sequence of 33 adenine (A) nucleotides ends SARS-CoV-2 genome (figure 2.4). To make sure, that no errors were made during the visualization I have checked the very FASTA sequence and came up with the conclusion that the visualization was done in the right way.

Despite the degeneracy, which is the main disadvantage of the method, it can plot DNA patterns without machine examination, as I have shown by visualizing the first chromosome of the smallest known eukaryotic genome that belongs to *Encephalitozoon Intestinalis* (figure 2.6).

To understand whether this method is suitable for visual comprasion of genomes of related species, I have visualized SARS-CoV-2 genome and genome of its closest relative SARS-CoV-1 virus next to each other. I have made a hypothesis that they must be very similar since they resemble each other in the terms of achieved visualization (figure 2.7), and to prove it, I have run the pairwise2 algorithm that confirmed, that their percentage of similarity is equal to 83.34%. I have

chosen that algorithm since it is among the best alignment algorithms.

2D Matrix method visualization was chosen to be performed, since tandem repeats in a genome sequence might be visually distinguishable without any machine examination. However, after the very visualization, I did not manage to find any of them (figure 2.9). This method also requires FASTA file with sequence.

The main disadvantage of this method is that point mutations in genome, or even significant ones, are almost not noticabe without comprehensive analyzis of the obtained image.

2D Matrix method improvement was intoduced by me to cope with the disadvantage of the previous method by using a hash-function.

I proved it by comparing original genome of SARS-CoV-2 and the same genome with a point mutation at the second nucleotide (T was substituted with G) using my method (figure 2.10).

Amino acid retrieval was performed to obtain SARS-CoV-2 proteins for required for the next method.

After achievment of proteins from the DNA sequence stored in a FASTA file, I have compared the results with existing SARS-CoV-2 proteins by performing a BLAST search. The compasion have shown, that almost no mistakes were made since the similarity ratio was very high. Because of my interest, I have also compared them with proteins of relative species that have also shown a high ratio of similarity (table 2.2).

ORF identification and visualization was performed using the GenBank annotation file in order to visualize those parts of SARS-CoV-2 genome, that are probably genes.

After searching for ORFs locations in the GBK file and comparing their indexes with the proteins obtained previously, removing the short ones, I have achieved 6 coding sequenses of genome (genes). The visualiation of achieved genes among other ORFs is visible at the figure 2.11 .

To verify the results, BLAST search was performed (table 2.3). It has shown, that I have visulized those genes properly, since they have a high similarity with the existing ones.

The next and the last step was to combine all the code that was used into a simple stand-alone console application which is capable of visualizing SARS-CoV-2 genome using previously described techniques. The detailed software architecture is described at the corresponding section.

From the main disadvantages of the developed program I can admit that it is rather simlple, has no graphical interface and, at the moment, not all methods

support visualiaton of any genome. Therefore, the developed visualiation tool can be claffied as *species-specific*.

4 Záver

During the work on this thesis I have dived into the field of bioinformatics, comprehensively analyzed the genome structure of different organisms, compared existing solutions and developed a new one.

The developed program operates upon the FASTA and GenBank files is capable of visualization SARS-CoV-2 genome and (partially) genomes of other organisms using different 2D visualization techniques. Among them there is also an Improved 2D Matrix Method that was developed by me.

Future works and improvements can be mainly aimed at intoducing new visualization techniques, improving existing ones, adding a support for other genomes to make a tool universal and at providing a program with a graphical interface.

Literatúra

1. REECE, Richard J. *Analysis of Genes and Genomes*. New York: Wiley, 2004. ISBN 978-0-470-84380-2.
2. BROWN, T. A. *Introduction to Genetics: A Molecular Approach*. 1. vyd. New York : Garland Science Pub., 2011. ISBN 9780815365099.
3. YOUNGSON, Robert M. *Collins Dictionary of Human Biology*. Glasgow: HarperCollins, 2006. ISBN 978-0-00-722134-9.
4. BROWN, T. A. *Genomes 3*. New York: Garland Science Pub., 2007. ISBN 978-0-470-84380-2.
5. SANDY B. PRIMROSE, Richard Twyman. *Principles of Gene Manipulation and Genomics*. 7. vyd. New York: Wiley, 2006. ISBN 978-1-405-13544-3.
6. FICKLETT, J. W. Finding genes by computer: the state of the art. 1996, s. 12:316–320.
7. LEWIN, Benjamin. *Genes (3rd ed.)* New York: Wiley, 1987. ISBN 0-471-83278-2.
8. D, Hebenstreit. "Methods, Challenges and Potentials of Single Cell RNA-seq". *Biology*. 2012. Dostupné z DOI: doi:10.3390/biology1030658.
9. HUBBARD T. Barker D., Birney E. The UCSC genome browser database. *Nucleic Acids Research*. 2003, roč. 31, č. 1, s. 51–54.
10. KAROLCHIK D. Baertsch R., Diekhans M. The Ensembl genome database project. *Nucleic Acids Research*. 2002, roč. 30, č. 1, s. 38–41.
11. WOLFSBERG, T. G. Using the NCBI Map Viewer to browse genomic sequence data. *Current Protocols in Human Genetics*. 2011, roč. 18, č. 18, s. 15.
12. DONLIN, Maureen J. Using the Generic Genome Browser (GBrowse). *Current Protocols in Bioinformatics*. 2009, roč. 28, č. 1, s. 9.9.1–9.9.25. Dostupné z DOI: 10.1002/0471250953.bi0909s28.
13. WANG J. Kong L., Zhao S. Rice-Map: a new-generation rice genome browser. *BMC Genomics*. 2011, roč. 12, č. 1, s. 165.

14. MAYYA SEDOVA, Lukasz Jaroszewski. Coronavirus3D: 3D structural visualization of COVID-19 genomic divergence. *Bioinformatics Oxford Academic*. 2020, roč. 36, s. 4360–4362. Dostupné z DOI: https://doi.org/10.1093/bioinformatics/btaa550.

15. PEARSON WR, Lipman DJ. "Improved tools for biological sequence comparison". *Proceedings of the National Academy of Sciences of the United States of America*. 1988, roč. 85, s. 2444–8. Dostupné z DOI: doi:10.1073/pnas.85.8.2444.

16. C. A. STEWARD A. P. Parker, B. A. Minassian. Genome annotation for clinical genomic diagnostics: strengths and weaknesses. *Genome Med*. 2017, roč. 9, s. 49. Dostupné z DOI: 10.1186/s13073-017-0441-1.

17. G. PETREA, M. Pertea. GFF Utilities: GffRead and GffCompare [version 2; peer review: 3 approved]. *F1000Research*. 2020, roč. 9, č. 304. Dostupné z DOI: 10.12688/f1000research.23297.2.

18. KAMRAN KARIMI, Daniel M Wuitchik. "Improved tools for biological sequence comparison". *Evol Bioinform Online*. 2018, roč. 14, s. 43. Dostupné z DOI: 10.1177/1176934318788866.

19. WUITSCHICK, JEFFREY D.; KARRER, KATHLEEN M. Analysis of Genomic G + C Content, Codon Usage, Initiator Codon Context and Translation Termination Sites In Tetrahymena Thermophila. *Journal of Eukaryotic Microbiology*. 1999, roč. 46, č. 3, s. 239–247. Dostupné z DOI: 10.1111/j.1550-7408.1999.tb05120.x.

20. LI Y., Yang X. GC usage of SARS-CoV-2 genes might adapt to the environment of human lung expressed genes. *Mol Genet Genomics*. 2020, roč. 295, s. 1537–1546. Dostupné z DOI: https://doi.org/10.1007/s00438-020-01719-0.

21. LEE, Benjamin D. Squiggle: a user-friendly two-dimensional DNA sequence visualization tool. *Bioinformatics*. 2018. Dostupné z DOI: 10.1093/bioinformatics/bty807.

22. BO LIAO, Xuyu Xiang; ZHU, Wen. Coronavirus phylogeny based on 2D graphical representation of DNA sequence. *Journal of computational chemistry*. 2006, roč. 27, s. 1196–1202. Dostupné z DOI: doi:10.1002/jcc.20439.

23. BIRNEY, Ewan; THOMPSON, Julie D.; GIBSON, Toby J. PairWise and SearchWise: Finding the Optimal Alignment in a Simultaneous Comparison of a Protein Profile against All DNA Translation Frames. *Nucleic Acids Research*. 1996, roč. 24, č. 14, s. 2730–2739. ISSN 0305-1048. Dostupné z DOI: 10.1093/nar/24.14.2730.

24. SEAMAN, Josiah; BUGGS, Richard J. A. FluentDNA: Nucleotide Visualization of Whole Genomes, Annotations, and Alignments. *Frontiers in Genetics*. 2020, roč. 11, s. 292. ISSN 1664-8021. Dostupné z DOI: 10.3389/fgene.2020.00292.

25. WOUTER PENARD, Tim van Werkhoven. On the Secure Hash Algorithm family. 2016, roč. 3, s. 41. Dostupné tiež z: https://web.archive.org/web/20160330153520/http://www.staff.science.uu.nl/~werkh108/docs/study/Y5_07_08/infocry/project/Cryp08.pdf.

26. SHAPIRO E Biezuner T, Linnarsson S. Single-cell sequencing-based technologies will revolutionize whole-organism science. *nature reviews genetics*. 2013, roč. 14, s. 201–213. Dostupné z DOI: doi:10.1038/nrg3542.

27. SIEBER, Patricia; PLATZER, Matthias; SCHUSTER, Stefan. The Definition of Open Reading Frame Revisited. *Trends in Genetics*. 2018, roč. 34, č. 3, s. 167–170. ISSN 0168-9525.