BRNO UNIVERSITY OF TECHNOLOGY

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

FAKULTA ELEKTROTECHNIKY
A KOMUNIKAČNÍCH TECHNOLOGIÍ

DEPARTMENT OF BIOMEDICAL ENGINEERING

ÚSTAV BIOMEDICÍNSKÉHO INŽENÝRSTVÍ

FUNCTIONAL ANNOTATION OF NON-MODEL BACTERIA BASED ON SEQUENTIAL HOMOLOGY

FUNKČNÍ ANOTACE NEMODELOVÝCH BAKTERIÍ S VYUŽITÍM SEKVENČNÍ HOMOLOGIE

BACHELOR'S THESIS
BAKALÁŘSKÁ PRÁCE

AUTHOR
AUTOR PRÁCE

Petra Polakovičová

SUPERVISOR
VEDOUCÍ PRÁCE

Mgr. Ing. Karel Sedlář, Ph.D.

BRNO 2022

BRNO UNIVERSITY OF TECHNOLOGY
FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

Bachelor's Thesis

Bachelor's study program **Biomedical Technology and Bioinformatics**

Department of Biomedical Engineering

Student: Petra Polakovičová ***ID:*** 220943

Year of study: 3 ***Academic year:*** 2021/22

TITLE OF THESIS:

Functional annotation of non-model bacteria based on sequential homology

INSTRUCTION:

1) Study the principle of computational annotation of bacterial genomes, especially pipelines for functional annotation of genes based on sequence similarity to model organisms and their classification to cluster of orthologous genes (COG). 2) Prepare a literature review of tools and databases for COG annotation. 3) Examine various tools using a suitable genome or genomes and compare the results of annotation. 4) Propose and implement your own method for improved COG annotation, for example by using multiple resources of annotated orthologous sequences. 5) Supplement the tool for functions allowing suitable visualization of results by exporting data to additional tools, e.g., DNAPlotter. 6) Compare your method to the existing ones, evaluate computational time and accuracy of various approaches and discuss the results.

RECOMMENDED LITERATURE:

[1] GALPERIN, Michael Y., Kira S. MAKAROVA, Yuri I. WOLF a Eugene V. KOONIN. Expanded microbial genome coverage and improved protein family annotation in the COG database. Nucleic Acids Research. 2015, 43(D1), D261-D269.

[2] HUERTA-CEPAS, Jaime, Damian SZKLARCZYK, Davide HELLER, et al. EggNOG 5.0: a hierarchical, functionally and phylogenetically annotated orthology resource based on 5090 organisms and 2502 viruses. Nucleic Acids Research. 2019, 47(D1), D309-D314.

Date of project specification: 7.2.2022 ***Deadline for submission:*** 27.5.2022

Supervisor: Mgr. Ing. Karel Sedlář, Ph.D.

doc. Ing. Jana Kolářová, Ph.D.
Chair of study program board

ABSTRACT

The genome sequence is an essential informational resource in the field of biology and it is, therefore, a constant subject of scientific research. Bioinformatics offers computational methods for genome's automatic analysis and processing. The bachelor thesis is dedicated to the bacterial genome, its organization, and fundamental characteristics, and subsequently description of its annotation. It mainly focuses on functional annotation (description of the biological function of predicted genes) using assignment to clusters of orthologous genes (COG) based on sequential homology. It describes the most commonly used tools and databases that use this type of annotation and then compares some of them by annotating seven bacterial genomes. Its main task is to propose a new method that improves COG annotation and makes it easy to visualize.

KEYWORDS

bacteria, bacterial genome annotation, functional annotation, COG, orthologous genes

ABSTRAKT

Genomová sekvence je důležitým zdrojem informací v oblasti biologie, a proto je neustálým předmětem zájmu vědeckého výzkumu. Na práci s genomem a jeho analýzu nabízí bioinformatika různé výpočetní metody. Tato práce se věnuje bakteriálnímu genomu, jeho organizaci, základním vlastnostem a následně popisuje jeho anotaci. Zaměřuje se hlavně na funkční anotaci (popis biologické funkce predikovaných genů) na základě přiřazení takzvaných klastrů ortologních genů (COG) s využitím sekvenční homologie. Popisuje nejpoužívanější nástroje a databáze, které se využívají pro COG anotaci a poté několik z nich porovnává při anotaci sedmi bakteriálních genomů. Jejím hlavním cílem je navrhnout metodu, která vylepší COG anotaci a zjednoduší její výslednou vizualizaci.

KLÍČOVÁ SLOVA

bakterie, anotace bakteriálního genomu, funkční anotace, COG, ortologní geny

ROZŠÍŘENÝ ABSTRAKT

Genomová sekvence je důležitým zdrojem informací v oblasti biologie, a proto je neustálým předmětem zájmu vědeckého výzkumu. Obsahuje genetickou informaci organismu uloženou v DNA a zahrnuje všechny geny i nekódující sekvence. Nejprve je však potřeba tyto informace z genomu získat. Bioinformatika poskytuje výpočetní metody na práci s genomem a jeho analýzu a díky nim je možné genom anotovat a zjistit tak jeho vlastnosti. První část bakalářské práce popisuje bakterii jako prokaryotický organismus, popisuje organizaci jejího genomu a také její základní vlastnosti. Prokaryota nemají žádné vnitřní membránové struktury, neobsahují jádro a kruhová DNA je uložena v chromozomu, nazývaném nukleoid. Tyto vlastnosti je bezprostředně oddělují od organismů eukaryotických. Kromě hlavního chromozomu mohou bakterie obsahovat i plazmidy, které je možné definovat jako malé části DNA, které jim zajišťují některé důležité vlastnosti, jako například rezistenci k antibiotikům, zvýšenou patogenitu nebo možnost metabolizovat i jiné zdroje potravy. Dalšími hlavními vlastnostmi bakteriálního genomu je přítomnost operonů nebo regionů CRISPR. Následně je popsána anotace bakteriálního genomu. Ta může být rozdělena na strukturní a funkční. V sekci zabývající se strukturní anotaci jsou vysvětleny základní pojmy, se kterými se při anotaci můžeme setkat – CDS, gen, pseudogen, ORF či nekódující RNA sekvence. Funkční anotace genomu zahrnuje přiřazení biologické funkce na základě sekvenční homologie s proteiny známé funkce. Typické anotační nástroje Prokka, PGAP nebo RAST po predikci genů využívají na přiřazení funkce nástroje jako BLASTP, DIAMOND a HMMER, které prohledávají různé proteinové databáze. Tento přístup má však několik závažných nedostatků, protože čistě jen sekvenční podobnost nezaručuje správně určenou funkci. Tyto problémy řeší metoda anotace, která využívá takzvané skupiny ortologních genů. Ortologní geny jsou takové, které se vyvinuly ze společného předka a zachovali si tak stejnou či podobnou funkci. Nástroje využívající tento přístup přiřazují funkci na základě prohledávání databází ortologních genů, například COG, KEGG, eggNOG, OrthoDB, nebo MBGD. Tato práce popisuje dvě z těchto databáz, a to konkrétně COG a eggNOG a také nástroje, které je využívají. COG je databáze, která vznikla za účelem evoluční klasifikace proteinových rodin, dnes je využívána na funkční anotaci či fylogenetickou analýzu. Databáze obsahuje manuálně vybrané skupiny ortologních genů a jejich funkční popis. Rozšíření této databáze tvoří databázi eggNOG, která obsahuje skupiny z COG, tak jako i další, které jsou vytvořené plně automaticky (OG). Nástroje které nabízí kompletní bakteriální anotaci s využitím tohoto přístupu, jsou například eggNOG-mapper nebo Operon-mapper. EggNOG-mapper plně využívá databázi eggNOG na přiřazení COG nebo OG skupiny pomocí nástroje DIAMOND, MMseqs2 nebo HHMER3. Nástroj Operon-mapper prohledává databáze COG a ROG (Remained Orthologous Groups) také pomocí HMMER3.

Praktická část této práce spočívá v porovnání třech anotačních nástrojů. Jde o nástroje eggNOG-mapper, Operon-mapper a Batch CD-Search, které jsou popsány z uživatelského hlediska. Těmito nástroji bylo anotováno několik bakteriálních genomů, včetně modelové bakterie *Escherichia coli*. Jako vstup sloužily proteinové sekvence dostupné v Genbank databázi. Protože každý nástroj poskytuje své výsledky v různých formátech a typech souborů, pro ulehčení následné analýzy byly jejich jednotlivé vstupy zpracovány do jednotného GFF formátu. Z výsledků pro jednotlivé genomy bylo možné porovnat, kolik procent z predikovaných CDS dokázaly nástroje přiřadit k jednotlivým ortologním skupinám a jejich kategoriím a v jakém rozsahu se přiřazení shoduje pro každou dvojici nástrojů, zvlášť pro *Escherichia coli* a nemodelové bakterie. Nejvíc procent CDS dokázal přiřadit nástroj eggNOG-mapper. Další v pořadí skončil Operon-mapper a následně Batch CD-Search, a to hlavně kvůli tomu, že prohledává proteiny jen v jedné databázi (COG). V porovnání jednotlivých dvojic bylo zjištěno, že Operon-mapper a eggNOG-mapper se shodují v menší míře, než je to u jiných dvojic. Jde o důkaz toho, že jejich databáze obsahují jiné ortologní skupiny. Dále bylo zjištěno, že se do velké míry neshoduje zařazení ortologních skupin do jejich kategorií mezi těmito nástroji. Další analýza spočívala v tom, že byly extrahovány všechny CDS, které dokázaly anotovat všechny tři nástroje. Bylo zjištěno, že při *E. coli* se v téměř 99% alespoň dva nástroje shodovaly. Pro nemodelové bakterie šlo o 97% shodu. Tato skutečnost byla dále využita pro zlepšení funkční anotace za pomoci konsenzuálního spojení těchto tří nástrojů. Za tímto účelem byl vytvořen nástroj s názvem COG-or, který byl implementován v jazyce Python. COG-or dokáže zlepšit funkční anotaci bakteriálních genomů, díky zmíněnému konsenzuálnímu spojení nástrojů eggNOG-mapper, Operon-mapper a Batch CD-Search. Dále zjednodušuje vizualizaci výsledné anotace tím, že je jeho finální výsledek kompatibilní s externím nástrojem DNAPlotter.

POLAKOVIČOVÁ, Petra. *Functional annotation of non-model bacteria based on sequential homology*. Brno: Brno University of Technology, Faculty of Electrical Engineering and Communication, Department of Biomedical Engineering, 2022, 65 p. Bachelor's Thesis. Advised by Mgr. Ing. Karel Sedlář, Ph.D.

Author's Declaration

Author:	Petra Polakovičová
Author's ID:	220943
Paper type:	Bachelor's Thesis
Academic year:	2021/22
Topic:	Functional annotation of non-model bacteria based on sequential homology

I declare that I have written this paper independently, under the guidance of the advisor and using exclusively the technical references and other sources of information cited in the paper and listed in the comprehensive bibliography at the end of the paper.

As the author, I furthermore declare that, with respect to the creation of this paper, I have not infringed any copyright or violated anyone's personal and/or ownership rights. In this context, I am fully aware of the consequences of breaking Regulation § 11 of the Copyright Act No. 121/2000 Coll. of the Czech Republic, as amended, and of any breach of rights related to intellectual property or introduced within amendments to relevant Acts such as the Intellectual Property Act or the Criminal Code, Act No. 40/2009 Coll. of the Czech Republic, Section 2, Head VI, Part 4.

Brno ....

author's signature*

*The author signs only in the printed version.

ACKNOWLEDGEMENT

I would like to express my gratitude to my advisor Mgr. Ing. Karel Sedlář, Ph.D. for his guidance and feedback throughout this thesis. I also want to express gratitude towards my parents for their support throughout the study and also my partner for his support and help with proofreading the text.

Contents

Introduction **13**

1 Bacterial Genome **14**
1.1 Classification of Bacteria as Prokaryotes 14
1.2 Genome Organization 15
1.2.1 Nucleoid 15
1.2.2 Plasmids 15
1.2.3 Operons 15
1.2.4 CRISPR 16
1.3 The Size and Gene Content 17
1.4 GC Content 17
1.5 Genome Annotation 18
1.5.1 Structural Annotation 18
1.5.2 Functional Annotation 19

2 Annotation Pipelines **21**
2.1 Prokka 21
2.2 PGAP 22
2.2.1 Gene Prediction 22
2.2.2 Protein Naming 23
2.3 RAST 25

3 COG Annotation **26**
3.1 COG Database 26
3.2 eggNOG Database 28
3.3 eggNOG-mapper 30
3.4 Operon-mapper 30

4 Comparison of Annotation Tools **32**
4.1 eggNOG-mapper 32
4.2 Operon-mapper 33
4.3 Batch CD-Search 33
4.4 Results for OG Assignment 34
4.5 Results for Category Assignment 36

5 COG-or **38**
5.1 The Main Idea 38
5.2 Implementation 40

5.2.1 Program Processor 40
5.2.2 Consensus 40
5.2.3 Track Manager 41
5.3 Results and Discussion 42

Conclusion **48**

Bibliography **50**

Symbols and abbreviations **59**

List of appendices **60**

A Schematic diagram of Consensus module **61**

B Comparison of tools **62**

C Confusion matrices constructed to obtain precision and recall **63**

D List of electronic attachments **65**

List of Figures

1.1 Cells of eukaryotes (left) and prokaryotes (right) [2]. 14
1.2 The lactose operon of *Escherichia coli* [6]. 16
1.3 The distributions of genome sizes and gene density across approximately 3000 bacterial genomes [12]. 17
2.1 Individual steps of functional annotation by PGAP [17]. 24
3.1 The COG of the *AGPS* (Alkylglycerone Phosphate Synthase) gene family [66]. 29
4.1 An example of eggNOG-mapper annotation for one query sequence. . 33
4.2 Operon-mapper's outputs. 34
4.3 The sample data of Batch CD-Search results. 34
4.4 Comparison of OG assignment by individual tools for *Escherichia coli*. 35
4.5 Comparison of OG assignment by individual tools for non-model bacteria. 36
4.6 Comparison of category assignment by individual tools for *Escherichia coli*. 37
4.7 Comparison of category assignment by individual tools for non-model bacteria. 37
5.1 Schematic diagram of annotation process using the COG-or. 39
5.2 Comparison of the primary visualisation (left) and one after the track template is uploaded (right) for the genome of *Schlegella thermodepolymerans* DSM 15344. 41
5.3 Time complexity of the COG-or. 43
5.4 Tools comparison with COG-or in OG assignment. 44
5.5 The chromosomal map of *Clostridium beijerinckii* DSM 791 drawn using DNAPlotter and the COG-or. 45
5.6 Distribution of COG categories in annotated bacteria. 46
5.7 Principal component analysis (PCA) plot based on the relative abundance of COG categories in the annotated bacteria. 47
5.8 GBDP tree of the annotated bacteria. 47

List of Tables

1.1 The examples of the size of a few bacterial genomes. 17
2.1 Software tools used for structural annotation by Prokka. 21
2.2 Databases and tools used for functional annotation by Prokka. 21
2.3 Databases and tools used by PGAP. 23
3.1 The 26 functional categories of the COG database. 28
4.1 The genomes chosen for comparison of annotation tools. 32
5.1 A simplified table generated by the *get_track_template*. 42
5.2 The chosen genomes for computation of time complexity of the COG-or. 43
5.3 The chosen rules for confusion matrix construction. 44
5.4 Precision and recall values for the annotation tools. 45
C.1 Confusion matrix for group assignment by COR-or and Batch CD-Search. 63
C.2 Confusion matrix for group assignment by COR-or and eggNOG-mapper. 63
C.3 Confusion matrix for group assignment by COR-or and Operon-mapper. 63
C.4 Confusion matrix for category assignment by COR-or and Batch CD-Search. 63
C.5 Confusion matrix for category assignment by COR-or and eggNOG-mapper. 63
C.6 Confusion matrix for category assignment by COR-or and Operon-mapper. 64

Introduction

By the end of 2021, the complete genome sequences of over 360 000 bacteria were available in the public databases. As the pace of genome sequencing is continuously increasing nowadays, a need arises to develop new computational methods for genome annotation because it is not possible to do it manually anymore.

The common approach for annotating a new bacterial genome is to use one of the existing pipelines, which uses the similarity of the gene sequence to known genes in various databases.

This bachelor thesis focuses on the functional annotation of the bacterial genome using clusters of orthologous genes (COGs). Orthologous genes are genes that have retained the same or similar function due to having evolved from a common ancestor. This method allows assigning a biological function to a gene that belongs to a group of orthologous genes with known function.

This thesis aims to introduce the reader to the bacterial genome, its fundamental properties, and its characteristics. Furthermore, the annotation of the bacterial genome is described, focusing on functional annotation based on sequence homology and classification to orthologous groups. The thesis analyzes databases and tools that use this approach for gene function assignment. Several tools for COG annotation are compared by annotating model bacterium *Echerichia coli* and non-model bacteria published by the Department of Biomedical Engineering.

However, a major disadvantage of COG annotation is that the existing tools often differ in assigning orthologous groups and their categories. Their output files are confusing, and difficult to be further analyzed and visualized. The aim of the practical part is to propose and implement the method for improved COG annotation and visualization of the final output. For this task, the COG-or tool was developed, a python package that processes the outputs of three tools for COG annotation, improves the functional annotation using consensus, and visualizes the final annotated genome with DNAPlotter.

1 Bacterial Genome

The first chapter is dedicated to fundamental questions about bacteria, bacterial genome, and its annotation.

1.1 Classification of Bacteria as Prokaryotes

The two most basic classes of living organisms are the prokaryotes (bacteria and archaea) and the eukaryotes.

Eukaryotic cells are compartmentalized and the chromosomes are located in the nucleus [1]. Prokaryotic cells lack extensive internal compartments, but their physical structure is nonetheless highly ordered. The organized DNA structure in prokaryotic cells is referred to as the nucleoid (Figure 1.1).

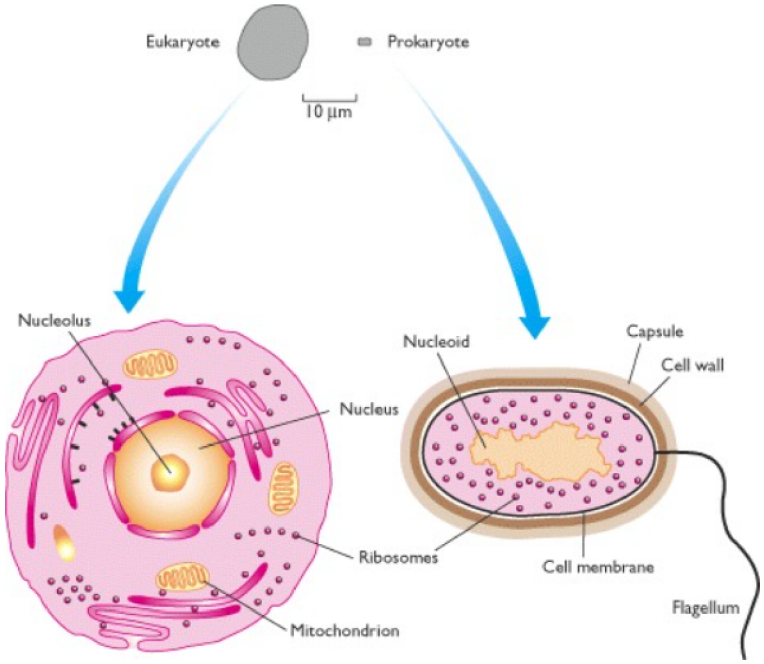


Fig. 1.1: Cells of eukaryotes (left) and prokaryotes (right) [2].

The upper part of the figure shows a typical human cell and bacterium drawn to scale. The lower depiction shows the internal structures of the eukaryotic and prokaryotic cells. Eukaryotic cells are characterized by their membrane-bound compartments, which are absent from prokaryotes [2].

1.2 Genome Organization

The genome of a bacterium consists of its entire collection of genes, and these can be located in a chromosome and on extra-chromosomal autonomous replicons such as plasmids [1].

1.2.1 Nucleoid

As mentioned above, a defining characteristic of prokaryotes so is that they do not possess a membrane-bound nucleus. Until recently, the traditional view has been that in a typical bacterium the genome is contained in a large, covalently closed circle of double-stranded DNA, localized within the nucleoid [3]. This is true for *Escherichia coli* and many of the other commonly studied bacteria. Nevertheless, there is a large number of bacterial species with linear chromosomes, including *Borrelia burgdorferi* and *Streptomyces coelicolor* [4].

The nucleoid is a lightly staining region of the otherwise featureless prokaryotic cell that contains genetic material but lacks the surrounding membrane. It is composed of the chromosome and associated molecules including RNA polymerase, DNA polymerase, DNA-binding proteins, and RNA molecules [3] [5].

1.2.2 Plasmids

A plasmid is a small piece of DNA, often but not always circular, that coexists with the main chromosome in a bacterial cell. There are different types of plasmids [3]. Some of them are able to integrate into the main genome, but others are thought to be permanently independent. Chromosomes carry genes for basic cell functions. In contrast, plasmids are not typically required for the survival of the cells [5], although they can provide essential resistance to stresses and the ability to survive in particular environments.

Genes carried by plasmids are useful by coding for properties such as antibiotic resistance or the ability to utilize complex compounds such as toluene as a carbon source. Furthermore, they are responsible for increasing the pathogenicity of the cell [2]. Many of them can transfer from one cell to another, and the same plasmids can be found in bacteria of different species.

1.2.3 Operons

One characteristic feature of bacterial genomes is the presence of operons [2]. An operon is a group of genes that are located adjacent to one another in the genome, with perhaps just one or two nucleotides between the end of one gene and the start of

the next. All the genes in an operon are expressed as a single unit. An example of this can be the lactose operon (Figure 1.2).

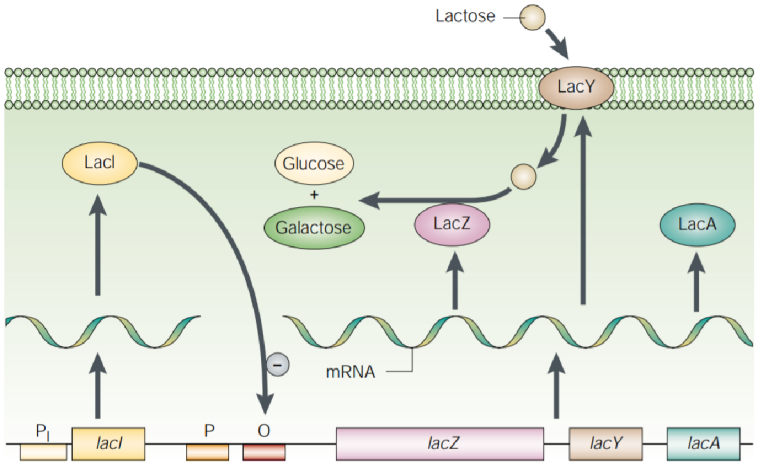


Fig. 1.2: The lactose operon of *Escherichia coli* [6].

The lactose operon contains three genes: *lacZ*, *lacY* and *lacA*, which are transcribed from a single promoter (P). The promoter produces a single mRNA from which these proteins are translated. This *lac* operon is regulated by LacI repressor, which is the product of the *lacI* gene. The repressor inhibits transcription by binding to the *lac* operator (O). Repressor binding to the operator is prevented by the unducer [6].

1.2.4 CRISPR

Clustered regularly interspaced short palindromic repeats (CRISPR) loci consist of an array of short (approximately 30 – 40 base pairs) and partially palindromic, repetitive sequences interspaced by intervening spacer sequences of a size similar to that of the repeated unit [7]. They are widespread among approximately 50% of sequenced bacterial genomes. These sequences are derived from DNA fragments of bacteriophages that had previously infected the prokaryote. Hence, they play a key role in the antiviral defense system and prove a form of acquired immunity [8].

CRISPR-associated protein 9 (Cas9) is an enzyme that uses CRISPR sequences as a guide to recognize and cleave specific strands of DNA that are complementary to the CRISPR sequence [8] [9]. Usage of this system is now popular genetic engineering technology that can be used to edit genes within organisms, for the development of which there was even the 2020 Nobel Prize awarded [10].

1.3 The Size and Gene Content

A bacterial genome varies in size from about 0.5 Mb to over 10 Mb, but most genomes consist of less than 5 Mb (Figure 1.3). It generally encodes 600 – 6000 proteins [11]. The examples of the size of a few bacterial genomes are shown in Table 1.1.

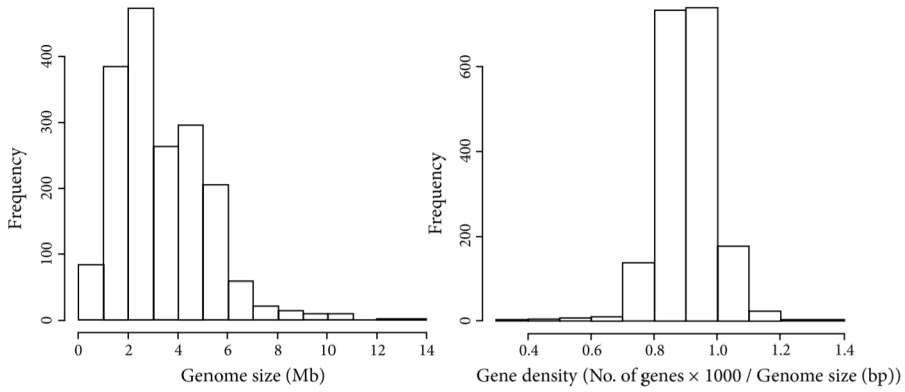


Fig. 1.3: The distributions of genome sizes and gene density across approximately 3000 bacterial genomes [12].

Tab. 1.1: The examples of the size of a few bacterial genomes.

Organism	AC Number	Size [Mb]	GC %	Protein Count
Mycobacterium tuberculosis H37Rv	NC_000962.3	4.41	65.6	3906
Escherichia coli K-12	NC_000913.3	4.64	50.8	4285
Chlamydia trachomatis D/UW-3/CX	NC_000117.1	1.04	41.3	887
Staphylococcus aureus NCTC 8325	NC_007795.1	2.82	32.9	2767
Salmonella enterica LT2	NC_003197.2	4.95	52.22	4548
Haemophilus influenzae NCTC 8143	NZ_LN831035	1.89	38.2	1754
Ureaplasma parvum ATCC 27815	NC_010503.1	0.75	25.5	598
Mycoplasma genitalium G-37	NC_000908.2	0.58	31.7	511

As for gene content, the chromosome sizes of prokaryotic organisms correlate with the number of genes. Protein coding regions typically occupy about 85% of a prokaryotic genome and the average gene density is approximately one gene per 1 kb DNA sequence (Figure 1.3) [3].

1.4 GC Content

One of the most highly variable features of bacterial genomes is overall base composition. Bacterial genomes are remarkably various in their GC content, which among

sequenced genomes range from 13% to 75% [4]. Interestingly, the GC content correlates with the genome and chromosome size. In all genera, protein-coding genes tend to have slightly higher GC content (about 5-10% on average) than intergenic sequences.

1.5 Genome Annotation

The genome sequence is a powerful informational resource that has no equal in the field of biology. But in the first place, the information has to be extracted by annotation.

The first cellular organism to have its entire genome sequenced was *Haemophilus influenzae* in 1995 [13]. By 2021, the complete sequences of the genomes of 362 175 bacteria were available in the public databases. Initially, trained annotators curated and completed the highest-quality genome annotations manually. The constantly accelerating pace of genome sequencing has evoked the need for computational annotation.

In bioinformatics, annotation means obtaining useful information from raw sequenced data [14]. It is a complex process, requiring the combination of plenty of tools. That is typically done using sequence annotation pipelines (i.e. a variety of software modules).

Annotation of prokaryotic sequences can be divided into structural and functional. Structural annotation includes a prediction of protein-coding genes, as well as other functional genome units such as structural RNAs, tRNAs, small RNAs, pseudogenes, control regions, direct and inverted repeats, insertion sequences, transposons, and other mobile elements. Functional annotation defines the role of the aforementioned genetic structures encoded in the DNA sequence [15].

1.5.1 Structural Annotation

This section will describe essential terms which can be encountered in a structural annotation.

Typically, the first step of annotation is the detection of coding sequences (CDS). CDS, a region of DNA or RNA derived from gene prediction, determines the sequence of amino acids in a protein [16].

Gene is a region of DNA that is transcribed as a single unit and carries information for a discrete hereditary characteristic, usually corresponding to a single protein or a single RNA [1].

In prokaryotes, the gene density is high (see Section 1.3), and the vast majority of genes have no introns. For this reason, finding genes in prokaryotic genomes is not as difficult as in eukaryotes.

There are many hybridization-based techniques and other experimental methods that can be used to detect genes in genomic DNA. However, due to the rapid accumulation of genomic sequence data, each annotation cannot practically be tested in a laboratory. That is why bioinformatics provides computational methods to identify potential genes. Three types of features can be recognized for a gene identification: signals, contents, and homologies [14]. Briefly, signals are discrete, local sequence motifs tend to have consensus sequences that can be searched by signal sensors algorithm. Contents do not have consensus sequences, but they do have conserved features that distinguish them from surrounding DNA. Homologies are matches to known genes.

Pseudogenes are nucleotide sequences of DNA that have accumulated multiple mutations that have rendered ancestral genes inactive and unfunctional [1]. They are usually identified because they contain frameshifts or internal stops when translated. Partial genes that occur in the middle of a sequence are also flagged as pseudo [17].

Protein-encoding genes have an open reading frame (ORF). ORF is a continuous nucleotide sequence free from stop codons in at least one of three reading frames (and thus with the potential to code for protein) [1]. In bacterial genomes, ORFs are easy to detect generally because they are uninterrupted by introns. This is done by carrying out a six-frame translation and identifying the longest ORF in the six possible protein sequences.

Some of the genetic information is not translated into protein, but nonetheless, it contains some form of information [18]. Therefore, it is very important to include RNA genes in the annotation. Ribonucleic acids (RNAs) are split into two distinct classes: messenger RNAs (mRNAs), which are translated into proteins, and non-protein-coding RNAs (ncRNAs), which function at the RNA level only. The term RNA gene is used for the DNA sequence from which an ncRNA is transcribed [19]. The functions of the ncRNAs are important and diverse. They can be divided into several classes which are involved in many biological processes, such as gene regulation, information transfer, RNA processing, and protein synthesis.

1.5.2 Functional Annotation

Functional annotation of the identified genes involves annotating or assigning a prediction of biological function based on similarity to known or other predicted functions or functional domains in databases [15].

In general, the annotation usually begins with classifying them into more manageable groups (protein families). This is commonly done at the level of protein sequence by homology to previously sequenced genes, information about which is accessible in public sequence databases, annotation is then assigned accordingly due to the relation to the known protein [12].

A typical protein annotation pipeline searches for similarities using the BLASTP [20] or, more recently, HMMER [21] against several different databases of protein sequences [22], such as Uniprot [16], Pfam [23], InterPro [24] and CDD [25]. This approach has several disadvantages that should not be overlooked [26]. The two proteins may have different functions if the sequence similarity is low. Furthermore, this method often classifies proteins as 'uncharacterized' or 'putative' even when the function of a close homolog is already known. And last but not least, differences in domain architectures of homologs may lead to an inaccurate functional assignment.

Other levels of classification are defined using clustering, which leads to groups of proteins [12]. The relationships between genes from different genomes are naturally represented as a system of homologous families that include both orthologs and paralogs. Orthologs are genes in different species that evolved from a common ancestral gene by speciation. They typically perform equivalent functions in the course of evolution. By contrast, paralogs are genes related by duplication within a genome and perform biologically distinct functions, even if related to the original one [27]. Thus, the identification of orthologs is critical for the reliable prediction of gene functions in newly sequenced genomes [28].

The COG (The Clusters of Orthologous Groups of proteins) database [29] shares some features with the aforementioned approach (it is built upon BLASTP comparison) but differs from them in several important aspects [26]. This thesis focuses mainly on this approach of functional annotation. The COG database, along with several other systems for orthology analysis, is described in Chapter 3.

2 Annotation Pipelines

The most commonly used pipelines are Prokka [30], NCBI Prokaryotic Genome Annotation Pipeline (PGAP) [31] and RAST [32], which will be described in this chapter.

2.1 Prokka

Prokka [30] is a software tool that is designed for bacterial genome annotation. It coordinates a suite of separate software tools and offers a fast and reliable genome annotation. These tools are listed in Tables 2.1 and 2.2. A typical bacterial genome can be annotated in about $\sim$ 10 minutes on a quad-core desktop computer.

Tab. 2.1: Software tools used for structural annotation by Prokka.

Tool	**Features predicted**
Prodigal [33]	Coding sequence (CDS)
RNAammer [34]	Ribosomal RNA genes (rRNAs)
Aragorn [35]	Transfer RNA genes (tRNAs)
SignalP [36]	Signal leader peptides
Infernal [37]	Non-coding RNA (ncRNAs)

Tab. 2.2: Databases and tools used for functional annotation by Prokka.

Database/tool	**Description**
Uniprot [16]	A database of protein sequences with functional information
RefSeq [38]	A database of prokaryotic genomes and sequence annotation
Pfam [23]	A database of protein families
TIGRFAM [39]	A database of protein family definitions
HMMER [21]	A tool for searching databases for homologs using hidden Markov models
Blast+ [20]	A tool for searching databases for similar sequences.

To use the Prokka pipeline, a DNA sequence in FASTA format is needed. The ideal input is a finished sequence without gaps, but a set of scaffold sequences produced by de novo assembly is acceptable.

Prokka annotates the genome in two stages. First, Prodigal [33] detects the genes (structural annotation). Second, the putative gene product is described by comparing the coding sequences to a large database of known sequences at a protein

sequence level. This is done in a hierarchical manner. It means starting with smaller trustworthy databases, moving to more extensive ones, and finally to curated models of protein families. The first step of functional annotation is an optional user-provided a set of annotated proteins which are expected to be trustworthy curated datasets. In the next point, Uniprot [16] database and RefSeq [38] are applied for comparing. In these cases, BLAST+ [20] is used for the search.

Subsequently, a series of hidden Markov model profile databases are applied, namely, Pfam [23] and TIGRFAMs [39]. This is performed using hmmscan from the HMMER3 [21].

Several files are produced as output, containing input sequence, translated sequence, all genomic features in FASTA and Genbank format, summary statistics, and other additional information like COG, EC number, and gene products.

Prokka is a freely available standalone software and can be used with the Anaconda platform.

2.2 PGAP

The NCBI Prokaryotic Genome Annotation Pipeline (PGAP) [40] is designed to annotate bacterial and archaeal genomes. It is developed by using an approach that combines alignment-based methods with methods of predicting protein-coding genes, RNA genes, and other functional elements directly from sequence. The input for the pipeline can be a complete genome or draft genome comprising multiple contigs. As output, PGAP produces reports in a wide variety of formats, including annotated genome objects, annotation in Genbank flat file (GFF), and statistics from the annotation process. The individual parts of the annotation are described in more detail below, and the usage of the tools and databases is summarized in Table 2.3.

PGAP is available as a standalone software package and can be run on Linux.

2.2.1 Gene Prediction

ORFs are predicted by ORFfinder in all six frames of the genome and searched against the libraries of the hidden Markov model, namely TIGRFAM [39], Pfam [23] and NCBI Protein Clusters Database (PRKs) [41] and also against BlastRules using BLAST and ProSplign (Protein to nucleotide alignment tool). Afterwards, the final set of predicted proteins is made based on the resulting aligning evidence and the *ab initio* gene-finding program GeneMark-S2+ [42].

Tab. 2.3: Databases and tools used by PGAP.

Database/tool	Description
ORF-finder	A tool for searching ORFs
GeneMark-S2+ [42]	A gene prediction tool
Infernal [37]	A tool for searching databases for RNA structure
tRNAscan-SE [44]	A tool for predicting tRNA genes
PILER-CR [45]	A tool for identification of CRISPR repeats
CRT [46]	CRISPR recognition tool
TIGRFAM [39]	A database of protein family definitions
Pfam [23]	A database of protein families
PRKs [41]	NCBI protein clusters database
RFAM [43]	The RNA families database
CDD [25]	A database of protein families and protein domain models
ProSlign	A global protein to genome alignment tool
Blast [20]	A tool for searching databases for similar sequences.
HMMER [21]	A tool for searching databases for homologs using

Structural RNAs and non-coding RNAs are annotated by searching RFAM [43] models with Infernal's [37] cmsearch. As for tranfer RNA genes, tRNAscan-SE [44] is used. All of the tools PGAP uses are listed in Table 2.3 [17].

2.2.2 Protein Naming

Three major types of evidence are used by PGAP to assign gene functions. They are Hidden Markov Models, BlastRules, and domain architectures [17]. Proteins that do not hit any of this evidence are named based on homology to protein cluster representatives. A flowchart depicting the sequence of steps required to name the proteins is shown in Figure 2.1 [17].

Hidden Markov Models

An HMM-based protein family is a probabilistic model used to determine which proteins belong to the protein family. To construct such models, multiple sequence alignments of proteins are converted into a scoring system to generate an HMM profile. Amino acids at each position on the seed alignment are given a score according to their frequency.

The proteins predicted on novel genomes are searched against a hidden Markov model library in order to assign a gene function (name of the protein) [17]. PGAP uses HMMs from several sources such as NCBI PRKs [41], TIGRFAMs [39], a subset

of Pfam [23] HMMs. The predicted proteins are matched to HMMs using the hmmsearch in HMMER3 [21] software. A protein is considered a hit and assigned to the name if its sequence and domain scores are above the cutoffs defined for the HMMs.

BlastRules

BlastRules is a type of method based on BLAST. A BlastRule consists of model proteins with known biological function, BLAST identity, and coverage cutoffs. If any protein aligned to a model is above the cutoffs it is considered a BlastRule hit [17].

BlastRules are generally created for proteins that may be indispensable in antibiotic resistance, evolution, and pathogenicity.

Domain architectures

Proteins can be grouped into evolutionary conserved families based on their domain architecture because they are commonly associated with specific functions [17].

The Conserved Domain Database (CDD) [25] contains a comprehensive collection of common domain architectures, obtained from the pre-computed annotation of proteins with domain footprints.

SPARCLE (Subfamily Protein Architecture Labeling Engine) [25] is used for naming proteins that have been grouped by their domain architecture.

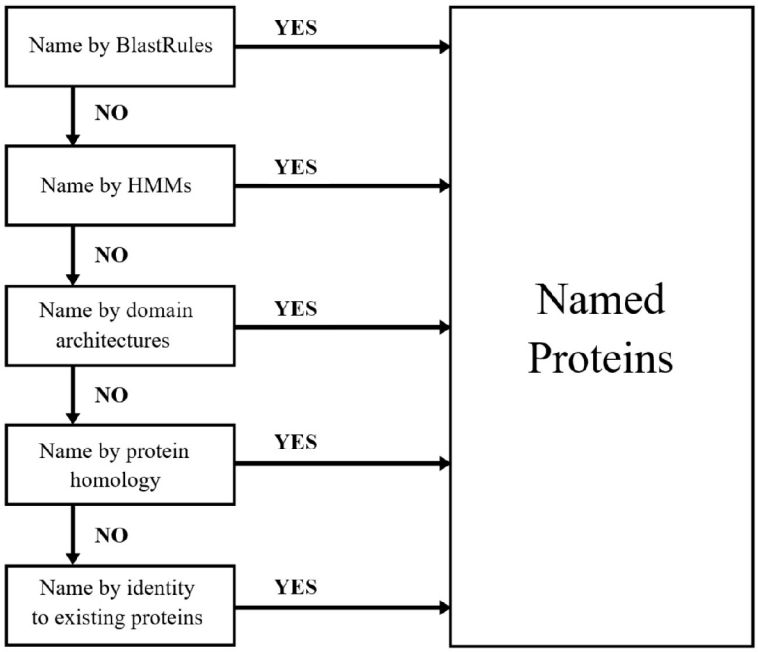


Fig. 2.1: Individual steps of functional annotation by PGAP [17].

2.3 RAST

RAST [32] (Rapid Annotations using Subsystems Technology) is an automatic annotation server for prokaryotic, phage, or plasmid genomes which is interconnected with the SEED [47].

The SEED [47] is a database that integrates different types of genomic data from a vast variety of sources. These include public genomes annotated by RAST, expert user annotations, metabolic modeling data, expression data, literature references verifying annotations and data from other popular resources, namely Swiss-Prot [48], GenBank [49], IMG [50], KEGG [51], CDD [25] and so forth. It also includes the subsystems, which are basically collections of functionally related families. These subsystems derive protein families called FIGfams [52]. They are sets of proteins that are end-to-end homologous and share a common function. FIGfams represent the core of the RAST annotation process [53].

The input of this pipeline is a DNA sequence (genome, phage, or plasmid), which is required to be in FASTA or GenBank format. A user can choose the annotation scheme RASTtk [20] (the current modular customizable production RAST pipeline) or RAST for the old pipeline. They differ in a few steps of the annotation process.

RAST involves the selenoproteins and pyrrolysoproteins identification, gene prediction with GLIMMER3 [54], tRNA and rRNA identification using tRNAscan-SE [44] for tRNAs [44] and BLASTN [20] against a set of RNA databases for rRNAs. The gene candidates from GLIMMER are searched for similarities with proteins in FIGfams. For this, the k-mers signature is used (sets of eight sequential amino acids). Iterative retrain GLIMMER3 on these validated genes follows. These two steps are repeated until no new gene candidates are found that are similar to those in subsystems. Functions of genes products are assigned by using BLASTP [20].

RASTtk [55] is a modular version of RAST that enables users to build a custom annotation pipeline. The main differences and improvements from classic RAST are: calling large repeat regions, finding CRISPR elements, and addition of a new version of the k-mer-based annotation. Instead of one gene prediction tool, RASTtk offers the option to choose GLIMMER3 [54], GeneMarkS [42] or Prodigal [33]. If no function can be found for a protein-encoding gene during the k-mer analysis, the BLAT [56] and BLASTP [20] are used for the final search.

As output, the user has several files to choose including Genbank, amino-acid, or nucleic-acid FASTA file, spreadsheet with all information from the annotation job (location of gene, strand, function, sequence).

RAST is available as a standalone tool or as an online service[1].

[1] Available at `https://rast.nmpdr.org/`

3 COG Annotation

At larger phylogenetic distances, the common approach of functional annotation, using the highest sequence similarity for assigning gene functions, is starting to fail. Annotation of newly sequenced genomes using clustering which leads to protein families is a more sophisticated method than just searching for the most similar sequence using BLAST [20] or HMMER [21]. If gene duplications would occur in each of the given two clades after their divergence, only a many-to-many relationship will adequately describe orthologs [28].

Identifying orthologs, which are more prone to retain their ancestral function than paralogs, constitutes a fundamental task in accurate functional predictions [57]. It allows us to assign functional information from one member to an entire cluster of orthologous groups (COG). A protein cluster is a group containing at least three orthologous proteins [12]. Each COG is assumed to have evolved from an individual ancestral gene through a series of speciation and duplication events. The COGs are being formed in the following process.

All pairwise sequence comparisons among protein sequences from several genomes are performed and for each protein, the best hit in each of the other genomes is detected. Numerous methods have been developed to derive orthologs and orthologous groups, the best hit triangle identification is the most common one [58]. This method allows the detection of orthologs among both slowly and quickly evolving genes because the consistency between best hits resulting in triangles does not depend on the absolute level of similarity between compared proteins.

Therefore, several databases and tools have been developed over the years that provide precomputed orthology predictions using different approaches and operational definitions. These are databases like COG [29], KEGG Orthology [51], OrthoMCL-DB [59], MBGD [60], eggNOG [57] or OrthoDB [61].

This chapter is dedicated to the two of these databases, namely COG, and eggNOG, and tools that use precomputed orthologous groups, eggNOG-mapper, and Operon-mapper.

3.1 COG Database

The COG database has been a popular tool for microbial genome annotation and comparative genomics for the past 24 years. Initially, it was created for purpose of evolutionary classification of protein families [26]. Currently, it is used for functional annotation of prokaryotic genome sequences, unification of annotation in groups of related organisms, analysis of genomic neighborhoods, analysis of metabolic pathways and prediction of alternative forms of enzymes, and so forth.

The database was initially created in 1997 [28]. The current update [62] includes 4877 COGs derived from complete genomes of 1187 bacteria and 122 archaea. In the current COG collection, all COGs are equal and there is no hierarchical structure unlike the database described below [26]. It also includes features like PDB (Protein Data Bank) links, COGs for proteins involved in CRISPR-Cas immunity, and a list of COGs grouped by pathways and functional systems.

The goal of the COG system is to represent a family of orthologous protein-coding genes. These COGs are merged into 26 functional categories which are listed in Table 3.1. It shows a list of functional categories as well as the COG example for each category and its annotation. Some COGs belong to more than one category. For instance, the COG2124 stands for Cytochrome P450 and it belongs to categories V and Q because this group is important for the biosynthesis of defensive compounds [63].

The success of this database is based on several key factors [64]. Firstly, it relies on the analysis of complete microbial genomes (proteomes). This allows reliable assignment of orthologs and paralogs for the most genes using a simple approach based on the search of triangles of bidirectional best hits. Due to this fact, recognition of distant homologs and separation of closely related paralogs is allowed.

Bidirectional best hit, also known as reciprocal best hit or reciprocal Blast hit, entails recognizing the pairs of genes in two different genomes that are more similar to each other than either is to any other gene in the other genome. It is a simple and widely used method to deduce orthology [65]. The procedure used to derive COGs included finding all triangles formed by best hits between the five major clades and merging those triangles that had a common side until no new ones can be joined [28]. The example of COG is shown in Figure 3.1 [66]. Solid lines show symmetrical best hits and broken likens show asymmetrical best hits.

Another factor is its orthology-based approach. This means assigning function to the entire set of identified orthologs using the function of the characterized member of the protein family. Finally, all the COGs are manually curated, whereby annotation errors and overpredictions are avoided [64].

When annotating a new bacterial genome using COG, a couple of approaches can be used. For example, searching the sequence against the COG database using CD-Search [68]. It performs functional annotation by using RPSBLAST (a variant of the PSI-BLAST) against the COG database with a protein FASTA file as input. Prokka (described in Section 2.1) also outputs COG in .tsv file.

Tab. 3.1: The 26 functional categories of the COG database.

	Category	**COG**	**Annotation**
J	Translation, ribosomal structure and biogenesis	COG0486	Leucyl-tRNA synthetase
A	RNA processing and modification	COG5186	Poly(A) polymerase Pap1
K	Transcription	COG1318	Predicted transcriptional regulator
L	Replication, recombination and repair	COG0305	Replicative DNA helicase
B	Chromatin structure and dynamics	COG5531	DNA-binding SWIB/MDM2 domain
D	Cell cycle control, cell division, chromosome partitioning	COG3087	Cell division protein FtsN
Y	Nuclear structure	-	-
V	Defense mechanisms	COG1421	CRISPR-Cas system type III
T	Signal transduction mechanisms	COG5599	Protein tyrosine phosphatase
M	Cell wall/membrane/envelope biogenesis	COG3511	Phospholipase C
N	Cell motility	COG1291	Flagellar motor component MotA
Z	Cytoskeleton	COG5023	Tubulin
W	Extracellular structures	COG5295	Autotransporter adhesin
U	Intracellular trafficking, secretion and vesicular transport	COG0681	Signal peptidase I
O	Posttranslational modification, protein turnover, chaperones	COG4826	Serine protease inhibitor
X	Mobilome: prophages, transposons	COG3293	Transposase
C	Energy production and conversion	COG0372	Citrate synthase
G	Carbohydrate transport and metabolism	COG1929	Glycerate kinase
E	Amino acid transport and metabolism	COG0031	Cysteine synthase
F	Nucleotide transport and metabolism	COG1001	Adenine deaminase
H	Coenzyme transport and metabolism	COG1612	Heme A synthase
I	Lipid transport and metabolism	COG0183	Acetyl-CoA acetyltransferase
P	Inorganic ion transport and metabolism	COG3721	Putative heme iron utilization protein
Q	Secondary metabolites biosynthesis, transport and catabolism	COG2124	Cytochrome P450
R	General function prediction only	COG1259	Bifunctional DNase/RNase
S	Function unknown	COG1470	Uncharacterized membrane protein

3.2 eggNOG Database

A major extension of the COG database is implemented in the eggNOG database [57] with an increased number of genomes included and new clusters of orthologs [64].

eggNOG, which stands for the evolutionary genealogy of genes: Non-supervised Orthologous Groups, was created in 2007 [58]. It expands the original idea of

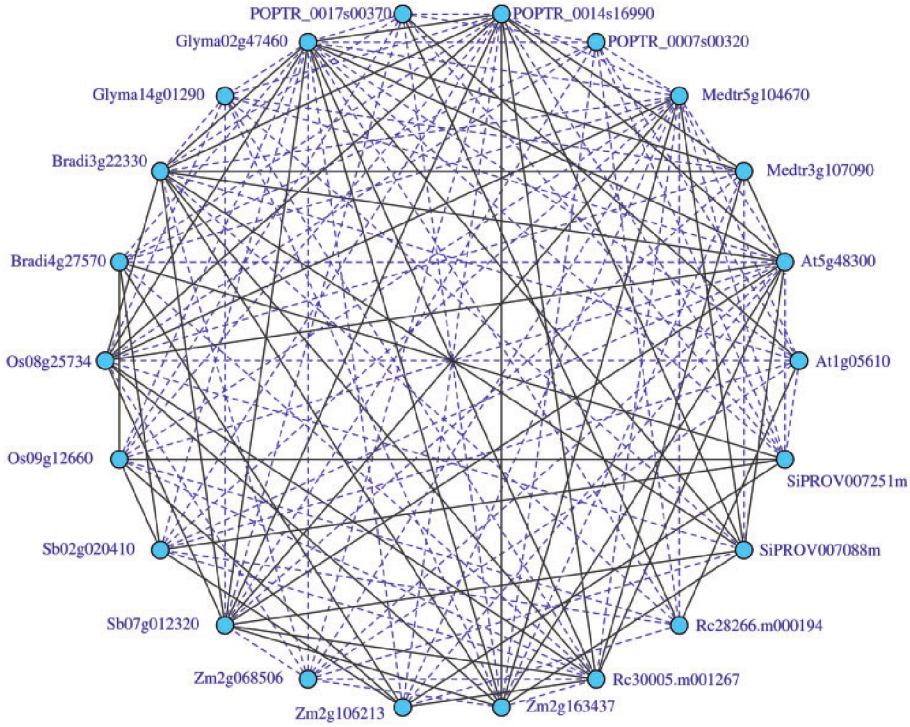


Fig. 3.1: The COG of the *AGPS* (Alkylglycerone Phosphate Synthase) gene family [66].

COGs to non-supervised orthologous groups constructed from numerous organisms. The current update contains up to 4.4 million orthologous groups derived from more than 5000 organisms (4445 bacteria) [57].

The eggNOG focuses on providing comprehensive functional annotations for the inferred orthologs, predictions across thousands of genomes covering the prokaryotes as well as eukaryotes, and hierarchical resolution of orthology assignments based on phylogenetic analysis.

An Orthologous Group (OG) is defined as a cluster of three or more orthologs. These groups were built using the best reciprocal hits derived from the all-against-all Smith-Waterman matrix.

First, Smith-Waterman similarities among all proteins in the database were computed. By joining triangles of best bidirectional hits, orthology between proteins was assigned. To enable unassigned proteins to join a group, the triangle criterion was relaxed by simple bidirectional best hits. All of the OGs were constructed in a hierarchical manner, simply by applying this procedure to several subsets of organisms [58]. In addition, the manually curated groups from the COG database were integrated into eggNOG.

Afterwards, functional annotation was performed. OGs were annotated using Gene Ontology [69], KEGG pathways [70], SMART/PFAM domains [71] and KEGG

modules [57].

The easiest way to annotate a new genome using the eggNOG database is to use the eggNOG-mapper [72], a tool for fast functional annotation of novel sequences using OGs and phylogenies from this database. This tool is briefly described in the next section.

3.3 eggNOG-mapper

The eggNOG-mapper [72] is a tool for functional annotation of large sets of sequences based on fast orthology assignments using precomputed clusters and phylogenies from the eggNOG database. The latest version of the tool, eggNOG-mapper v2 [73], is available standalone and as an online service.

The annotation algorithm consists of four elementary steps. The eggNOG-mapper starts with gene prediction using Prodigal [33]. Afterwards, sequence mapping is performed with three available modes (DIAMOND [67], MMseqs2 [74] and HMMER3 [21]).

The HMMER3 mode is significantly slower but offers much higher sensitivity. For each query, the best matching sequence, which points to a protein in eggNOG, is used to retrieve a list of fine-grained orthology assignments from a database of pre-analyzed eggNOG phylogenetic trees. Additional filters such as bit-score or e-value threshold can be used during this step to avoid inferring functional data for query sequences.

3.4 Operon-mapper

Operon-mapper [75] is a web server that predicts the operons of any prokaryotic genome sequence. The input for this tool is the genomic nucleotide sequence in FASTA format. The analysis itself is performed in several steps. First, ORF prediction using Prokka software [30]. Second, homology gene assignments are determined based on HMMs search using hmmsearch [21]. These models represent each of the COGs and Remained Orthologous Groups (ROGs). The next step is operon prediction. It is performed with an artificial neural network with the intergenic distance between the genes and a score of functional relationships of their protein products as inputs. The scores were defined in the STRING database [76] and they are presented for different pairs of proteins according to their related COG or ROG. The last step consists of gene function assignment based on the most significant hit using DIAMOND [67] against Uniprot [16].

As output, several files are produced based on user preference. Operon-mapper provides the predicted operonic gene pairs, a list of operons with their corresponding genes, the coordinates, and DNA or protein sequences of the predicted ORFs, the homology assignments of the proteins, corresponding to their COG or ROG, and the functional descriptions of the proteins.

4 Comparison of Annotation Tools

This chapter is dedicated to testing selected tools for COG annotation and comparing obtained results. Three tools were chosen, eggNOG-mapper, Operon-mapper, and Batch CD-Search which were described in Chapter 3.

The genome of *Escherichia coli* K-12 MG1655 as a model bacterium was selected for this task. As representatives of non-model bacteria, several genomes published by the Department of Biomedical Engineering were utilized, aiming to compare the OG assignments on the model as well as non-model bacteria. They are listed in Table 4.1. The genomes and other required data were downloaded from the Genbank database.

Tab. 4.1: The genomes chosen for comparison of annotation tools.

Bacterium	**AC Number**
Escherichia coli K-12 MG1655	NC_000913
Clostridium beijerinckii DSM 791	CP073653.1
Clostridium diolis DSM 15410	CP043998.1
Schlegelella thermodepolymerans DSM 15344	CP064338.1
Rhodospirillum rubrum DSM 467	CP077803.1
Tepidimonas taiwanensis LMG 22826	CP083911.1
Aneurinibacillus thermoaerophilus CCM 8960	CP090864.1

Already predicted genes are used as inputs for these tools, so functional annotation can be primarily compared.

The assignment of orthologous groups is observed in the first part of this comparison. From the processed outputs of the utilized tools the following outcomes can be deduced: How many percent of CDS was assigned to the particular OGs. To what extent, in percentage, did the tools match in their assignment of OGs, and finally to what percentage of results did at least two tools match.

The second part of the tool comparison is dedicated to the COG category assignment. In this case, only the percentage of tool matches is observed.

4.1 eggNOG-mapper

The eggNOG-mapper v2 is available online at `http://eggnog-mapper.embl.de/`. This tool is used to assign a group with its category from the eggNOG database to the predicted CDS, that is, a COG or other group that is part of this database. To annotate individual proteins, the proteins option was selected. The uploaded protein sequences were in FASTA format.

The six files are produced by the eggNOG-mapper (CSV, Excel, GFF, Orthologs, Seeds, and Proteins), one of which is important to obtain the required data - the GFF file. These outputs can be downloaded from the aforementioned website. It also offers an online exploration of annotation, an example of which is shown in Figure 4.1.

Query	Seed ortholog	e-value	Score	EggNOG OGs	Max Annotation Level	COG category
lcl|NC_000913.3_prot_NP_414627.1_86	316407.21321966	0.0	988.0	COG0769@1|root COG0769@2|Bacteria 1MU6P@1224|Proteobacteria 1RMD6@1236|Gammaproteobacteria 3XNJS@561|Escherichia	1236|Gammaproteobacteria	M

Fig. 4.1: An example of eggNOG-mapper annotation for one query sequence.

4.2 Operon-mapper

Operon-mapper is also available as an online tool at `https://biocomputo.ibt.unam.mx/operon_mapper/`. It assigns individual CDS to a COG or ROG and its category.

A FASTA sequence is required as an input for the genome annotation. To annotate individual CDSs, the ORF coordinates were uploaded in GFF format as an optional input.

Seven files are produced by the Operon-mapper as can be seen in Figure 4.2. They can be downloaded directly from the website, however, there is no option for online exploration of results. Predicted ORF coordinates and COG assignations are important for comparison purpose.

4.3 Batch CD-Search

Another way to assign proteins to their COGs is to use the Batch CD-search, available at: `https://www.ncbi.nlm.nih.gov/Structure/bwrpsb/bwrpsb.cgi`. Domain-model alignments are converted PSSMs and protein sequences can be scanned against these PSSMs with RPS-BLAST [68]. The maximal number of proteins the Batch CD-Search accepts is 4000. So in some cases, proteins had to be annotated in multiple jobs. They were uploaded in FASTA format and the COG database was chosen for the search.

Operon Mapper Job: *Escherichia coli*

Your Data	
FASTA	sequence with 1 sequences. 4,641,652 nucleotides (ecoli.fasta) [ascii file]
GFF	data with 9493 lines (ecoli.gff3) [ascii file]
e-mail address	not given
Job description (3840445)	*Escherichia coli*
Output Options	
1 Predicted operonic gene pairs	
2 Predicted Operons	
3 Predicted ORFs coordinates	
4 DNA sequences of the predicted ORFs	
5 Protein sequences of the translated predicted ORFs	
6 COGs assignations	
7 ORFs functional descriptions	
8 Compressed file with all output files	

Fig. 4.2: Operon-mapper's outputs.

The Batch CD-Search offers many options for online analysis and browsing results. A sample data which shows partial results is also available (see example in Figure 4.3). For the purpose of this comparison, domain hits were downloaded.

Query	Hit type	PSSM-ID	From	To	E-Value	Bitscore	Accession	Short name
Q#2 - >lcl|NC_000913.3_prot_NP_417730.1_3207	specific	224228	1	201	1.04619e-15	72.6637	COG1309	AcrR
Q#3 - >lcl|NC_000913.3_prot_NP_417731.1_3208	specific	223914	1	373	4.95141e-55	185.298	COG0845	AcrA
Q#4 - >lcl|NC_000913.3_prot_NP_417732.1_3209	specific	223911	1	1018	0	1253.28	COG0841	AcrB
Q#6 - >lcl|NC_000913.3_prot_YP_026209.1_3211	specific	226960	1	393	0	605.63	COG4597	BatB
Q#7 - >lcl|NC_000913.3_prot_NP_417736.2_3212	specific	223836	138	364	2.0904e-70	219.683	COG0765	HisM
Q#8 - >lcl|NC_000913.3_prot_NP_417737.1_3213	specific	224051	11	250	5.52834e-171	470.837	COG1126	GlnQ
Q#9 - >lcl|NC_000913.3_prot_NP_417738.4_3214	specific	223735	2	184	2.96873e-84	245.961	COG0663	PaaY

Fig. 4.3: The sample data of Batch CD-Search results.

4.4 Results for OG Assignment

Regarding the OG assignment for each CDS, the results are visualized in Figure 5.4 and complete data is available in the Attachment B. The eggNOG-mapper and Operon-mapper were able to achieve the highest percentage of assignment of OG for all genomes. This is due to the fact that they also assign groups from databases other than the COG, so they were able to assign groups from eggNOG or the ROG by Operon-mapper, respectively. The lowest numbers achieved the Batch CD-Search, which searches proteins against the COG database by RPS-BLAST. It did not assign even 65% of the CDS to the COG in most of the annotated genomes. This results from the fact that the COG database itself is not extensive enough to annotate non-model bacteria.

Looking at the results from a genomic perspective, eggNOG-mapper and Operon-mapper were able to handle the assignment comparably well for both *E. coli* and

non-model bacteria. However, in the Batch CD-Search the percentages dropped significantly for non-model genomes.

Furthermore, the similarity of results for each pair of tools was analyzed. Results can be seen in Figures 4.4 and 4.5, where the similarities for each pair of tools are represented by percentages separately for *Escherichia coli* and non-model bacteria, where the average similarity over all six non-model bacteria was calculated.

As can be observed in the results for the *E. coli*, the Batch CD-Search has the great match with eggNOG-mapper and Operon-mapper as the similarity is near 97% (Figure 4.4), while the match of eggNOG-mapper with Operon-mapper dropped to 80%. The percentage drop can be explained by the fact, that these tools search against two different databases (eggNOG and ROG), where the same group could be named differently. However, this idea cannot be confirmed as the ROG database is not publicly available. The same trend can also be observed for non-model bacteria (Figure 4.5).

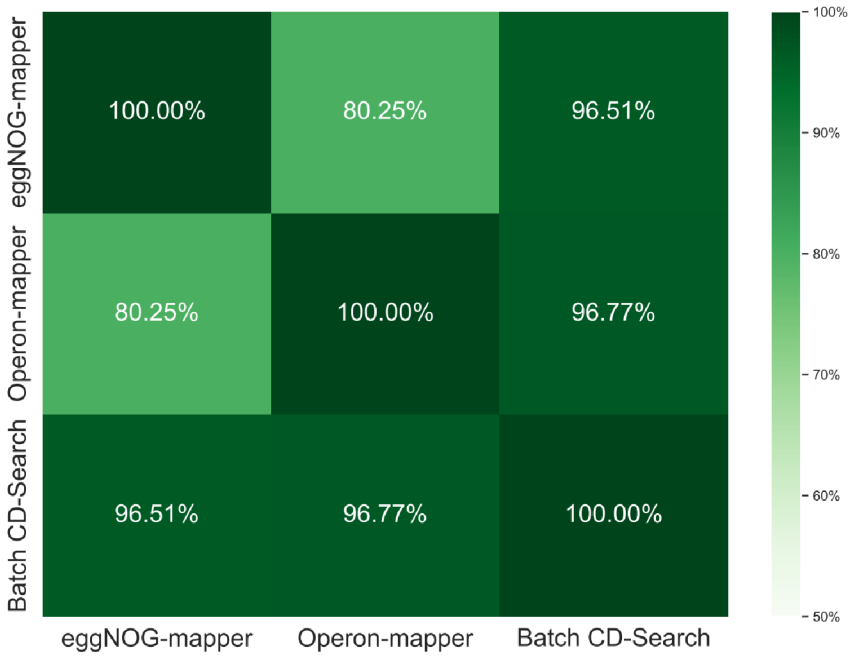


Fig. 4.4: Comparison of OG assignment by individual tools for *Escherichia coli*.

It was also concluded that if the OG has been assigned by all three tools, in 99.82% (for *Escherichia coli*) of these cases at least two tools matched, for non-model bacteria they matched in 99.22%. This fact can be used to further improve the functional annotation of bacterial genomes, which will be addressed in the next part of this thesis. Furthermore, it can be deduced from the observations that the COG database itself is not sufficient to annotate newly sequenced bacteria, and extended databases based on the automatic creation of OGs are required.

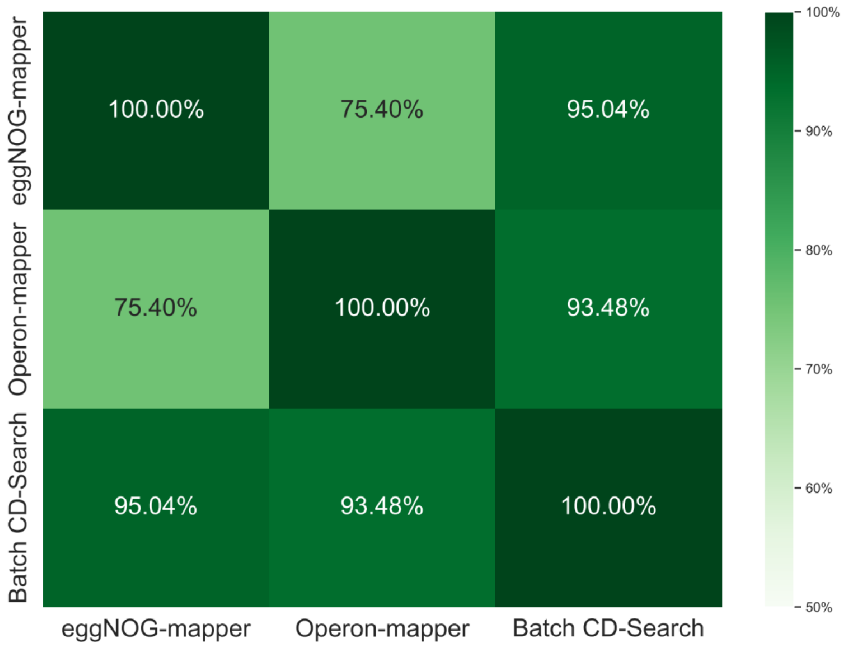


Fig. 4.5: Comparison of OG assignment by individual tools for non-model bacteria.

4.5 Results for Category Assignment

Regarding the category assignments, the similarity of results for each pair of tools was analyzed. Results can be seen in Figures 4.6 and 4.7, where the matches for each pair of tools are represented by percentages separately for *Escherichia coli* and non-model bacteria, just as the group comparison above.

Compared to heat-maps in Figures 4.4 and 4.5, the percentages dropped significantly for all tool pairs. Thanks to this comparison, a contradiction in categories between eggNOG, ROG, and COG databases was discovered. For example, in the eggNOG database, the COG4667 belongs to the category S, while in the COG database it belongs to the I category, and in the ROG database, into the R category.

Because of this, an analysis of the entire COG database and the eggNOG database was performed. The ROG database could not be analyzed because it is not publicly available. The comparison was carried out as follows. All OGs that the COG database and the eggNOG database have in common (4613 groups in total) were extracted to a file with their categories. This data is attached in the electronic attachment. If the COG database assigned a category that was not in the assignment by the eggNOG database, the mismatch was counted. A total of 1604 mismatches were found, representing 34.77% of their common groups.

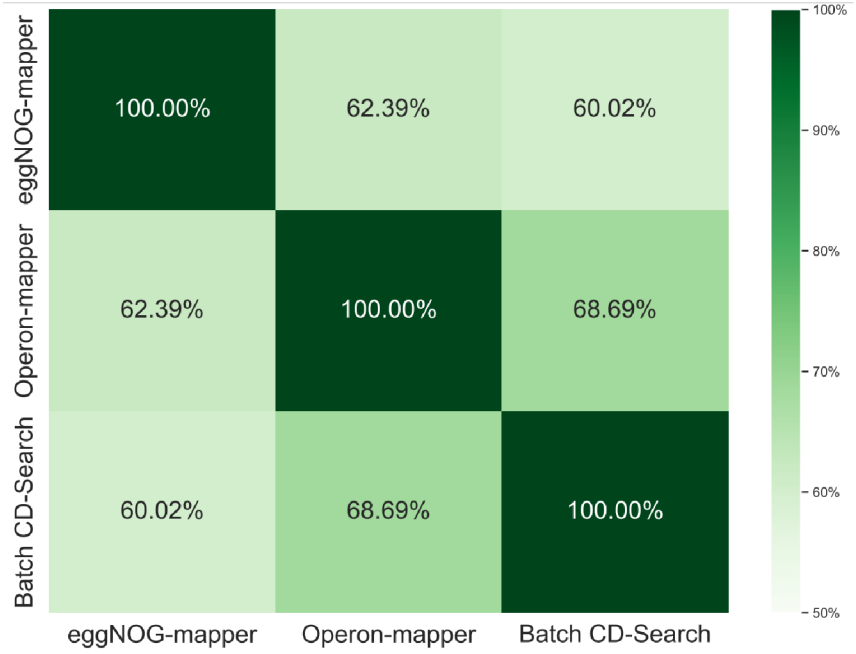


Fig. 4.6: Comparison of category assignment by individual tools for *Escherichia coli*.

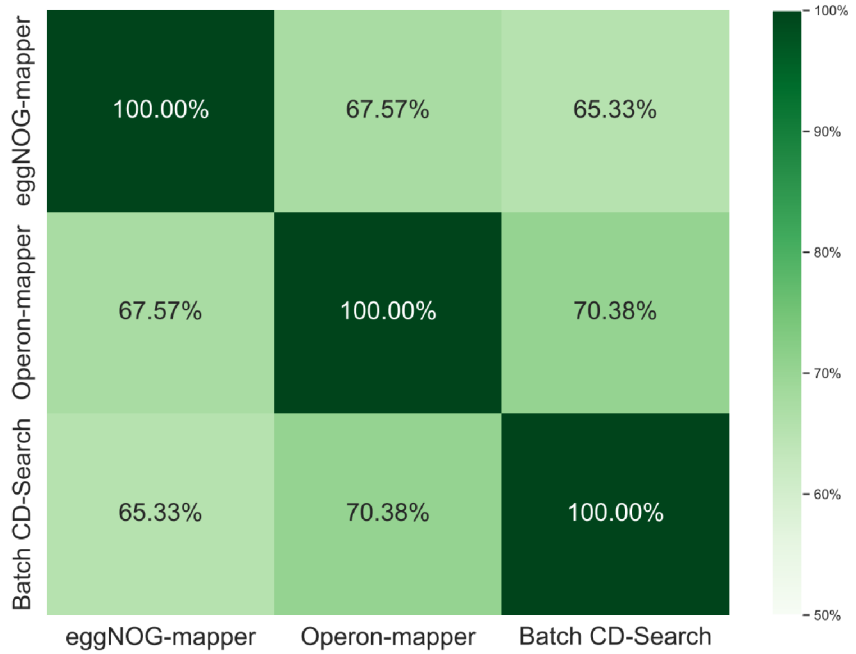


Fig. 4.7: Comparison of category assignment by individual tools for non-model bacteria.

5 COG-or

The previous chapters described the databases and tools for COG annotation that have been used so far. The main goal of this thesis is to propose and implement an improved method for COG annotation. Since it has been discovered, that at least two of the three best tools matched in approximately 99%, improvement can be achieved by combining the suitable tools, namely eggNOG-mapper, Operon-mapper, and Batch CD-Search.

The COG-or program was developed for this purpose, it is a python package that offers a more accurate assignment of genes to orthologous groups, a clearer output file, and the possibility to visualize the final annotation with DNAPlotter [77], which is an interactive Java application for generating circular and linear representations of genomes.

The COG-or is freely available at `https://github.com/xpolak37/the-COG-or` and in the electronic attachment of this thesis. It can be also downloaded using pip install command:

```
pip install COGor
```

5.1 The Main Idea

The main idea of COG-or is the combination of output files of the mentioned tools. Each tool performs sequence mapping with a different search tool, eggNOG-mapper uses DIAMOND, Operon-mapper works with hmmsearch and Batch CD-Search has RPS-BLAST for searching. By combining them, it is possible to achieve more reliable results and a greater percentage of coverage of the whole genome.

However, the output files of these tools are confusing and hard to be visualized. Moreover, there is a contradiction in categories between their databases. The COG-or package works with the output files of these tools, modifies them appropriately, and generates one final file containing the consensus assignment of OGs. The whole process of using the package is illustrated in Figure 5.1. The yellow blocks are part of the COG-or.

At first, a CDS file in FASTA format is necessary for the eggNOG-mapper and Batch CD-Search. The input of the Operon-mapper tool should be the genomic sequence in FASTA format along with a GFF file containing the coordinates of the individual features (CDS, pseudogenes, RNA genes).

The outputs of these tools are processed by the COG-or into the unified GFF files with adjusted categories according to the COG database and a consensus from their groups assignment is created. That means, it assigns an OG if at least two

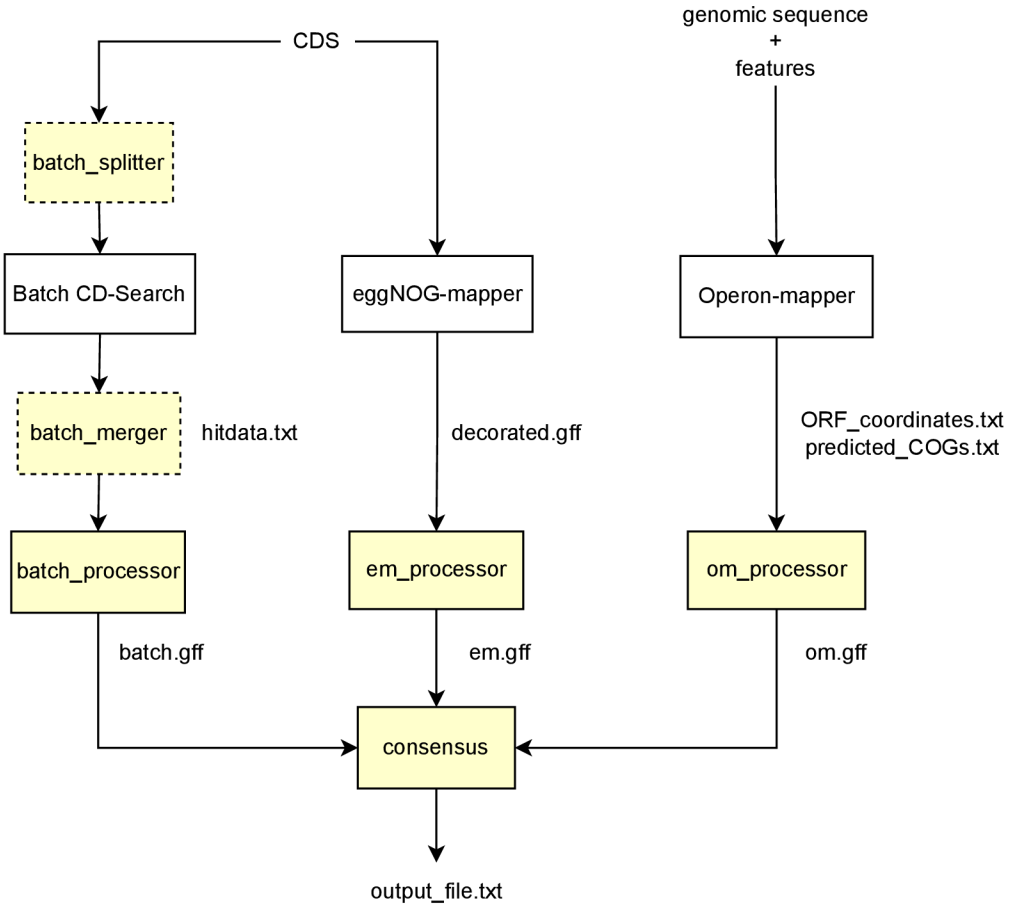


Fig. 5.1: Schematic diagram of annotation process using the COG-or.

tools agree in its assignment. If such a situation does not occur, it assigns the most appropriate output.

The final output of the COG-or package is an improved annotation in GFF format together with the genome sequence in FASTA format. This file is suitable for visualization in the DNAPlotter, which renders the bacterial genome and the 26 COG categories (listed in Table 3.1), distinguished by color.

The user can use individual functions from the package and run the steps himself or run the whole program using a simple command:

```
py cogor.py -n organism_name -i input_path -o output_path -t
```

The COG-or documentation with tutorial is available in the electronic attachment.

5.2 Implementation

The package is implemented in Python language (version 3) and consists of three different modules: Program processor, Consensus, and Track manager.

5.2.1 Program Processor

Program processor module consists of five functions: *em_processor*, *om_processor*, *batch_processor*, *batch_splitter*, and *batch_merger*. The first three of them are used to process the output files from the tools eggNOG-mapper, Operon-mapper, and Batch CD-Search. As the outputs from these tools are each slightly different, each function is developed to process a specific output file. The processed data are saved into a unified GFF file, which contains *sequence id, source, feature type, start, end, score, strand, frame*, and *attribute* with CDS id, COG number, and COG category. Furthermore, in Chapter 4 it was also discovered that there is a contradiction in categories between eggNOG, ROG, and COG database. Therefore, the categories are rewritten according to the COG database (data were downloaded from `https://ftp.ncbi.nih.gov/pub/COG/COG2020/data/cog-20.def.tab`) since it is managed by NCBI. If the attributed groups were not part of the COG database, the categories were retained. For reasons of subsequent plotting and analysis, only the first attributed category was always retained.

Functions *batch_splitter*, and *batch_merger* can be used to simplify work with the Batch CD-Search tool because of its inability to annotate more than 4000 sequences at once. In case the bacterium contains more CDSs, the user can split the single CDS file in half and merge them again after annotation. Inputs and outputs of these functions can be seen in Figure 5.1.

5.2.2 Consensus

The next step to improve functional annotation is to combine the processed outputs of the Program processor. The Consensus module contains three functions: *consensus* and additional functions *read_file* and *get_features* used by this function. The flowchart describing function's algorithm is depicted in the Attachment A.

The *consensus* function is based on the idea that the most reliable tool is the eggNOG-mapper, which also contains the most information about a given CDS. Therefore, if deciding between two tools, the function selects eggNOG-mapper. The Batch CD-Search tool is selected only if no other tool has assigned the COG, otherwise, it mainly serves as a confirmation of the assignment. The inputs of this function are, in addition to the processed files from the previous step, a FASTA file, and information about whether the user wants to add additional features to the

annotation, such as pseudogenes or ncRNAs, which are not included in the tools. In this case, an additional file in GFF format needs to be uploaded.

The last step of this function is to add the genomic sequence in FASTA format in addition to previously produced data. This output file is ready to be visualized in DNAPlotter.

5.2.3 Track Manager

To visualize the genome with the differentiation of each COG category, the consensus file alone is not entirely sufficient. In DNAPlotter, after loading the file, it plots the genome according to the specified features without distinguishing individual COG categories as shown in Figure 5.2 (left). To choose the color of the COG categories and other features, the user has to import another file in the Track Manager option of DNAPlotter. This track template can be generated along with the legend by the two functions from the Track manager module: *get_track_template* and *get_legend*.

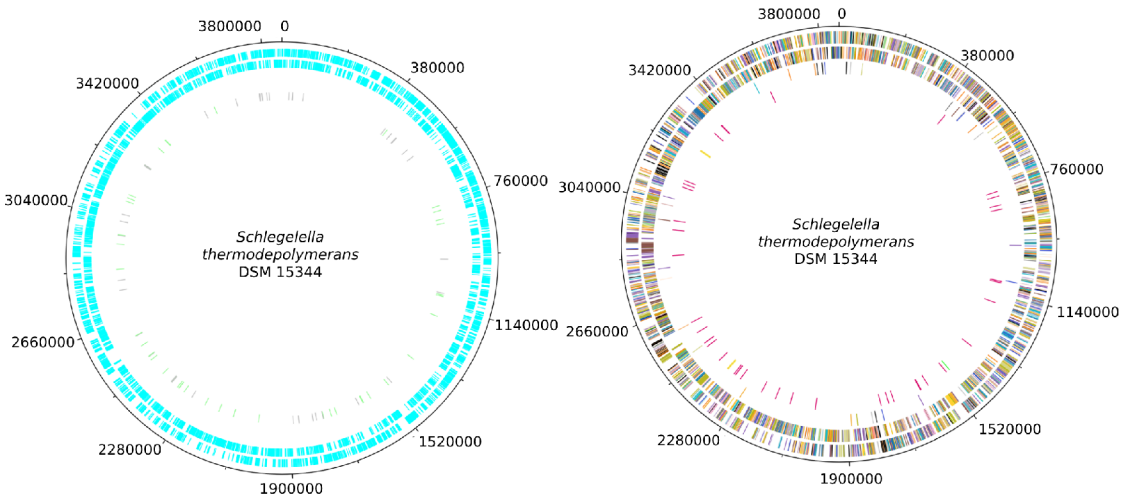


Fig. 5.2: Comparison of the primary visualisation (left) and one after the track template is uploaded (right) for the genome of *Schlegella thermodepolymerans* DSM 15344.

The function *get_track_template* generates the file with plot settings, with which four circles can be drawn - from outermost: CDS forward strand, CDS reverse strand, pseudogenes, and RNA genes.

This file contains 84 lines and 12 columns. Each column represents a setting parameter and each line represents an individual feature to plot. A simplified structure can be seen in Table 5.1.

With this data, a couple of features can be plotted: CDS and pseudogene which COG belongs to the J category, CDS and pseudogene to which no COG has been attributed, and RNA genes. The color is set in the last column in RGB format. The user can set the track positions and track size himself, other settings are firmly established.

After uploading the file from this function to DNAPlotter, the genomic map shown in Figure 5.2 (right) can be obtained.

Tab. 5.1: A simplified table generated by the *get_track_template*.

Position	Size	Forward	Reverse	Not	Any	Key	Qualifier	Value	Color
0.95	10.0	true	false	false	false	CDS	CAT	J	30:117:176
0.9	10.0	false	true	false	false	CDS	CAT	J	30:117:176
0.95	10.0	true	false	false	false	CDS	CAT	-	0:0:0
0.9	10.0	false	true	false	false	CDS	CAT	-	0:00:00
0.85	10.0	true	true	false	false	pseudogene	CAT	J	30:117:176
0.85	10.0	true	true	false	false	pseudogene	CAT	-	0:0:0
0.8	10.0	true	true	false	false	tRNA	nul	null	217:3:104
0.8	10.0	true	true	false	false	rRNA	nul	null	250:243:62
0.8	10.0	true	true	false	false	ncRNA	nul	null	30:252:30

Since DNAPlotter does not generate a legend for the plot of the genome, the user can generate one using the *get_legend* function. This function uses the PIL python package to create an image in PNG format with an explanatory color legend.

5.3 Results and Discussion

At first, the computational time was measured on several bacteria with different numbers of CDS (listed in Table 5.2). This analysis was performed using Measure-Command PowerShell command. It measures the time it takes to execute the COG-or tool in the command line. This command was used as follows:

```
Measure-Command {py cogor.py -n organism_name -i input_path
-o output_path -t}
```

These measured data points were approximated by a suitable curve as shown in Figure 5.3. The COG-or has linear time complexity *O(n)*. It means that the execution takes proportionally longer as the number of CDS grows.

Tab. 5.2: The chosen genomes for computation of time complexity of the COG-or.

Bacterium	**Number of CDS**
Candidatus Hodgkinia cicadicola TETCHI1b	84
Mycoplasma yeatsii GM274B	754
Campylobacter jejuni CFSAN054107	1853
Aneurinibacillus thermoaerophilus CCM 8960	3589
Rhodospirillum rubrum DSM 467	3850
Clostridium beijerinckii DMS 791	5061
Streptomyces coeruleorubidus ATCC 13740	8398
Sorangium cellulosum So0157-2	10884
Minicystis rosea DSM 24000	14018

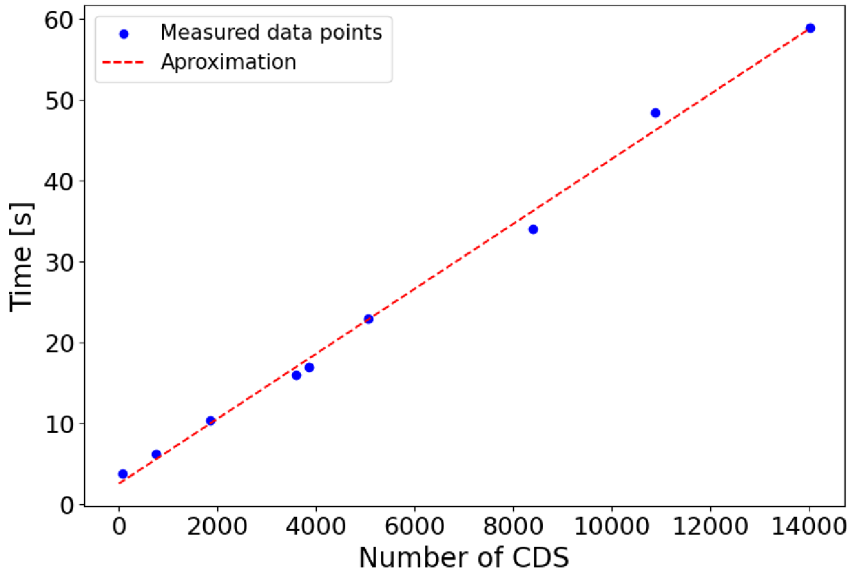


Fig. 5.3: Time complexity of the COG-or.

Additionaly, the developed tool was tested on seven bacteria which were also used in the tool comparison. They are listed in Table 4.1. For each genome, several files were downloaded from the Genbank database as inputs for the annotation process: bacterial genome in FASTA format, its features in GFF3, and CDS file in FASTA format. These genomes were annotated by eggNOG-mapper, also by Batch CD-Search with CDS as input, and lastly using Operon-mapper with FASTA and GFF serving as inputs. Retrieved outputs were used as inputs for the COG-or package.

The total numbers of CDS annotated by different tools are visualized in Figure 5.4 and the complete data is available in the Attachment B. In all cases, the COG-or increased the percentage of assigned OGs. For example, in *A. thermoaerophilus*, it was able to increase the percentage of OG assignments by 4.21%.

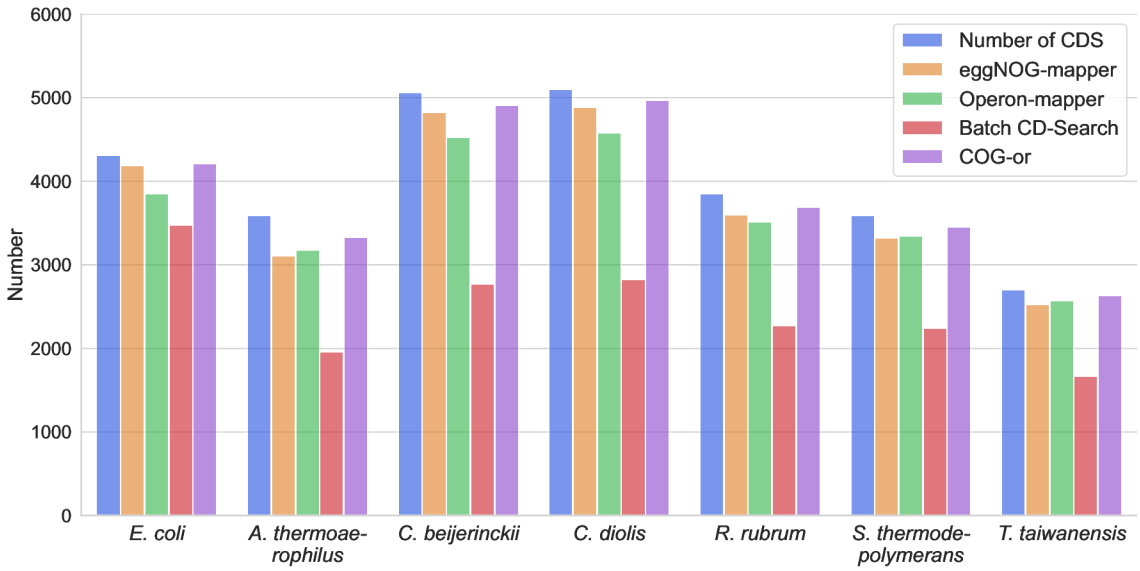


Fig. 5.4: Tools comparison with COG-or in OG assignment.

However, neither combination of the three tools was sufficient to assign 100% of CDS to its OG. This may be due to the fact that existing databases of orthologous genes are not yet complex enough to identify all proteins, despite the extensive amount of data in the eggNOG database.

For the results obtained using the COG-or, confusion matrices were constructed (available in Attachment C) and the precision and recall for each tool were determined (Table 5.4). Each one of the confusion matrices was constructed according to the rules listed in the Table 5.3. This time, all data were computed together from all bacteria separately after adjusting the categories according to the COG database.

Tab. 5.3: The chosen rules for confusion matrix construction.

Classification	**Description**
TP	The tools agreed in assignment
TN	Both tools were unable to assign COG
FP	Assigned COG of COG-or and analyzed tool differed
FN	COG-or assigned COG while the analyzed tool did not

Tab. 5.4: Precision and recall values for the annotation tools.

	OGs		Categories	
	Precision [%]	Recall [%]	Precision [%]	Recall [%]
eggNOG-mapper	100	98.64	100	98.64
Operon-mapper	91.28	81.67	95.17	82.28
Batch CD-Search	95.24	63.18	97.86	63.81

The highest precision and recall can be observed for eggNOG-mapper. When analysing groups and categories, precision reached 100% and recall almost 99%. This analysis confirms the results in Figure 5.4, where it can be seen that the eggNOG-mapper attributed the most number of CDS to their group and category, respectively. The Operon-mapper reached the smallest precision values. This means that COG-or changed its assignment in the most cases among the used tools. This comes from the fact deducted from results in Figures 4.4 and 4.5 that the Operon-mapper has the lowest agreement with the other tools. The smallest recall values were reached by the Batch CD-Search. It is also clear from the former results that Batch CD-Search assigned the least CDS to their orthologous group (Figure 5.4).

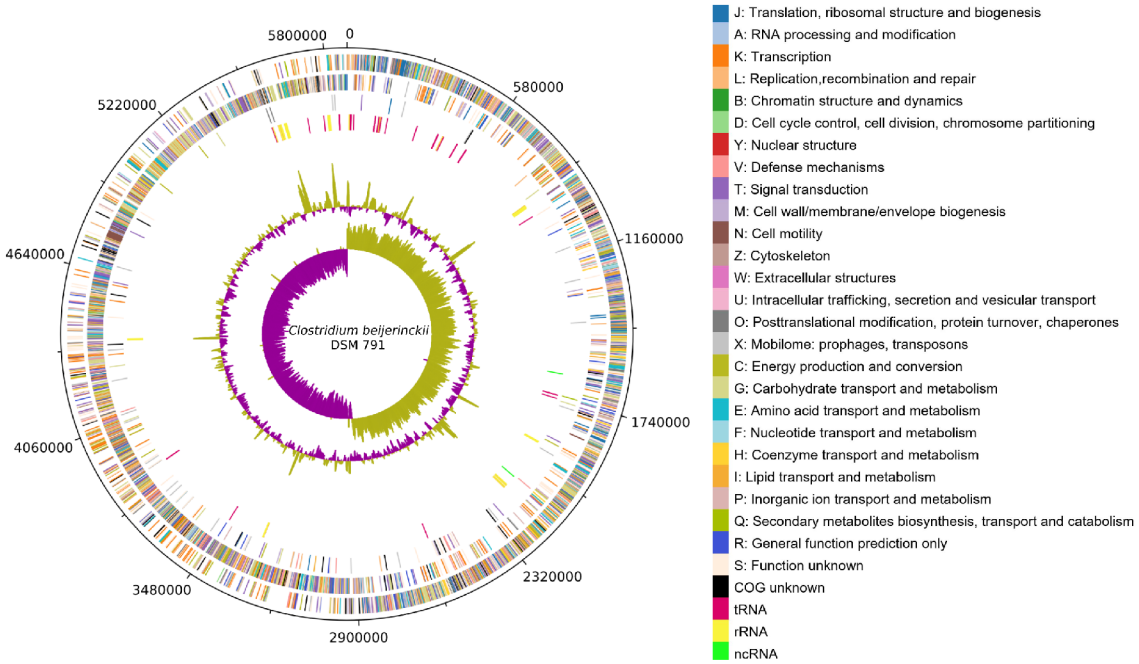


Fig. 5.5: The chromosomal map of *Clostridium beijerinckii* DSM 791 drawn using DNAPlotter and the COG-or.

The outputs of an analysis performed with the COG-or were visualized with DNAPlotter. The chromosomal map of *Clostridium beijerinckii* DSM 791 is shown

in Figure 5.5 as an example. The various colors represent the 26 COG categories as well as RNA genes. The first and the second circles show CDSs on the forward and reverse strands. The third circle represents pseudogenes and in the fourth circle, the RNA genes are plotted. The two inner circles represent GC content and GC skew.

Finally, the distribution of COG categories in annotated bacteria was observed (see Figure 5.6). The graph represents the relative abundance of these categories (y-axis) within annotated bacteria (x-axis).

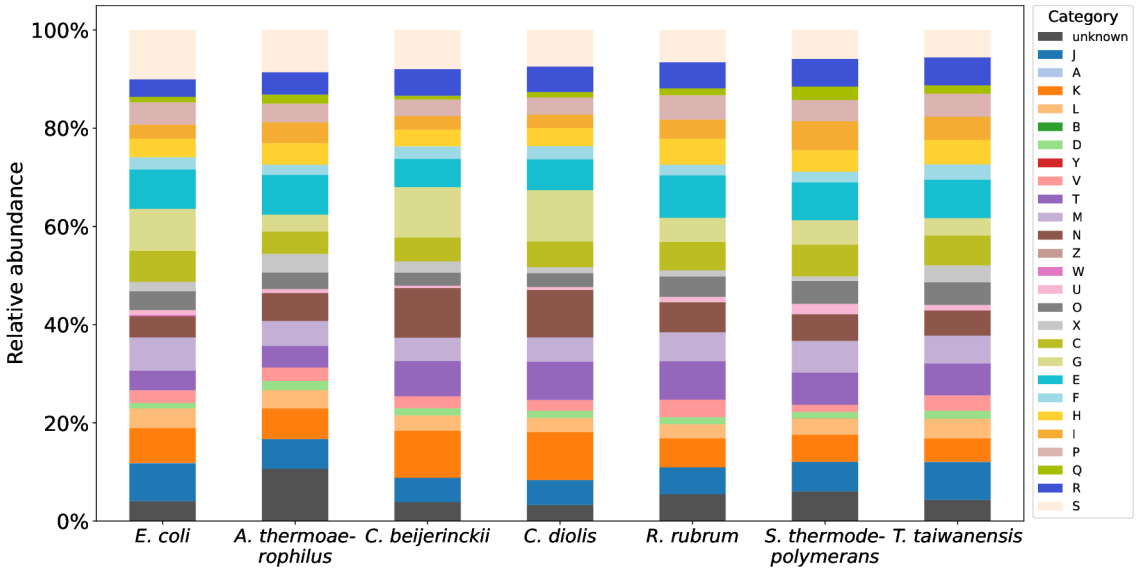


Fig. 5.6: Distribution of COG categories in annotated bacteria.

Whereas orthologous groups represent a phylogenetic classification of the proteins, such visualization can identify the functional coding potential of the bacterium and its phylogenetic analysis. At the first sight, it is clear that the distribution of COG categories in *Clostridium beijerinckii* is highly similar to the distribution in *Clostridium diolis*, because they are different strains of the same bacterium. Principal component analysis of the COG distribution is represented in Figure 5.7 to confirm this claim. A strong similarity between these two bacteria can be seen in this visualization. Furthermore, an explicit difference between the model bacterium and non-model bacteria emerged. The validity of this reasoning was verified also by the construction of a GBDP phylogenetic tree based on 16S rDNA gene sequence using the Type Strain Genome Server (TYGS) [78] (see Figure 5.8).

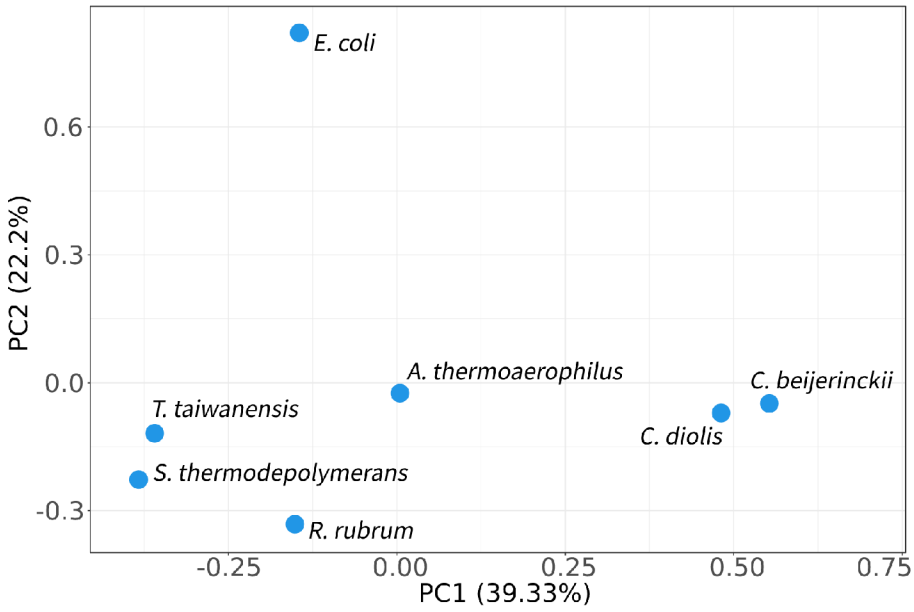


Fig. 5.7: Principal component analysis (PCA) plot based on the relative abundance of COG categories in the annotated bacteria.

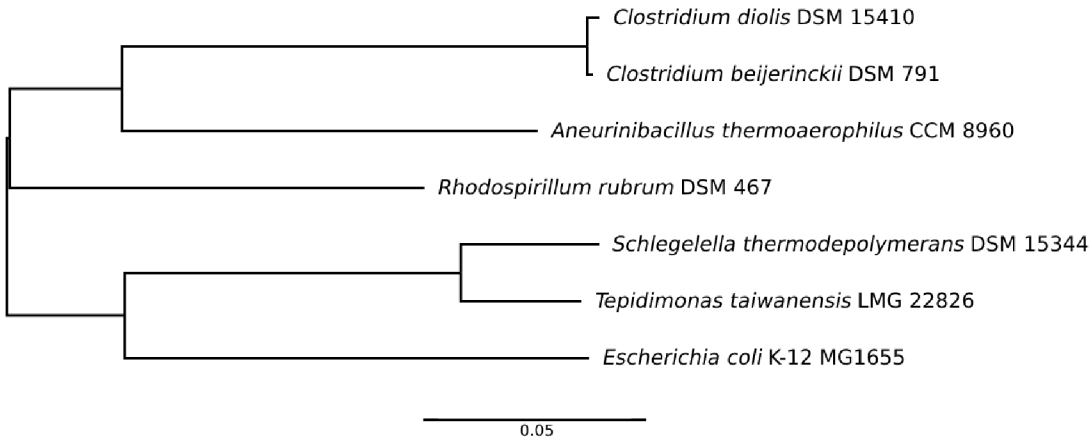


Fig. 5.8: GBDP tree of the annotated bacteria.

However, despite the significant improvement in annotation results, the large portions of bacterial genomes are still genes of unknown function (S category). This is something that cannot be significantly improved by any available tool or database at the moment. Even though genomes can be readily accessed with sequencing, the lack of functional information is still a major challenge in microbiological research.

Conclusion

The main topic of this thesis was to describe a functional annotation of the bacterial genome using clusters of orthologous genes to compare three tools that use COG annotation, and finally, to propose and implement the improved method for functional annotation of bacterial genomes using this approach.

The first part of the thesis was dedicated to the bacterial genome and its annotation. Afterwards, COG annotation was described with its principles and advantages. The two databases and tools that use this approach were analyzed. Firstly, the COG database, which is the first database of orthologous genes that is based on manually curated groups, and secondly, the eggNOG database, which contains automatically curated groups on a much larger sample of organisms.

The fourth chapter focused on the comparison of three tools that use OG assignment for the prediction of genes' function: eggNOG-mapper, Operon-mapper, and Batch CD-Search. These tools were described from a user's point of view, and they were further compared by annotating seven bacteria, which are listed in Table 4.1.

For each tool, the number of assigned CDSs to the particular OGs was determined (Figure 5.4). Then, the matches in OG and categories assignment for each pair of tools were expressed as percentages separately for *Escherichia coli* and non-model bacteria (Figures 4.4, 4.5, 4.6, and 4.7).

The eggNOG-mapper achieved the highest percentage of assignment of OG for all genomes in this comparison. The next most efficient tool was the Operon-mapper and the last was the Batch CD-Search.

These results indicate that the COG database itself is not sufficient to annotate new non-model bacteria, so an extended database must be used even though it contains automatically generated groups. Furthermore, a contradiction in categories between the eggNOG, ROG, and COG database was discovered. Specifically, the COG and eggNOG databases differ by more than 34% in their categories.

In the final chapter, a newly developed tool (the COG-or), which improves the functional annotation of bacterial genomes, was described. It was implemented in Python language and its main idea is to combine three suitable tools for COG annotation, namely eggNOG-mapper, Operon-mapper, and Batch CD-Search. It can work efficiently with the outputs of these tools to further improve the assignment of individual CDSs to their COG group and category. This tool was tested on the annotation of seven bacterial genomes, which were also used for comparison purposes in Chapter 4. Thanks to the combination of different approaches, the percentage of assigned OGs was improved in all cases. The COG-or can also be used for visualization of final annotation by exporting its generated files into DNAPlotter and thus distinguishing individual features and COG categories.

Despite this improvement of the COG annotation through the consensus of different tools, a certain percentage of genomes unassigned to the particular OG still remains. It is mainly because the existing databases are not yet complex enough to identify all proteins. Also, large portions of bacterial genomes are still genes of unknown function and their clarification remains a major challenge in microbiological research.

Bibliography

[1] ALBERTS, Bruce, 2008. *Molecular biology of the cell.* 5th ed. New York: Garland Science. ISBN 978-0-8153-4105-5.

[2] Genome Anatomies, 2002. BROWN, Terence A. *Genome* [online]. 2. Oxford: Wiley-Liss, chapter 2. ISBN 0-471-25046-5. Available at: https://www.ncbi.nlm.nih.gov/books/NBK21120/

[3] SUMMERS, Anne, MRÁZEK, Jan, ed., 2008. General Characteristics of Prokaryotic Genomes. XU, Ying a J. Peter GOGARTEN. *Computational methods for understanding bacterial and archaeal genomes.* London: Imperial College Press; Distributed by World Scientific Publishing, s. 1-37. ISBN 978-1-86094-982-1.

[4] OCHMAN, Howard and Alejandro CARO-QUINTERO, 2016. Genome Size and Structure, Bacterial. *Encyclopedia of Evolutionary Biology* [online]. Elsevier, 2016, s. 179-185. ISBN 9780128004265. Available at: https://doi.org/10.1016/B978-0-12-800049-6.00235-3

[5] DORMAN, Charles J., 2020. *Structure and Function of the Bacterial Genome* [online]. Wiley. ISBN 9781119308799. Available at: https://doi.org/10.1002/9781119309697

[6] SHUMAN, Howard A. and Thomas J. SILHAVY, 2003. The art and design of genetic screens: Escherichia coli. *Nature Reviews Genetics* [online]. **4**(6), 419-431. ISSN 1471-0056. Available at: https://doi.org/10.1038/nrg1087

[7] MOJICA, Francisco J.M., César DÍEZ-VILLASEÑOR, Jesús GARCÍA-MARTÍNEZ and Elena SORIA, 2005. Intervening Sequences of Regularly Spaced Prokaryotic Repeats Derive from Foreign Genetic Elements. *Journal of Molecular Evolution* [online]. **60**(2), 174-182 . ISSN 0022-2844. Available at: https://doi.org/10.1007/s00239-004-0046-3

[8] MARRAFFINI, Luciano A., 2015. CRISPR-Cas immunity in prokaryotes. *Nature* [online]. **526**(7571), 55-61. ISSN 0028-0836. Available at: https://doi.org/10.1038/nature15386

[9] BARRANGOU, Rodolphe, 2015. The roles of CRISPR—Cas systems in adaptive immunity and beyond. *Current Opinion in Immunology* [online]. **32**, 36-41. ISSN 09527915. Available at: https://doi.org/10.1016/j.coi.2014.12.008

[10] WESTERMANN, Lukas, Björn NEUBAUER and Michael KÖTTGEN, 2021. Nobel Prize 2020 in Chemistry honors CRISPR: a tool for rewriting the code of life. *Pflügers Archiv - European Journal of Physiology* [online]. **473**(1), 1-2. ISSN 0031-6768. Available at: https://doi.org/10.1007/s00424-020-02497-9

[11] PEVSNER, Jonathan, 2015. *Bioinformatics and functional genomics*. Third edition. Hoboken, New Jersey: John Wiley. ISBN 978-1-118-58178-0.

[12] SESHASAYEE, Aswin Sai Narain, 2015. *Bacterial genomics: genome organization and gene expression tools*. Delhi: Cambridge University Press. ISBN 978-1-107-07983-0.

[13] FLEISCHMANN, Robert D., Mark D. ADAMS, Owen WHITE, et al., 1995. Whole-Genome Random Sequencing and Assembly of Haemophilus influenzae Rd. *Science* [online]. **269**(5223), 496-512. ISSN 0036-8075. Available at: https://doi.org/10.1126/science.7542800

[14] HODGMAN, Charlie, Andrew FRENCH and David WESTHEAD, 2009. *BIOS Instant Notes in Bioinformatics* [online]. Taylor & Francis. ISBN 9781134158874. Available at: https://doi.org/10.4324/9780203967249

[15] BECKLOFF, Nicholas, Shawn STARKENBURG, Tracey FREITAS and Patrick CHAIN, 2012. Bacterial Genome Annotation. NAVID, Ali, ed. *Microbial Systems Biology* [online]. Totowa, NJ: Humana Press, 2012-4-20, s. 471-503. Methods in Molecular Biology. ISBN 978-1-61779-826-9. Available at: https://doi.org/10.1007/978-1-61779-827-6_16

[16] BATEMAN, Alex, Maria-Jesus MARTIN, Sandra ORCHARD, et al., 2021. UniProt: the universal protein knowledgebase in 2021. *Nucleic Acids Research* [online]. **49**(D1), D480-D489. ISSN 0305-1048. Available at: https://doi.org/10.1093/nar/gkaa1100

[17] NCBI Prokaryotic Genome Annotation Pipeline. *NCBI* [online]. Bethesda (MD): National Center for Biotechnology Information, U.S. National Library of Medicine [cit. 13.11.2021]. Available at: https://www.ncbi.nlm.nih.gov/genome/annotation_prok/

[18] MATTICK, John S. and Igor V. MAKUNIN, 2006. Non-coding RNA. *Human Molecular Genetics* [online]. **15**(1). ISSN 1460-2083. Available at: https://doi.org/10.1093/hmg/ddl046

[19] HÜTTENHOFER, Alexander, Peter SCHATTNER and Norbert POLACEK, 2005. Non-coding RNAs: hope or hype? *Trends in Genetics* [online]. **21**(5), 289-297. ISSN 01689525. Available at: https://doi.org/10.1016/j.tig.2005.03.007

[20] CAMACHO, Christiam, George COULOURIS, Vahram AVAGYAN, Ning MA, Jason PAPADOPOULOS, Kevin BEALER and Thomas L MADDEN, 2009. BLAST+: architecture and applications. *BMC Bioinformatics* [online]. **10**(1). ISSN 1471-2105. Available at: https://doi.org/10.1186/1471-2105-10-421

[21] EDDY, Sean R. and William R. PEARSON, 2011. Accelerated Profile HMM Searches. *PLoS Computational Biology* [online]. **7**(10). ISSN 1553-7358. Available at: https://doi.org/10.1371/journal.pcbi.1002195

[22] MÉDIGUE, Claudine and Ivan MOSZER, 2007. Annotation, comparison and databases for hundreds of bacterial genomes. *Research in Microbiology* [online]. **158**(10), 724-736. ISSN 09232508. Available at: https://doi.org/10.1016/j.resmic.2007.09.009

[23] PUNTA, Marco, Penny C. COGGILL, Ruth Y. EBERHARDT, et al., 2011. The Pfam protein families database. *Nucleic Acids Research* [online]. **40**(D1), D290-D301. ISSN 0305-1048. Available at: https://doi.org/10.1093/nar/gkr1065

[24] MITCHELL, Alex L., Teresa K. ATTWOOD, Patricia C. BABBITT, et al., 2019. InterPro in 2019: improving coverage, classification and access to protein sequence annotations. *Nucleic Acids Research* [online]. **47**(D1), D351-D360. ISSN 0305-1048. Available at: https://doi.org/10.1093/nar/gky1100

[25] LU, Shennan, Jiyao WANG, Farideh CHITSAZ, et al., 2020. CDD/SPARCLE: the conserved domain database in 2020. *Nucleic Acids Research* [online]. **48**(D1), D265-D268. ISSN 0305-1048. Available at: https://doi.org/10.1093/nar/gkz991

[26] GALPERIN, Michael Y, David M KRISTENSEN, Kira S MAKAROVA, Yuri I WOLF and Eugene V KOONIN, 2019. Microbial genome analysis: the COG approach. *Briefings in Bioinformatics* [online]. **20**(4), 1063-1070. ISSN 1467-5463. Available at: https://doi.org/10.1093/bib/bbx117

[27] KOONIN, Eugene V., 2005. Orthologs, Paralogs, and Evolutionary Genomics. *Annual Review of Genetics* [online]. **39**(1), 309-338. ISSN 0066-4197. Available at: https://doi.org/10.1146/annurev.genet.39.073003.114725

[28] TATUSOV, Roman L., Eugene V. KOONIN and David J. LIPMAN, 1997. A Genomic Perspective on Protein Families. *Science* [online]. **278**(5338), 631-637. ISSN 0036-8075. Available at: https://doi.org/10.1126/science.278.5338.631

[29] TATUSOV, Roman. The COG database: a tool for genome-scale analysis of protein functions and evolution. *Nucleic Acids Research* [online]. **28**(1), 33-36. ISSN 13624962. Available at: https://doi.org/10.1093/nar/28.1.33

[30] SEEMANN, Torsten, 2014. Prokka: rapid prokaryotic genome annotation. *Bioinformatics* [online]. **30**(14), 2068-206. ISSN 1367-4803. Available at: https://doi.org/10.1093/bioinformatics/btu153

[31] TATUSOVA, Tatiana, Michael DICUCCIO, Azat BADRETDIN, et al., 2016. NCBI prokaryotic genome annotation pipeline. *Nucleic Acids Research* [online]. **44**(14), 6614-6624. ISSN 0305-1048. Available at: https://doi.org/10.1093/nar/gkw569

[32] AZIZ, Ramy K., Daniela BARTELS, Aaron A .BEST, et al., 2008. The RAST Server: Rapid Annotations using Subsystems Technology. *BMC Genomics* [online]. **9**(1). ISSN 1471-2164. Available at: https://doi.org/10.1186/1471-2164-9-75

[33] HYATT, Doug, Gwo-Liang CHEN, Philip F LOCASCIO, Miriam L LAND, Frank W LARIMER and Loren J HAUSER, 2010. Prodigal: prokaryotic gene recognition and translation initiation site identification. *BMC Bioinformatics* [online]. **11**(1). ISSN 1471-2105. Available at: https://doi.org/10.1186/1471-2105-11-119

[34] LAGESEN, Karin, Peter HALLIN, Einar Andreas RØDLAND, Hans-Henrik STÆRFELDT, Torbjørn ROGNES and David W. USSERY, 2007. RNAmmer: consistent and rapid annotation of ribosomal RNA genes. *Nucleic Acids Research* [online]. **35**(9). ISSN 1362-4962. Available at: https://doi.org/10.1093/nar/gkm160

[35] LASLETT, Dean, 2004. ARAGORN, a program to detect tRNA genes and tmRNA genes in nucleotide sequences. *Nucleic Acids Research* [online]. **32**(1), 11-16. ISSN 1362-4962. Available at: https://doi.org/10.1093/nar/gkh152

[36] ALMAGRO ARMENTEROS, José Juan, Konstantinos D. TSIRIGOS, Casper Kaae SØNDERBY, Thomas Nordahl PETERSEN, Ole WINTHER, Søren BRUNAK, Gunnar VON HEIJNE and Henrik NIELSEN, 2019. SignalP 5.0 improves signal peptide predictions using deep neural networks. *Nature Biotechnology* [online]. **37**(4), 420-423. ISSN 1087-0156. Available at: https://doi.org/10.1038/s41587-019-0036-z

[37] KOLBE, Diana L. and Sean R. EDDY, 2011. Fast filtering for RNA homology search. *Bioinformatics* [online]. **27**(22), 3102-3109. ISSN 1367-4803. Available at: https://doi.org/10.1093/bioinformatics/btr545

[38] HAFT, Daniel H., Michael DICUCCIO, Azat BADRETDIN, et al., 2018. RefSeq: an update on prokaryotic genome annotation and curation. *Nucleic*

Acids Research [online]. **46**(D1), D851-D860. ISSN 0305-1048. Available at: https://doi.org/10.1093/nar/gkx1068

[39] HAFT, Daniel H., Jeremy D. SELENGUT, Roland A. RICHTER, Derek HARKINS, Malay K. BASU and Erin BECK, 2012. TIGRFAMs and Genome Properties in 2013. *Nucleic Acids Research* [online]. **41**(D1), D387-D395. ISSN 0305-1048. Available at: https://doi.org/10.1093/nar/gks1234

[40] LI, Wenjun, Kathleen R. O-NEILL, Daniel H. HAFT, et al., 2021. RefSeq: expanding the Prokaryotic Genome Annotation Pipeline reach with protein family model curation. *Nucleic Acids Research* [online]. **49**(D1), D1020-D1028. ISSN 0305-1048. Available at: https://doi.org/10.1093/nar/gkaa1105

[41] KLIMKE, William, Richa AGARWALA, Azat BADRETDIN, et al., 2009. The National Center for Biotechnology Information's Protein Clusters Database. *Nucleic Acids Research* [online]. **37**(1), D216-D223. ISSN 1362-4962. Available at: https://doi.org/10.1093/nar/gkn734

[42] LOMSADZE, Alexandre, Karl GEMAYEL, Shiyuyun TANG and Mark BORODOVSKY, 2018. Modeling leaderless transcription and atypical genes results in more accurate gene prediction in prokaryotes. *Genome Research* [online]. **28**(7), 1079-1089. ISSN 1088-9051. Available at: https://doi.org/10.1101/gr.230615.117

[43] NAWROCKI, Eric P., Sarah W. BURGE, Alex BATEMAN, et al., 2015. Rfam 12.0: updates to the RNA families database. *Nucleic Acids Research* [online]. **43**(D1), D130-D137. ISSN 1362-4962. Available at: https://doi.org/10.1093/nar/gku1063

[44] CHAN, Patricia P. and Todd M. LOWE, 2019. TRNAscan-SE: Searching for tRNA Genes in Genomic Sequences. KOLLMAR, Martin, ed. *Gene Prediction* [online]. New York, NY: Springer New York, 2019-04-25, s. 1-14. Methods in Molecular Biology. ISBN 978-1-4939-9172-3. Available at: https://doi.org/10.1007/978-1-4939-9173-0_1

[45] EDGAR, Robert C, 2007. PILER-CR: Fast and accurate identification of CRISPR repeats. *BMC Bioinformatics* [online]. **8**(1). ISSN 1471-2105. Available at: https://doi.org/10.1186/1471-2105-8-18

[46] BLAND, Charles, Teresa L RAMSEY, Fareedah SABREE, Micheal LOWE, Kyndall BROWN, Nikos C KYRPIDES and Philip HUGENHOLTZ, 2007. CRISPR Recognition Tool (CRT): a tool for automatic detection of clustered

regularly interspaced palindromic repeats. *BMC Bioinformatics* [online]. **8**(1). ISSN 1471-2105. Available at: https://doi.org/10.1186/1471-2105-8-209

[47] OVERBEEK, Ross, 2005. The Subsystems Approach to Genome Annotation and its Use in the Project to Annotate 1000 Genomes. *Nucleic Acids Research* [online]. **33**(17), 5691-5702. ISSN 0305-1048. Available at: https://doi.org/10.1093/nar/gki866

[48] BOECKMANN, Brigitte. The SWISS-PROT protein knowledgebase and its supplement TrEMBL in 2003. *Nucleic Acids Research* [online]. **31**(1), 365-370. ISSN 13624962. Available at: https://doi.org/10.1093/nar/gkg095

[49] BENSON, Dennis A., Mark CAVANAUGH, Karen CLARK, Ilene KARSCH-MIZRACHI, David J. LIPMAN, James OSTELL and Eric W. SAYERS, 2012. GenBank. *Nucleic Acids Research* [online]. **41**(D1), D36-D42. ISSN 0305-1048. Available at: https://doi.org/10.1093/nar/gks1195

[50] CHEN, I-Min A., Ken CHU, Krishnaveni PALANIAPPAN, et al., 2021. The IMG/M data management and analysis system v.6.0: new tools and advanced capabilities. *Nucleic Acids Research* [online]. **49**(D1), D751-D763. ISSN 0305-1048. Available at: https://doi.org/10.1093/nar/gkaa939

[51] MAO, Xizeng, Tao CAI, John G. OLYARCHUK and Liping WEI, 2005. Automated genome annotation and pathway identification using the KEGG Orthology (KO) as a controlled vocabulary. *Bioinformatics* [online]. **21**(19), 3787-3793. ISSN 1367-4803. Available at: https://doi.org/10.1093/bioinformatics/bti430

[52] MEYER, Folker, Ross OVERBEEK a Alex RODRIGUEZ, 2009. FIGfams: yet another set of protein families. *Nucleic Acids Research* [online]. **37**(20), 6643-6654. ISSN 1362-4962. Available at: https://doi.org/10.1093/nar/gkp698

[53] OVERBEEK, Ross, Robert OLSON, Gordon D. PUSCH, et al., 2013. The SEED and the Rapid Annotation of microbial genomes using Subsystems Technology (RAST). *Nucleic Acids Research* [online]. **42**(D1), D206-D214. ISSN 0305-1048. Available at: https://doi.org/10.1093/nar/gkt1226

[54] DELCHER, Arthur L., Kirsten A. BRATKE, Edwin C. POWERS and Steven L. SALZBERG, 2007. Identifying bacterial genes and endosymbiont DNA with Glimmer. *Bioinformatics* [online]. **23**(6), 673-679. ISSN 1460-2059. Available at: https://doi.org/10.1093/bioinformatics/btm009

[55] BRETTIN, Thomas, James J. DAVIS, Terry DISZ, et al., 2015. RASTtk: A modular and extensible implementation of the RAST algorithm for building custom annotation pipelines and annotating batches of genomes. *Scientific Reports* [online]. **5**(1). ISSN 2045-2322. Available at: https://doi.org/10.1038/srep08365

[56] KENT, W. James, 2002. BLAT -The BLAST -Like Alignment Tool. *Genome Research* [online]. **12**(4), 656-664. ISSN 1088-9051. Available at: https://doi.org/10.1101/gr.229202

[57] HUERTA-CEPAS, Jaime, Damian SZKLARCZYK, Davide HELLER, et al., 2019. EggNOG 5.0: a hierarchical, functionally and phylogenetically annotated orthology resource based on 5090 organisms and 2502 viruses. *Nucleic Acids Research* [online]. **47**(D1), D309-D314. ISSN 0305-1048. Available at: https://doi.org/10.1093/nar/gky1085

[58] JENSEN, Lars Juhl, Philippe JULIEN, Michael KUHN, Christian von MERING, Jean MULLER, Tobias DOERKS and Peer BORK, 2007. EggNOG: automated construction and annotation of orthologous groups of genes. *Nucleic Acids Research* [online]. **36**(Database), D250-D254. ISSN 0305-1048. Available at: https://doi.org/10.1093/nar/gkm796

[59] CHEN, Feng, 2006. OrthoMCL-DB: querying a comprehensive multi-species collection of ortholog groups. *Nucleic Acids Research* [online]. **34**(90001), D363-D368. ISSN 0305-1048. Available at: https://doi.org/10.1093/nar/gkj123

[60] UCHIYAMA, Ikuo, Motohiro MIHARA, Hiroyo NISHIDE, Hirokazu CHIBA and Masaki KATO, 2019. MBGD update 2018: microbial genome database based on hierarchical orthology relations covering closely related and distantly related comparisons. *Nucleic Acids Research* [online]. **47**(D1), D382-D389. ISSN 0305-1048. Available at: https://doi.org/10.1093/nar/gky1054

[61] KRIVENTSEVA, Evgenia V., Fredrik TEGENFELDT, Tom J. PETTY, Robert M. WATERHOUSE, Felipe A. SIMÃO, Igor A. POZDNYAKOV, Panagiotis IOANNIDIS and Evgeny M. ZDOBNOV, 2015. OrthoDB v8: update of the hierarchical catalog of orthologs and the underlying free software. *Nucleic Acids Research* [online]. **43**(D1), D250-D256. ISSN 1362-4962. Available at: https://doi.org/10.1093/nar/gku1220

[62] GALPERIN, Michael Y., Yuri I. WOLF, Kira S. MAKAROVA, Roberto VERA ALVAREZ, David LANDSMAN and Eugene V. KOONIN, 2021. COG database update: focus on microbial diversity, model organisms, and

widespread pathogens. *Nucleic Acids Research* [online]. **49**(D1), D274-D281. ISSN 0305-1048. Available at: https://doi.org/10.1093/nar/gkaa1018

[63] GREULE, Anja, Jeanette E. STOK, James J. DE VOSS and Max J. CRYLE, 2018. Unrivalled diversity: the many roles and reactions of bacterial cytochromes P450 in secondary metabolism. *Natural Product Reports* [online]. **35**(8), 757-791. ISSN 0265-0568. Available at: https://doi.org/10.1039/C7NP00063D

[64] GALPERIN, Michael Y., Kira S. MAKAROVA, Yuri I. WOLF and Eugene V. KOONIN, 2015. Expanded microbial genome coverage and improved protein family annotation in the COG database. *Nucleic Acids Research* [online]. **43**(D1), D261-D269. ISSN 1362-4962. Available at: https://doi.org/10.1093/nar/gku1223

[65] DALQUEN, Daniel A. and Christophe DESSIMOZ, 2013. Bidirectional Best Hits Miss Many Orthologs in Duplication-Rich Clades such as Plants and Animals. *Genome Biology and Evolution* [online]. **5**(10), 1800-1806. ISSN 1759-6653. Available at: https://doi.org/10.1093/gbe/evt132

[66] LI, Chun, Qi-Gang LI, Jim M. DUNWELL and Yuan-Ming ZHANG, 2012. Divergent Evolutionary Pattern of Starch Biosynthetic Pathway Genes in Grasses and Dicots. *Molecular Biology and Evolution* [online]. **29**(10), 3227-3236. ISSN 1537-1719. Available at: https://doi.org/10.1093/molbev/mss131

[67] BUCHFINK, Benjamin, Chao XIE and Daniel H HUSON, 2015. Fast and sensitive protein alignment using DIAMOND. *Nature Methods* [online]. **12**(1), 59-60. ISSN 1548-7091. Available at: https://doi.org/10.1038/nmeth.3176

[68] MARCHLER-BAUER, Aron and Stephen H. BRYANT, 2004. CD-Search: protein domain annotations on the fly. *Nucleic Acids Research* [online]. **32**(Web Server), W327-W331. ISSN 0305-1048. Available at: https://doi.org/10.1093/nar/gkh454

[69] ASHBURNER, Michael, Catherine A. BALL, Judith A. BLAKE, et al., 2000. Gene Ontology: tool for the unification of biology. *Nature Genetics* [online]. **25**(1), 25-29. ISSN 1061-4036. Available at: https://doi.org/10.1038/75556

[70] KANEHISA, Minoru, Miho FURUMICHI, Mao TANABE, Yoko SATO and Kanae MORISHIMA, 2017. KEGG: new perspectives on genomes, pathways, diseases and drugs. *Nucleic Acids Research* [online]. **45**(D1), D353-D361. ISSN 0305-1048. Available at: https://doi.org/10.1093/nar/gkw1092

[71] LETUNIC, Ivica and Peer BORK, 2018. 20 years of the SMART protein domain annotation resource. *Nucleic Acids Research* [online]. **46**(D1), D493-D496. ISSN 0305-1048. Available at: https://doi.org/10.1093/nar/gkx922

[72] HUERTA-CEPAS, Jaime, Kristoffer FORSLUND, Luis Pedro COELHO, Damian SZKLARCZYK, Lars Juhl JENSEN, Christian VON MERING and Peer BORK, 2017. Fast Genome-Wide Functional Annotation through Orthology Assignment by eggNOG-Mapper. *Molecular Biology and Evolution* [online]. **34**(8), 2115-2122. ISSN 0737-4038. Available at: https://doi.org/10.1093/molbev/msx148

[73] CANTALAPIEDRA, Carlos P, Ana HERNÁNDEZ-PLAZA, Ivica LETUNIC, Peer BORK, Jaime HUERTA-CEPAS and Koichiro TAMURA, 2021. EggNOG-mapper v2: Functional Annotation, Orthology Assignments, and Domain Prediction at the Metagenomic Scale. *Molecular Biology and Evolution* [online]. ISSN 0737-4038. Available at: https://doi.org/10.1093/molbev/msab293

[74] STEINEGGER, Martin and Johannes SÖDING, 2017. MMseqs2 enables sensitive protein sequence searching for the analysis of massive data sets. *Nature Biotechnology* [online]. **35**(11), 1026-1028. ISSN 1087-0156. Available at: https://doi.org/10.1038/nbt.3988

[75] TABOADA, Blanca, Karel ESTRADA, Ricardo CIRIA, Enrique MERINO and John HANCOCK, 2018. Operon-mapper: a web server for precise operon identification in bacterial and archaeal genomes. *Bioinformatics* [online]. **34**(23), 4118-4120. ISSN 1367-4803. Available at: https://doi.org/10.1093/bioinformatics/bty496

[76] SZKLARCZYK, Damian, Annika L. GABLE, David LYON, et al., 2019. STRING v11: protein—protein association networks with increased coverage, supporting functional discovery in genome-wide experimental datasets. *Nucleic Acids Research* [online]. **47**(D1), D607-D613. ISSN 0305-1048. Available at: https://doi.org/10.1093/nar/gky1131

[77] CARVER, T., N. THOMSON, A. BLEASBY, M. BERRIMAN and J. PARKHILL. 2008. DNAPlotter: circular and linear interactive genome visualization. *Bioinformatics* [online]. **25**(1), 119-120. Available at: https://doi.org/10.1093/bioinformatics/btn578

[78] MEIER-KOLTHOFF, Jan P. and Markus GÖKER. 2019. TYGS is an automated high-throughput platform for state-of-the-art genome-based taxonomy. *Nature Communications* [online]. **10**(1). Available at: https://doi.org/10.1038/s41467-019-10210-3

Symbols and abbreviations

AGPS	Alkylglycerone Phosphate Synthase
Cas	CRISPR associated protein
CDS	Coding Sequence
CDD	Conserved Domain Database
COG	the Clusters of Orthologous Groups
CRISPR	Clustered Regularly Interspaced Short Palindromic Repeats
GBDP	Genome BLAST Distance Phylogeny
HMMs	Hidden Markov Models
IMG	Integrated Microbial Genomes
KEGG	Kyoto Encyclopedia of Genes and Genomes
Mb	Megabase
MBGD	Microbial Genome Database
mRNA	messenger RNA
OG	Orthologous Group
ncRNA	non-coding RNA
ORF	Open Reading Frame
PGAP	Prokaryotic Genome Annotation Pipeline
Pfam	the Protein Families Database
PSSM	a Position-Specific Scoring Matrix
RefSeq	Reference Sequence database
ROG	Remained Orthologous Groups
Rfam	the RNA Families database
tRNA	tranfer RNA
TYGS	Type Strain Genome Server

List of appendices

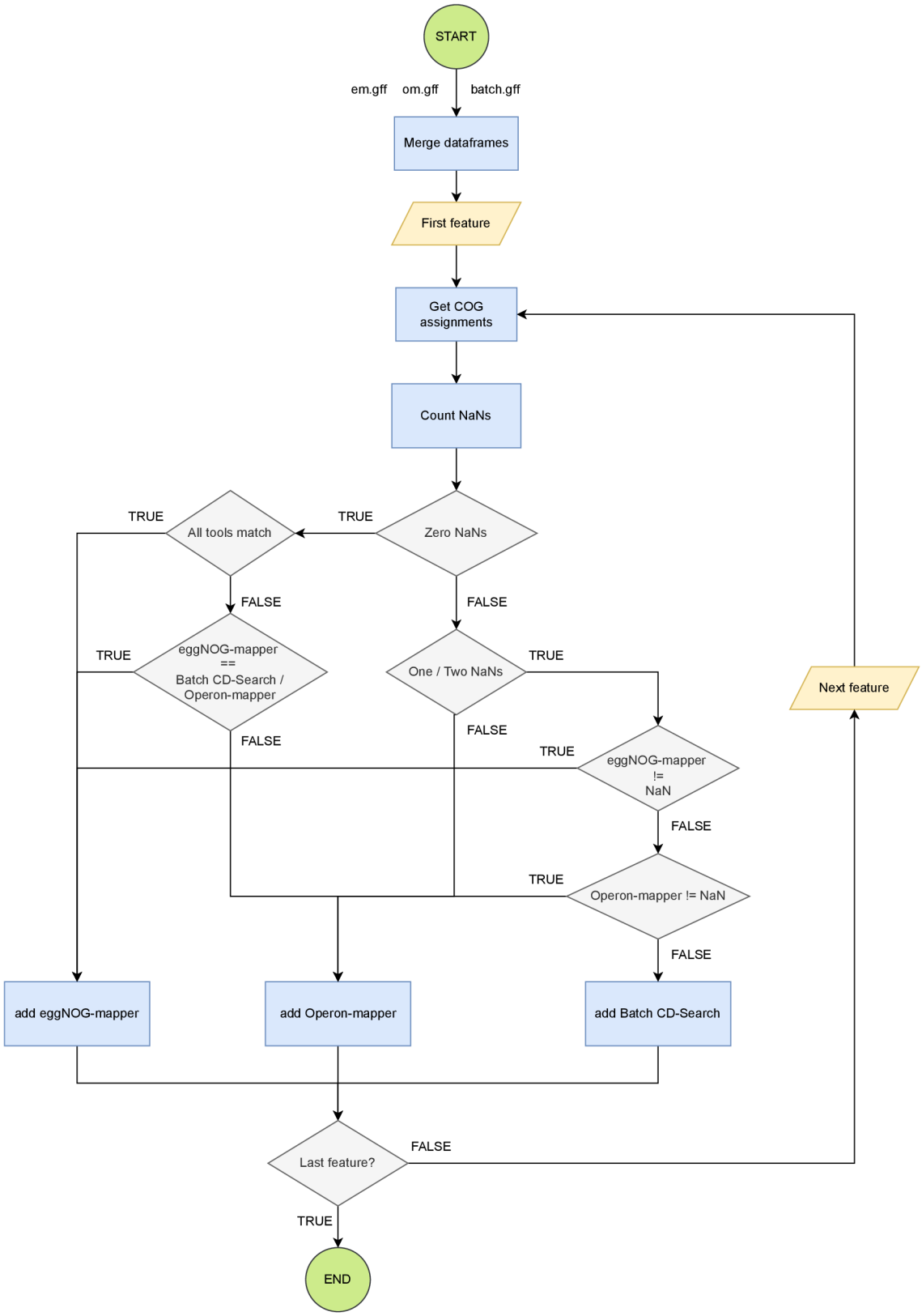
A Schematic diagram of Consensus module **61**

B Comparison of tools **62**

C Confusion matrices constructed to obtain precision and recall **63**

D List of electronic attachments **65**

A Schematic diagram of Consensus module



B Comparison of tools

		Assigned OGs							
		eggNOG-mapper		Operon-mapper		Batch CD-Search		COG-or	
Bacterium	Number of CDS	Number	%	Number	%	Number	%	Number	%
E. coli	4315	4193	97.17	3849	89.20	3479	80.63	4216	97.71
A. thermoaerophilus	3592	3113	86.66	3178	88.47	1962	54.62	3329	92.68
C. beijerinckii	5061	4823	95.30	4526	89.43	2771	54.75	4910	97.02
C. diolis	5102	4884	95.73	4581	89.79	2825	55.37	4973	97.47
R. rubrum	3850	3601	93.53	3512	91.22	2274	59.06	3690	95.84
S. thermodepolymerans	3589	3324	92.62	3345	93.20	2246	62.58	3454	96.24
T. taiwanensis	2700	2526	93.56	2571	95.22	1670	61.85	2633	97.52

C Confusion matrices constructed to obtain precision and recall

Tab. C.1: Confusion matrix for group assignment by COR-or and Batch CD-Search.

	COG-or	
Batch CD-Search	TP = 16487	FP = 824
	FN = 9609	TN = 2191

Tab. C.2: Confusion matrix for group assignment by COR-or and eggNOG-mapper.

	COG-or	
eggNOG-mapper	TP = 26553	FP = 0
	FN = 367	TN = 2191

Tab. C.3: Confusion matrix for group assignment by COR-or and Operon-mapper.

	COG-or	
Operon-mapper	TP = 20392	FP =1949
	FN = 4578	TN = 2191

Tab. C.4: Confusion matrix for category assignment by COR-or and Batch CD-Search.

	COG-or	
Batch CD-Search	TP = 16941	FP = 370
	FN = 9609	TN = 2191

Tab. C.5: Confusion matrix for category assignment by COR-or and eggNOG-mapper.

	COG-or	
eggNOG-mapper	TP = 26553	FP = 0
	FN = 367	TN = 2191

Tab. C.6: Confusion matrix for category assignment by COR-or and Operon-mapper.

	COG-or	
Operon-mapper	TP = 21263	FP =1078
	FN = 4578	TN = 2191

D List of electronic attachments

- Data for the bacterium *Aneurinibacilus thermoaerophilus* CCM 8960: inputs, outputs and visualization
- Comparison table of the COG and eggNOG databases
- Source code of the COG-or
- COG-or documentation