University of Warsaw
Faculty of Mathematics, Informatics and Mechanics

Michał Woźniak

Computational aspects of the presence of drug resistance mechanisms

PhD dissertation

Thesis supervisor
prof. dr hab. Jerzy Tiuryn
Institute of Informatics
University of Warsaw

Thesis co-advisor
prof. Limsoon Wong
School of Computing
National University of Singapore

March 2015

Author's declaration:
aware of legal responsibility I hereby declare that I have written this dissertation myself and all the contents of the dissertation have been obtained by legal means.

March 5, 2015 .....

date *Michał Woźniak*

Supervisor's declaration:
the dissertation is ready to be reviewed

March 5, 2015 .....

date *prof. dr hab. Jerzy Tiuryn*

Co-advisor's declaration:
the dissertation is ready to be reviewed

March 5, 2015 .....

date *prof. Limsoon Wong*

Acknowledgements

First, I want to thank my advisor Jerzy Tiuryn and my co-advisor Limsoon Wong for working with me on this project. They gave me guidance and freedom, setting a great example of how to approach research problems. The scientific collaboration they established enabled me to work for almost two years at the School of Computing of the National University of Singapore. I will always remain indebted for that. I consider them my true mentors who advised and helped me whenever I was in trouble.

Second, I am very grateful to all the members of the computational biology group at our department. Specifically, I want to thank Aleksander Jankowski, Mateusz Łącki, Michał Startek and Maciej Sykulski for being great colleagues and friends, willing to discuss and share their knowledge. No less grateful I am to my colleagues and friends from National University of Singapore — Narmada Sambaturu, Sucheendra Kumar and Chern Han Yong — thanks for all the discussions we had. I wouldn't have had such an exciting time in Singapore without you guys. My special thanks also go to Hufeng Zhou for involving me in his research project in which I was able to have my little contribution. I also want to thank Janusz Dutkowski with whom I worked on my first research project.

Third, I want to thank Jan Makaruk, my high school biology teacher who brought my interest to genomics, through his lessons on the subject of evolution.

I am also thankful to my friend Piotr Kuśka for reading a draft version of the manuscript and suggesting some corrections which have improved its clarity.

Many thanks to my parents who have kept supporting and encouraging me to pursue both my scientific and personal dreams.

Finally, I would like to thank my wife. Kasia, thank you for your constant trust and understanding throughout these years. I have no words to express how happy I am to expect our son Filip!

During the preparation of this thesis I was supported by the Polish Ministry of Science and Higher Education grant no. 2012/05/ST6/03247 and Singapore Ministry of Education Tier-2 grant no. MOE2009-T2-2-004.

Dedicated to Kasia and Filip.

Thesis supervisor: prof. dr hab. Jerzy Tiuryn Michał Woźniak
Thesis co-advisor: prof. Limsoon Wong

Computational aspects of the presence of drug resistance mechanisms

Abstract

The development of drug resistance in bacteria causes antibiotic therapies to be less effective and more costly. The rapidly growing number of bacterial genomes being fully sequenced and publicly available opens new possibilities for using large-scale computational approaches to improve our understanding of drug resistance mechanisms.

In the first part of this thesis we present *eCAMBer*, a tool we have developed for efficient support of comparative analysis of multiple bacterial strains within the same species. eCAMBer works in two phases. First, it transfers gene annotations among all considered bacterial strains. In this phase, it also identifies homologous gene families and annotation inconsistencies. Second, eCAMBer, tries to improve the quality of annotations by resolving the gene start inconsistencies and filtering out gene families arising from annotation errors propagated in the previous phase. We present results suggesting that eCAMBer efficiently identifies and resolves annotation inconsistencies and it outperforms other competing tools both in terms of running time and accuracy of produced annotations.

In the second part of the thesis we present *GWAMAR*, a tool we have developed for detection of drug resistance-associated mutations in bacteria through comparative analysis of whole-genome sequences. The pipeline of GWAMAR comprises several steps. First, it employs eCAMBer for a set of bacterial genomes and annotations. Second, based on the computed multiple alignments of gene families, it identifies mutations among the strains. Third, it calculates several statistics to predict which of the mutations are most likely to be associated with drug resistance. We present results of applying GWAMAR to three datasets retrieved from publicly available data for *M. tuberculosis*, and *S. aureus*.

Keywords: antibiotic resistance, bacteria, evolution, comparative genomics
ACM Classification: J.3 Biology and genetics

Contents

1 Introduction **1**

1.1 The problem of drug resistance 2

1.1.1 The discovery of teixobactin 4

1.2 Biological background 4

1.2.1 Actions of drugs and molecular mechanisms of resistance . 5

1.2.2 Fitness cost and compensatory mutations 6

1.3 Other related work 6

1.4 Problems faced and approached in this work 7

1.5 Organization of the dissertation and articles 9

2 Comparative genome annotation **11**

2.1 Introduction 12

2.1.1 Protein-coding genes in bacteria 12

2.1.2 Bacterial genome annotations 13

2.1.3 Genome annotation inconsistencies 13

2.1.4 Comparative approaches for genome annotation 14

2.1.5 Other related work 15

2.1.6 Basic concepts and notations 16

2.1.7 The problem setting 17

2.2 CAMBer: comparative analysis of multiple bacterial strains 18

2.2.1 The closure procedure 18

2.2.2 Consolidation graphs 20

2.2.3 The refinement procedure 21

2.2.4 Time complexity 24

2.3 eCAMBer: efficient support for large-scale comparative analysis of multiple bacterial strains . . . 24
2.3.1 Overview . . . 24
2.3.2 The closure procedure . . . 26
2.3.3 Sequence consolidation graph . . . 30
2.3.4 Homologous gene clusters . . . 32
2.3.5 Refinement procedure . . . 33
2.3.6 TIS voting procedure . . . 34
2.3.7 Clean up procedure . . . 35
2.3.8 Other features of eCAMBer . . . 35
2.4 CAMBerVis: visualization software to support comparative analysis of multiple bacterial strains . . . 37
2.4.1 Example usage . . . 37
2.5 Results and discussion . . . 39
2.5.1 Results for CAMBer . . . 39
2.5.2 Results for eCAMBer . . . 56
2.6 Summary . . . 66

3 Drug resistance-associated mutations 69
3.1 Introduction . . . 70
3.1.1 Genotype data . . . 70
3.1.2 Phenotype data . . . 70
3.1.3 Gold standard associations . . . 71
3.1.4 Phylogenetic information . . . 72
3.1.5 Basic definitions . . . 72
3.1.6 Problem setting . . . 75
3.2 GWAMAR: Genome-wide assessment of mutations associated with drug resistance in bacteria . . . 75
3.2.1 The pipeline of GWAMAR . . . 75
3.2.2 Association scores . . . 76
3.2.3 Time complexity . . . 90
3.3 Results and Discussion . . . 90
3.3.1 S. aureus dataset . . . 90
3.3.2 M. tuberculosis datasets . . . 102
3.4 Summary . . . 114

4 Conclusions **117**
4.1 Comparative genome annotations 118
4.2 Drug resistance-associated mutations 119

References **121**

"It is not difficult to make microbes resistant to penicillin in the laboratory by exposing them to concentrations not sufficient to kill them, and the same thing has occasionally happened in the body."

Alexander Fleming, Nobel Prize lecture, 1945

1 Introduction

Drug resistance is a broad concept used to describe situations of reduced effectiveness of a drug in curing a disease or condition. Antibiotic resistance is a form of drug resistance when some sub-population of a microorganism, typically bacteria, is capable of surviving when exposed to an antimicrobial drug. The development of drug resistance in bacteria makes antimicrobial drugs less effective and increases the cost of therapies. Since the time when first antibiotics were introduced to treat bacterial infections, due to various factors — such as irresponsible dosage of antibiotics, naturally occurring mutations, transmission of drug-resistant strains — drug resistance in pathogens has become a serious health problem. Hence, the evolution of drug resistance in bacteria is a relevant field of research in molecular biology and bioinformatics. In this thesis, we focus on computational aspects of the presence of drug resistance mechanisms in bacteria. In particular, we propose and implement a new approach, called GWAMAR, to identify mutations associated with drug resistance by comparative analysis of whole-genome sequences of multiple bacterial strains. As an important part of this work, we also propose a new approach, called eCAMBer, to support comparative analysis of multiple bacterial strains.

1.1 The problem of drug resistance

Since the first antibiotics were discovered and introduced for treatment of bacterial infections, the initial optimism has been tempered by the emergence of antibiotic resistance (Levy and Marshall, 2004). Among the 15 major classes of antibiotics introduced into treatment of bacterial infections, none has escaped the emergence of drug resistance (see Table 1.1, data adopted from Lewis (2013)).

Antibiotic class; example	Discovery	Introduction	Resistance	Mechanism of action	Activity or target species
Sulfadrugs; prontosil	1932	1936	1942	Inhibition of dihydropteroate synthetase	Gram-positive bacteria
β-lactams; penicillin	1928	1938	1945	Inhibition of cell wall biosynthesis	Broad-spectrum activity
Aminoglycosides; streptomycin	1943	1946	1946	Binding of 30S ribosomal subunit	Broad-spectrum activity
Chloramphenicols; chloramphenicol	1946	1948	1950	Binding of 50S ribosomal subunit	Broad-spectrum activity
Macrolides; erythromycin	1948	1951	1955	Binding of 50S ribosomal subunit	Broad-spectrum activity
Tetracyclines; chlortetracycline	1944	1952	1950	Binding of 30S ribosomal subunit	Broad-spectrum activity
Rifamycins; rifampicin	1957	1958	1962	Binding of RNA polymerase β-subunit	Gram-positive bacteria
Glycopeptides; vancomycin	1953	1958	1960	Inhibition of cell wall biosynthesis	Gram-positive bacteria
Quinolones; ciprofloxacin	1961	1968	1968	Inhibition of DNA synthesis	Broad-spectrum activity
Streptogramins; streptogramin B	1963	1998	1964	Binding of50S ribosomal subunit	Gram-positive bacteria
Oxazolidinones; linezolid	1955	2000	2001	Binding of 50S ribosomal subunit	Gram-positive bacteria
Lipopetides; daptomycin	1986	2003	1987	Depolarization of cell membrane	Gram-positive bacteria
Fidaxomicin	1948	2011	1977	Inhibition of RNA polymerase	Gram-positive bacteria
Diarylquinolines; bedaquiline	1997	2012	2006	Inhibition of F_1,F_O -ATPase	Narrow-spectrum activity

Table 1.1: This table, with data adopted from Lewis (2013), presents the timeline of the discovery and introduction of antibiotics. Columns (from left to right) correspond to: the name of a class of antibiotics with an example of this class, the year the antibiotics were discovered, the year the antibiotics were introduced, the year resistance to the antibiotics was observed, the brief description of the mechanism of action and finally the spectrum of the antibiotics activity.

Furthermore, drug resistance can be accumulated, giving rise to emergence of bacteria resistant to multiple antibiotics (Nikaido, 2009; Magiorakos et al., 2012).

Such bacteria are of special interest as the bacterial infections caused by them are difficult to treat (Matteelli et al., 2014; Lange et al., 2014; Trauner et al., 2014). The worldwide spread of drug-resistant bacteria has become a serious threat to public health systems and may require some global solutions (Alanis, 2005; Laxminarayan et al., 2013).

As a consequence, the problem has drawn attention of major health organizations such as WHO (World Health Organization), ECDC (European Centre for Disease Prevention and Control) and CDC (Centers for Disease Control and Prevention), which monitor and report the spreading of drug-resistant pathogens in the world (WHO, 2014; ECDC, 2012; CDC, 2013).

The recent report on antimicrobial resistance by WHO (2014) states that national data obtained for *Escherichia coli* (*E. coli*), *Klebsiella pneumoniae* (*K. pneumoniae*) and *Staphylococcus aureus* (*S. aureus*) show that in many settings more than 50% of isolated strains are resistant to commonly used antibacterial drugs.

Another recent report by WHO (2013) on *M. tuberculosis* estimates that the bacteria was responsible for around 1.3 million deaths worldwide in 2013. According to the report, 3.6% of new cases of tuberculosis in 2012 were multi-drug resistant (MDR-TB). The report estimates that only around 48% of patients from a cohort from 2010 with MDR-TB were successfully treated. Moreover, 92 countries reported cases of extensively-drug-resistant (XDR-TB) isolates of the bacteria. According to WHO (2014), MDR-TB is defined as resistance to the first-line drugs: isoniazid and rifampicin; whereas XDR-TB is defined as MDR-TB plus resistance to at least a fluoroquinolone and one second-line injectable agent (amikacin, kanamycin or capreomycin).

Moreover, recently new forms of totally-drug resistant *M. tuberculosis* (TDR-TB) have been discovered in Iran (Velayati et al., 2009), and in India (Udwadia et al., 2012). Also, new findings which have been recently published by Klopper et al. (2013) suggest emergence of TDR-TB strains in South Africa. The TDR-TB strains are characterized by a very broad spectrum of resistance, which makes infections caused by TDR-TB virtually untreatable (Velayati et al., 2013). However, there is no consensus on the precise definition of TDR-TB (Cegielski et al., 2012).

The problem of drug resistance is also appalling, as we observe a slowing down in the last twenty years of the pace of discovering new drugs against bacteria

(Lewis, 2013). Moreover, it has been argued that it may not be economically justifiable for pharmaceutical companies to develop new antibiotics (Alanis, 2005; Shlaes et al., 2004). For example, Projan (2003) argues that, based on the cost and complexity of drug discovery and development, investment into long-term treatment of chronic diseases is more attractive than investment into short-course therapies.

1.1.1 The discovery of teixobactin

Recently, Ling et al. (2015) announced in *Nature* their discovery of a new antibiotic, they called teixobactin. The researchers used a new approach to screen for substances which are naturally produced by bacteria and could also be active against other bacteria. In the device they developed, called iChip, colonies of different bacteria are cultured together with the target bacteria (*S. aureus*) in different chambers. Then, the chambers with inhibited growth of the target bacteria contain antibiotics naturally produced by the cultured bacteria. It turned out, that teixobactin is produced by a newly discovered bacteria called *Eleftheria terrae*. Notably, employing their approach, the researchers identified 25 antimicrobial drugs, out of which teixobactin was the most active. The drug turned out to be active both against *S. aureus* and *M. tuberculosis*. Moreover, no resistance to the drug was observed in the experiments. The authors said that clinical trials on people could start within two years.

1.2 Biological background

Different aspects of drug resistance in bacteria have been studied intensively for a few decades now (Levy and Marshall, 2004; Davies and Davies, 2010). In general, drug resistance emerges as a result of evolution which adapts bacteria to the environment with antibiotics. Some bacteria, even within wild-type communities, may have naturally increased level of drug resistance (Turnidge et al., 2006). Exposure to a drug selects these bacteria with the increased level of ability for survival in the environment with this drug. Those bacteria will produce a generation which will inherit the higher level of drug resistance. In fact, this process is so fast, that it can be reproduced and traced in a laboratory (Zhang et al., 2011) — providing a perfect example of Darwinian principles of evolution

by natural selection (Sykes, 2010).

It should be noted, however, that the concept of drug resistance may also be applied in the context of bacteria which are naturally resistant to a specific drug. This type of resistance is called *intrinsic*, as opposed to acquired resistance. Several examples of intrinsic resistance can be found in the review article on intrinsic resistance by Cox and Wright (2013).

Another phenomenon, called *phenotypic* drug resistance, describes a situation when drug resistance is not reflected by genetic variations of an individual bacteria. Examples of this phenomenon include creation of biofilms, persistence or stationary growth phase. For more details on this subject, see the review by Corona and Martinez (2013).

1.2.1 Actions of drugs and molecular mechanisms of resistance

Antimicrobial drugs bind their molecular targets inside the bacterial cell in order to disrupt some biological processes which are essential for the bacteria (Juhas et al., 2012). In order for a drug to be effective, the following three conditions should be satisfied: (i) its drug target is in the bacterial cell, (ii) the antibiotic reaches the target in sufficient quantity, and (iii) the antibiotic is not inactivated or modified by the bacteria (Džidić et al., 2008; Blair et al., 2015).

The known drug resistance mechanisms can be categorized following the review by Wright (2011): (i) drug target modification — preventing binding of a drug; (ii) efflux — reduced accumulation of the drug inside a bacteria cell by pumping out the drug; (iii) chemical modification — modification of drug molecule by specialized enzymes; and (iv) molecular bypass — alternative metabolic pathways which can substitute for the disturbed pathways used in drug-susceptible bacteria.

On the molecular level, the process of acquisition of drug resistance is typically associated with genetic changes. These changes include chromosomal point mutations and Horizontal Gene Transfer (HGT) (Džidić et al., 2008; Davies and Davies, 2010).

For example, rifampicin acts by binding to the RNA polymerase — the enzyme responsible for transcription of DNA — one of the essential processes in bacterial cell. It forms a stable complex with the β sub-unit (encoded by gene *rpoB*) of the enzyme. As a result it suppresses the initiation of the transcription process

leading to death of bacterial cell (Campbell et al., 2001). The most common mechanism of resistance to rifampicin (reported to be present in around 97% of the cases) is based on the drug target modification by point mutation in the *rpoB* gene, inside the rifampicin resistance-determining region (RRDR) (Bravo et al., 2009).

1.2.2 Fitness cost and compensatory mutations

The process of acquisition of drug resistance is often associated with some additional cost, called fitness, which reduces the general viability of the bacteria (Andersson and Hughes, 2010; Koch et al., 2014). Results presented by Smith et al. (2014) suggest also that MDR-TB and XDR-TB strains may have reduced virulence. Moreover, it has been argued, that drug resistance in bacteria can be reversed by the presence of drug susceptible bacteria which would win the competition in a drug-free environment (Levy, 2002).

On the other hand, it has been observed, that the deleterious effect of drug resistance mutations may be reversed completely or partially by secondary mutations, called *compensatory mutations* (Maisnier-Patin and Andersson, 2004; Wiesch et al., 2010; Koch et al., 2014). According to our knowledge, there is no database for compensatory mutations, and the information about them is spread out through the literature.

1.3 Other related work

A promising approach to address the problem of drug resistance has been proposed by Chong and Sullivan (2007). It is to use computational modeling to identify old drugs that were designed for treating other diseases, but could also be effective against pathogens. An effort in this direction was undertaken in a research study on *M. tuberculosis* by Kinnings et al. (2009). The authors used three-dimensional docking to identify *in silico* some putative drug-target interactions. As a result, they predicted Comtan, a drug used in treating Parkinson's disease, as potentially effective against *M. tuberculosis* infections.

Systems biology is a large and well-founded field of computational biology which focuses on global interactions within a biological system rather than its elements separately (Sauer et al., 2007). It has been argued that systems biology

approaches may be useful to deepen our understanding of the process of emergence of antibiotic resistance (Wong and Liu, 2010). For example, Raman and Chandra (2008) proposed an interesting concept of a *co-target*, defined as a gene or a protein which, when targeted simultaneously with the primary target of a drug, inhibits the pathway leading to the emergence of drug resistance. In the framework developed by the authors co-targets are identified based on computational analysis of an information flow in the protein-protein interaction (PPI) network.

Since the idea of employing systems biology approaches to bacteria is promising, it requires high-quality interaction networks. However, unlike for human or yeast, the number of PPI networks experimentally obtained for bacteria remains very small (Wong and Liu, 2010). Moreover, a comparative analysis of two PPI networks available for *M. tuberculosis* by Zhou and Wong (2011), revealed that the overlap between the available networks is unexpectedly low, raising questions on reliability of the data.

A similar phenomenon was observed for metabolic pathways, Zhou et al. (2012) compared available datasets of *M. tuberculosis* metabolic pathways from KEGG, WikiPathways, and BioCyc. It turned out, that the pairwise overlap between the data sources is only 0.3%-4%. The above results suggest that much yet has to be done to improve the reliability of interaction networks for bacteria.

Some other fields of application for computational methods in the context of drug resistance include: rational drug design (Lewis, 2013), combinatorial drug usage (Jia et al., 2009), drug-target discovery (Chung et al., 2013; Zoraghi and Reiner, 2013), and prediction of drug-target interactions (Bakheet and Doig, 2010; Felciano et al., 2013; Amir et al., 2014).

Interestingly, Lambert et al. (2011) points out multiple analogies in the process of rapid evolution of drug resistance to chemotherapy in cancer cell communities and the evolution of antibiotic resistance in bacteria. It sets perspectives for transferring some methodologies from studying drug resistance in bacteria to study drug resistance in cancer.

1.4 Problems faced and approached in this work

We address in this thesis the problem of using whole-genome sequences to identify and associate genetic changes with drug-resistance phenotypes by compara-

tive analysis of multiple closely related bacterial strains. Thus, conceptually our approach is similar to Genome-Wide Association Study (GWAS) approaches, which have been successfully applied to identify single nucleotide polymorphisms (SNPs) associated with phenotype for various human diseases (Manolio, 2010), cancer (Stadler et al., 2010), and intelligence (Davies et al., 2011).

We hypothesize that similar approaches, when applied to bacteria, should bring interesting results, enriching our knowledge on the molecular aspects of drug resistance. For example, better knowledge of mutations associated with drug resistance may help to design molecular test on drug resistance, such as *Xpert MTB/RIF* (Köser et al., 2014). This relatively cheap test allows for rapid and accurate tests for rifampicin resistance based on the presence of point mutations in *rpoB* (Boehme et al., 2010). It has become a front-line diagnostic tool in South Africa (Zumla et al., 2013). The potential of using whole-genome comparative approaches to understand bacterial drug resistance has been discussed in the recent articles by Köser et al. (2014), Hasman et al. (2014) , Lázár et al. (2014), Trauner et al. (2014) and Blair et al. (2015).

We note, however, that the methodology may require some modifications to transfer it to bacteria. For example, horizontal gene transfer (HGT) plays an important role in the development of drug resistance in bacteria (Warnes et al., 2012); thus it may be needed to focus not only at SNPs, but also at gene gain/losses.

One challenge we faced during this project was caused by inconsistent and poor-quality annotations of bacterial strains available in public databases. It has been argued in various articles that these inconsistencies are due to different annotation methodologies used by different sequencing laboratories (Overbeek et al., 2007; Dunbar et al., 2011; Chai et al., 2014). It has also been shown, that poor-quality annotations may complicate or bias the comparative analysis of bacterial strains (A Palleja et al., 2008; Cock and Whitworth, 2010; Dunbar et al., 2011; Yu et al., 2011; Wood et al., 2012). The tools we developed to tackle this problem are presented in chapter 2.

The next challenge we faced was to collect sufficient amount of data on outcomes of drug susceptibility tests for different strains. These outcomes would constitute the phenotype data for further analysis. It turned out that this data is spread throughout the literature and databases, thus not easy to gather. We made the collected information publicly available at the website of our project,

http://bioputer.mimuw.edu.pl/gwamar.

Finally, having collected the phenotype and genotype, we approach the problem of associating the identified genetic variations with the drug-resistance phenotypes. In order to achieve this, we implemented several association scores, including two new association scores, called *weighted support* (WS) and *tree-generalized hypergeometric* (TGH) score. We also propose a new association score, called *Rank-based metascore* (RBM) for combining multiple scores into one in order to compromise between different approaches used to define different scores. The newly proposed scores employ phylogenetic information to improve the prediction of genetic changes associated with drug resistance. Our approach and the tools we have developed are presented in chapter 3.

1.5 Organization of the dissertation and articles

The dissertation is organized into four chapters. Most of the results presented in the thesis have been published in peer-reviewed articles.

In the first chapter, we briefly reviewed the current background knowledge of the drug resistance mechanisms. A broader coverage of the biological background can be found in the review articles by Levy and Marshall (2004), Džidić et al. (2008), Davies and Davies (2010) and by Wright (2011).

In the second chapter, we present our work on supporting comparative analysis of multiple bacterial strains. We present and describe the methods and the software we have developed to support that analysis. The presentation is based on our three articles describing CAMBer (Woźniak et al., 2011a), CAMBerVis (Woźniak et al., 2011b), and eCAMBer (Woźniak et al., 2014b).

In the third chapter, we present our work on detection of drug resistance-associated mutations based on comparative analysis of whole-genome sequences of multiple bacterial strains. In particular, we present GWAMAR, the tool we have developed to support that analysis. The presentation is based on our two articles (Woźniak et al., 2012) and (Woźniak et al., 2014a).

In the last chapter, we summarize our results and suggest possible directions of further research.

"The major credit I think Jim and I deserve, considering how early we were in our research careers, is for selecting the right problem and sticking to it. It's true that by blundering about we stumbled on gold, but the fact remains that we were looking for gold. Both of us had decided, quite independently of each other, that the central problem in molecular biology was the chemical structure of the gene."

Francis Crick, What Mad Pursuit, 1988

2

Comparative genome annotation

In this chapter we present our approach and the tools we have developed to improve consistency and overall accuracy of genome annotations, thus supporting the comparative analysis of multiple bacterial strains. The need to address this problem appeared when we attempted to use the publicly available genomes to identify genes and mutations associated with drug resistance. It turned out that in some settings the inconsistent and poor-quality annotations of bacterial strains may bias that analysis. In section 2.1 we review other related approaches which tackle this problem. We also introduce basic concepts and notations used in this work. In section 2.2 we present CAMBer — the first tool we developed for supporting comparative analysis of multiple bacterial strains. Next, in section 2.3 we present eCAMBer, which is a highly optimized revision of CAMBer, scaling it up for significantly larger datasets comprising hundreds of bacterial strains. In section 2.4 we present CAMBerVis, a tool we have developed for visual analysis of annotation inconsistencies. Finally, in section 2.5 we discuss results obtained by applying these tools to the case-study datasets for *M. tuberculosis*, *S. aureus* and *E. coli*. The presented results show that eCAMBer is faster and produces more reliable annotations than other currently available tools.

2.1 Introduction

Due to advances in high-throughput sequencing technologies (Loman et al., 2012; Weinstock and Peacock, 2014), the number of bacterial genome sequences available in public databases is growing rapidly. One database with bacterial genome sequences is PATRIC, developed by Gillespie et al. (2011). Notably, from June 8, 2011 to February 12, 2014, the total number of whole-genome sequences available in the PATRIC database grew from 3303 to 14114, reaching 21786 in September 24, 2014. By then, there were 1653 whole-genome sequences of *E. coli* and 642 whole-genome sequences of *S. enterica* strains available in the database (Gillespie et al., 2011).

The fast-growing number of available bacterial genome sequences enable new interesting comparative analysis of multiple bacterial strains (Binnewies et al., 2006; Hiller et al., 2007; Laing et al., 2011). In particular, it opens new opportunities to use whole-genome comparative approaches to analyse drug resistance mechanisms (Hasman et al., 2014; Alam et al., 2014).

2.1.1 Protein-coding genes in bacteria

Genes are the most fundamental units of heredity in living organisms. On the molecular level gene is a fragment of the genome which is associated with regulatory regions, transcribed regions, and or other functional sequence regions (Pearson, 2006).

The most important class of genes in bacteria constitute the *protein-coding genes*, which are transcribed into mRNA and then translated into proteins. Unlike in higher order organisms, in bacteria, protein-coding genes constitute the vast majority of genes. Protein-coding genes are also densely packed inside the bacterial genome. For example, the genome of *M. tuberculosis* is 4.4M bp. long and contains around 4,000 genes, of which about 3,900 are protein-coding genes, which span around 90% of the genome (Cole et al., 1998). Similarly, the genome sequence of *E. coli* is around 4.64M bp. and contains around 4,140 protein coding-genes (of 4,500 genes), which span about 84% of the genome (Blattner et al., 1997).

2.1.2 Bacterial genome annotations

The concept of genome annotation may refer to many different aspects of attaching biological information to genome sequences, such as: identifying gene locations (Karp et al., 2007), assigning functions to genes (Richardson and Watson, 2013), and assigning network context to gene products (Kasif and Steffen, 2010).

In this work, we focus on identifying locations of protein-coding genes. The translation unit of a protein-coding gene is a continuous fragment of DNA, of length divisible by 3. It begins with a *start codon* (typically ATG, but also GTG and TTG are common) at the translation initiation site (TIS), and ends with a *stop codon* (TAA, TAG or TGA). An *open-reading frame* (ORF) is any fragment of DNA which satisfies the above conditions, thus it has the potential to code for a protein. However, the presence of an ORF does not imply that the region is translated.

We use the term *gene annotation* (or *ORF annotation*) to refer to genome coordinates of the translation unit of a protein-coding gene from its TIS (alternatively called *gene start*) to the nearest stop codon (alternatively called *gene end*). Note that each ORF annotation is unambiguously determined by specifying strand and position of its start codon. Thus, we can use the term *TIS annotation* as a synonym to ORF annotation.

Typically, but not always, together with the newly published genome sequences of bacterial strains, genome annotations with the locations of genes are released. For example, in the dataset of 173 *M. tuberculosis* strains, studied in chapter 3, only 128 have their genome annotations deposited in the RefSeq database (Tatusova et al., 2014).

2.1.3 Genome annotation inconsistencies

It has been observed that there are common inconsistencies and inaccuracies in genome annotations among closely related bacterial strains. Moreover, it has also been argued that most of these inconsistencies are not reflected by sequence discrepancies, but arise as a result of different annotation methodologies applied by different laboratories (Wood et al., 2012; Richardson and Watson, 2013; Dunbar et al., 2011).

In fact, Dunbar et al. (2011) has shown that the use of the same tool to

annotate a set of bacterial genomes increases annotation consistency. However, these annotation inconsistencies among closely related genomes can even arise from annotations produced by the same annotation tool or made by the same laboratory.

The observed inconsistencies are mostly of two types: mis-identification of gene presence (false-positive and false-negative predictions are possible) and inconsistent gene starts. Some works, however, distinguish a separate class of annotation inconsistencies, caused by frameshift mutations (Angiuoli et al., 2011).

It has also been argued, that comparative analyses may be complicated or biased due to the inconsistencies in genome annotations among closely related bacterial strains (Salzberg, 2007; Dunbar et al., 2011; Yu et al., 2011). This includes identification of overlapping genes (A Palleja et al., 2008; Cock and Whitworth, 2010), estimation of core-genome size (Ali, 2013), and gene functional annotation (Chai et al., 2014).

Interestingly, the presence of annotation inconsistencies is an expected phenomenon when single-genome prediction tools are applied independently. For example, suppose we annotate independently $k = 20$ genomes, and assume that the missing gene error rate is $\epsilon = 0.035$, which is the corresponding Prodigal (Hyatt et al., 2010) error rate estimated on the *E. coli* dataset. Then, since $1 - (1 - \epsilon)^k = 0.51$, about 51% of core gene families would have at least one missing gene annotation.

2.1.4 Comparative approaches for genome annotation

It has been proposed by Poptsova and Gogarten (2010) that the accuracy of single-genome annotation tools can be improved by comparative annotation among multiple genomes. However, even though there are many annotation tools dedicated to a single-genome, there are relatively few tools supporting comparative annotation and analysis of multiple bacterial genomes. Hence, there is a need to develop more tools to improve consistency of genome annotations across multiple bacterial strains (Poptsova and Gogarten, 2010).

Mugsy-Annotator, developed by Angiuoli et al. (2011), is a tool which helps in the curation of annotations of multiple bacterial genomes by identifying annotation inconsistencies. First, this tool computes whole-genome multiple alignment by employing Mugsy (Angiuoli and Salzberg, 2011). Then, based on annotated

gene coordinates mapped on genomes in the multiple-genome alignment, Mugsy-Annotator identifies orthologous gene families and annotation inconsistencies, and proposes changes to the input annotations.

Two new majority voting-like approaches have been recently proposed to improve annotation accuracy and consistency among multiple genomes: ORFcor, developed by Klassen and Currie (2013) and GMV, developed by Wall et al. (2011). However, these tools approach only on the problem of inconsistent TIS annotations.

Another notable tool which employs the idea of comparing gene annotations among closely related genomes is GenePRIMP, developed by Pati et al. (2010). This tool identifies and reports gene annotation anomalies based on protein BLAST queries run against the NCBI nr database. These reports are helpful for manual curation of genome annotations. A similar feature has also been implemented in CAMBerVis (Woźniak et al., 2011b) — the tool we have developed for visualization and analysis of annotation inconsistencies. The importance of software to support manual curation of bacterial genome annotations have also been discussed by Salzberg (2007), who proposes a wiki-based solution.

Finally, a promising idea to improve annotation accuracy and consistency — by combining outputs of several single-genome annotation tools — has been explored with a few proposed approaches by Pavlović et al. (2002), Yada et al. (2003), Shah et al. (2003) and Ederveen et al. (2013). However, these meta-approaches can be viewed as single-genome annotation tools.

2.1.5 Other related work

An idea, called compressive genomics, has recently been proposed with new approaches to optimize BLAST search time of sequence databases (Loh et al., 2012; Daniels et al., 2013). However, one significant conceptual difference, between these methods and the closure procedure in eCAMBer, is that these approaches try to reduce the size of the target database, whereas the eCAMBer closure procedure reduces the redundancy among BLAST queries. It may be interesting, for further research, to combine these ideas.

Since the most time-consuming operations for CAMBer and eCAMBer are BLAST queries during the closure procedure for transferring the gene annotations, thus it is worth considering to replace BLAST with one of the recently

published tools with better running times. One such tool is DIAMOND, developed by Buchfink et al. (2015). The authors present results suggesting their tool achieves better accuracy than BLAST. Moreover, in some settings, it performs up to 20,000 times faster.

2.1.6 Basic concepts and notations

In this chapter we assume that we have a set of closely related bacterial strains, typically within the same species, whose genomes have been sequenced and annotated. More precisely, we consider a set of bacterial strains $\mathcal{S}$. For each strain $S \in \mathcal{S}$, we have its input *genome annotation* A_S and a set of contig sequences which constitutes the genome G_S, indexed by contig identifiers. Additionally, let O_S denote the set of all ORFs (annotated and not annotated) in the genome sequence of strain S.

Each genome annotation $A_S \subseteq O_S$ is represented as a set of *gene annotations*, where each gene annotation $x \in A_S$ is represented as a tuple:

$$(\text{length}, \text{end}, \text{strand}, \text{contig}). \tag{2.1}$$

Here:

- length - a number which corresponds to the length of the nucleotide sequence (including the start and stop codons) of the gene;
- end - a number which corresponds to the position of the last character of the nucleotide sequence of the gene within the contig;
- strand - location of the gene, on the positive ('+') or opposite strand ('-');
- contig - identifier of the contig with the gene sequence.

Since we consider only protein-coding gene annotations we assume additionally that length is divisible by 3.

Based on this notation, the location of the gene start corresponding to gene annotation $x \in A_S$ may be calculated using the following formula:

$$\text{start} = \begin{cases} \text{end} - \text{length} + 1 & if\ \text{strand} = \text{'+'} \\ \text{end} + \text{length} - 1 & if\ \text{strand} = \text{'-'} \end{cases} \tag{2.2}$$

Due to gene annotation inconsistencies, multiple ORF annotations — corresponding to the same gene in different strains — may suggest different lengths of the gene. In order to account for this situation, we generalize the concept of a gene annotation. We introduce a term we called a *multigene annotation*. Each multigene annotation is represented as a tuple:

$$(\text{set of ORF lengths}, \text{end}, \text{strand}, \text{contig}). \tag{2.3}$$

Here:

- set of ORF lengths - a set of numbers, representing different different ORF lengths which correspond to different gene starts of the multigene;
- end - a number which corresponds to the position of the last character of the nucleotide sequence of the gene within the contig;
- strand - location of the gene, on the positive ('+') or opposite strand ('-');
- contig - identifier of the contig with the gene sequence.

Gene annotations corresponding to different ORF lengths in a multigene annotations are called elements of the multigene. Obviously gene annotations can be viewed as multigene annotations with just one element.

Additionally, we introduce the following auxiliary functions: (i) *elts*, for a given multigene, returns the set of ORF annotations corresponding to the multigene (sharing the same stop codon); (ii) *mults*, for a given set of ORF annotations, returns the smallest set of its corresponding multigene annotations; (iii) function *seq*, for a given ORF annotation, returns its corresponding nucleotide sequence.

It should also be noted that, there is a gene identifier associated with each gene annotation. If available, for some gene annotations, additional information about its gene name is attached.

2.1.7 The problem setting

The goal is to arrive at revised genome annotations which arise from comparison and consolidation of annotations among the considered strains.

2.2 CAMBer: comparative analysis of multiple bacterial strains

Here we present in detail the methodology of CAMBer (Woźniak et al., 2011a), which is the first tool we have developed to support comparative analysis of multiple bacterial strains.

2.2.1 The closure procedure

The first and the key procedure of CAMBer is called *the closure procedure*. In this step, starting from the input genome annotations, CAMBer iteratively transfers gene annotations between the considered strains, using the BLAST software for sequence similarity searching (Camacho et al., 2009).

Let us assume CAMBer tries to transfer an ORF annotation $x \in A_S$ which corresponds to a gene in S on strain T. Then, it runs BLAST with the sequence of x, denoted $q = \text{seq}(x)$, as the query against the genome sequence of strain $T \in \mathcal{S}$.

Let y' be a hit in T returned by BLAST to the query q and let y be the ORF annotation obtained from y' by extending it to the nearest in-frame stop codon (in the 3' direction on the same strand as y'). We call the BLAST hit extension y an *acceptable hit* with respect to the query sequence q, if the following five conditions are satisfied:

- the BLAST hit y' corresponds to one of the appropriate start codons: ATG, GTG, TTG;
- the BLAST hit y' has its beginning aligned with the beginning of the query sequence q;
- the e-value score of the BLAST hit from q to y' is below a given threshold e_t (typically it is set to 10^{-10} or 10^{-20});
- the ratio of the length of y to q is less than $1 + p_t$ and greater than $1 - p_t$, where p_t is a given threshold (typically 0.2 or 0.3). This condition is imposed in order to keep similar lengths of related sequences;
- the percent of identity of the hit (calculated as the number of identities divided by the query length times 100) is above a length-dependent threshold

given by the HSSP curve (Rost, 1999). The curve was originally designed for amino-acid queries, in our case we use the formula:

$$i_t(L) = \begin{cases} 100 & L \leq 11 \\ n_t + 480 \cdot L^{-0.32 \cdot (1+e^{-L/1000})} & 11 < L \leq 450 \\ n_t + 19.5 & L > 450 \end{cases} \tag{2.4}$$

where L is the floor of the number of aligned nucleotide residues divided by 3. Typically n_t is set to 30.5% or 50.5%.

Let $q \to y$ denote that there is an acceptable hit y from the query sequence q.

Now, assume that at step $i \geq 0$, for each strain $S \in \mathcal{S}$, we have annotation A_S^i associated with that strain. Let N_S^i denote the set of gene annotations which will be used as queries in the next iteration, initially $N_S^0 = A_S^0$, for every $S \in \mathcal{S}$. Then, for every target strain $T \in \mathcal{S}$, the annotation A_T^{i+1} in the step $i+1$ is obtained by taking all extensions of acceptable hits in T for the queries ranging over all genes annotated in N_S^i, for every other strain S. All new ORF annotations extend N_S^{i+1} — the set of queries for the next iteration. This process stops when no new acceptable hit is obtained.

Thus, for every strain $T \in \mathcal{S}$, the genome annotations A_T^i in the subsequent iterations of the closure procedure in CAMBer, are defined as follows:

$$\begin{cases} N_T^0 = A_T^0 \\ H_{ST}^i = \{y \in O_T : \exists_{x \in N_S^i} \mathrm{seq}(x) \to y\} \\ A_T^{i+1} = A_T^i \cup \bigcup_{\substack{S \in \mathcal{S} \\ S \neq T}} H_{ST}^i \\ N_T^{i+1} = A_T^{i+1} \setminus A_T^i \end{cases} \tag{2.5}$$

The pseudocode 1 gives a more detailed view on the algorithm of computing the closure procedure in eCAMBer. The procedure will terminate, since the total number of ORFs among all the strains is finite.

For each strain $S \in \mathcal{S}$, we denote by A_S^c the set of gene annotations resulting from the closure procedure. Similarly, for each strain $S \in \mathcal{S}$, we denote by M_S^c the set of multigenes resulting from the closure procedure.

Assuming that we use BLAST with default parameters, our method has three specific parameters defining conditions for an acceptable hit: e-value threshold e_t,

Algorithm 1 The closure procedure in CAMBer (pseudocode)

Require: A set $\mathcal{S}$ of bacterial strains; and for each $S \in \mathcal{S}$, a set A_S^0 of genome annotations, a set G_S of sequences constituting the genome of S. The mapping function $\text{sequences}_S(A)$ returns the set of sequences in the genome G_S corresponding to the set of annotations A.

$N_S^0 \leftarrow A_S^0$ (for all $S \in \mathcal{S}$)
$i \leftarrow 0$
while $\exists_S N_S^i \neq \emptyset$ **do**
  for all $T \in \mathcal{S}$ **do**
    $N_T^{i+1} \leftarrow \emptyset$
    for all $S \in \mathcal{S}$ where $\neq T$ **do**
      $H_{ST}^i \leftarrow$ acceptable hits from sequences of N_S^i on genome G_T
      $N_T^{i+1} \leftarrow N_T^{i+1} \cup H_{ST}^i \setminus A_T^i$
    end for{BLAST computations are done in parallel for each pair $S, T \in \mathcal{S}$. Here, S, T denote the source and target strains, respectively}
    $A_T^{i+1} \leftarrow A_T^i \cup N_T^{i+1}$
  end for
  $i \leftarrow i + 1$
end while
return genome annotations A_S^i, for all $S \in \mathcal{S}$

length tolerance threshold p_t, and length-dependent-percent-of-identity threshold implied by n_t.

2.2.2 Consolidation graphs

Having computed the closure procedure we can construct now an *ORF consolidation graph* G_O. In this graph $G_O = (V_O, E_O)$, each node $o \in V_O$ represents an ORF annotation in A_S^c, for some $S \in \mathcal{S}$. There is an undirected edge $\{o_1, o_2\} \in E_O$ between a pair of ORF annotations, if there is an acceptable hit from the sequence of o_1 to o_2 or from the sequence of o_2 to o_1.

More formally, we define the graph G_O as follows:

$$\begin{cases} V_O = \bigcup_{S \in \mathcal{S}} A_S^c \\ E_O = \{\{o_1, o_2\} : \text{seq}(o_1) \rightarrow o_2 \vee \text{seq}(o_2) \rightarrow o_1\} \end{cases} \tag{2.6}$$

Second, we introduce the multigene consolidation graph $G_M = (V_M, E_M)$. Each node $m \in V_M$ in the graph is a multigene obtained after the closure procedure,

for some $S \in \mathcal{S}$. There is an undirected edge $\{m_1, m_2\} \in E_M$ between a pair of multigenes, if there is a pair of ORFs $o_1 \in \text{elts}(m_1)$ and $o_2 \in \text{elts}(m_2)$, such that there is an edge between them in the *ORF consolidation graph* (i.e., such that $\{o_1, o_2\} \in E_O$).

More formally, having the gene consolidation graph $G_O = (V_O, E_O)$ we can construct the multigene consolidation graph $G_M = (V_M, E_M)$ as follows:

$$\begin{cases} V_M = \bigcup_{S \in \mathcal{S}} M_S^c \\ E_M = \{\{m_1, m_2\} : \exists_{o_1 \in \text{elts}(m_1)} \exists_{o_2 \in \text{elts}(m_2)} \{o_1, o_2\} \in E_O\} \end{cases} \tag{2.7}$$

Figure 2.1 illustrates the process of computing the closure procedure, as well as a construction of the consolidation graphs.

2.2.3 The refinement procedure

We assume the connected components of a multigene consolidation graph G_M to represent multigene families with a common gene ancestor. Our next goal is refining the multigene homology relation represented by edges in G_M to obtain as many one-to-one homology classes as possible, i.e. having at most one multigene per strain in such a class. We call a connected component of G_M an *anchor* if it includes at most one multigene for every strain. One-element anchors are called *orphans*. *Non-anchors* are the components which fail to be anchors.

Multigenes in the same anchor are potentially orthologous to each other. In contrast, a non-anchor contains at least two multigenes that are potentially non-orthologous. Genomic context information has been successfully used to clarify gene relationships and improve gene function prediction (Wolf et al., 2001). So, we propose exploiting genomic context information to analyse and decompose non-anchors into smaller connected subgraphs that can emerge as anchors in the resulting refined consolidation graph.

The refinement procedure proceeds in stages. At each stage we carry a graph which is a subgraph of the graph from the previous stage. At stage 0, the original multigene consolidation graph G_M is used as the initial input graph G_M^0.

Suppose we have at stage i a graph G_M^i. We restrict this graph by performing the following test on each pair (m, m') of multigenes originating from strains S and T, connected by an edge in G_M^i which belongs to a non-anchor component

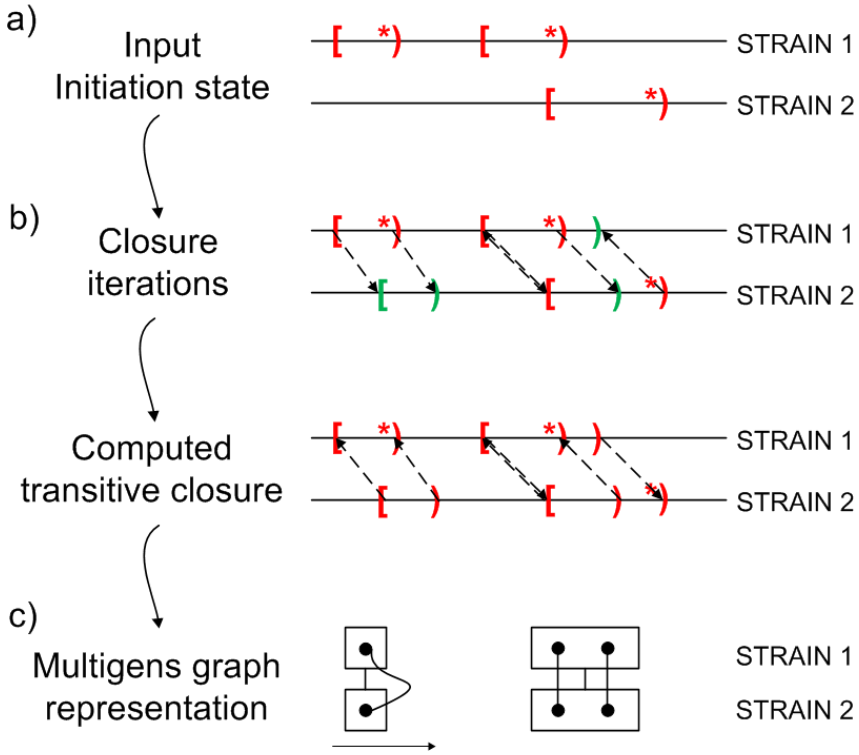

Figure 2.1: Schema of the closure procedure in CAMBer and the method to represent its outcome, including construction of the gene and multigene consolidation graphs. For clarity of the presentation only one step of the procedure is shown, for the same reason we restrict the presentation to only two strains. Square brackets indicate positions of gene ends of gene annotations (input and predicted), while round brackets with a star indicate positions of gene starts for input gene annotations. Round brackets without a star indicate positions of gene starts corresponding to predicted ORF annotations (new elements of the multigene) added during the closure procedure. **a)** Input annotations for strains indicate the initial state of the procedure. **b)** Dashed arrows indicate acceptable hits. The reader should notice a birth of a second element, rendering a multigene with two elements. **c)** Two examples of edges in the consolidation graph. Dots represent different elements of a multigene which is represented here as a rectangle. Edges of the ORF consolidation graph (connecting dots) represent acceptable hits (after ignoring their directions). Edges between rectangles represent edges of the multigene consolidation graph.

of G^i. Let a be the nearest left neighbor multigene of m in S which belongs to an anchor of G^i_M containing a multigene from T. Let b be the nearest right neighbor multigene of m in S which belongs to an anchor of G^i_M containing a multigene from T. Analogously we define left (a') and right (b') neighbors of m' in T. Assuming that all four multigenes a, a', b, b' exist, we keep the edge connecting m and m' in G^{i+1}_M if either (a, a') and (b, b') (see Figure 2.2 (a)), or (a, b') and (b, a') (see Figure 2.2 (b)) are edges in G^i_M, i.e. the corresponding pairs are in the

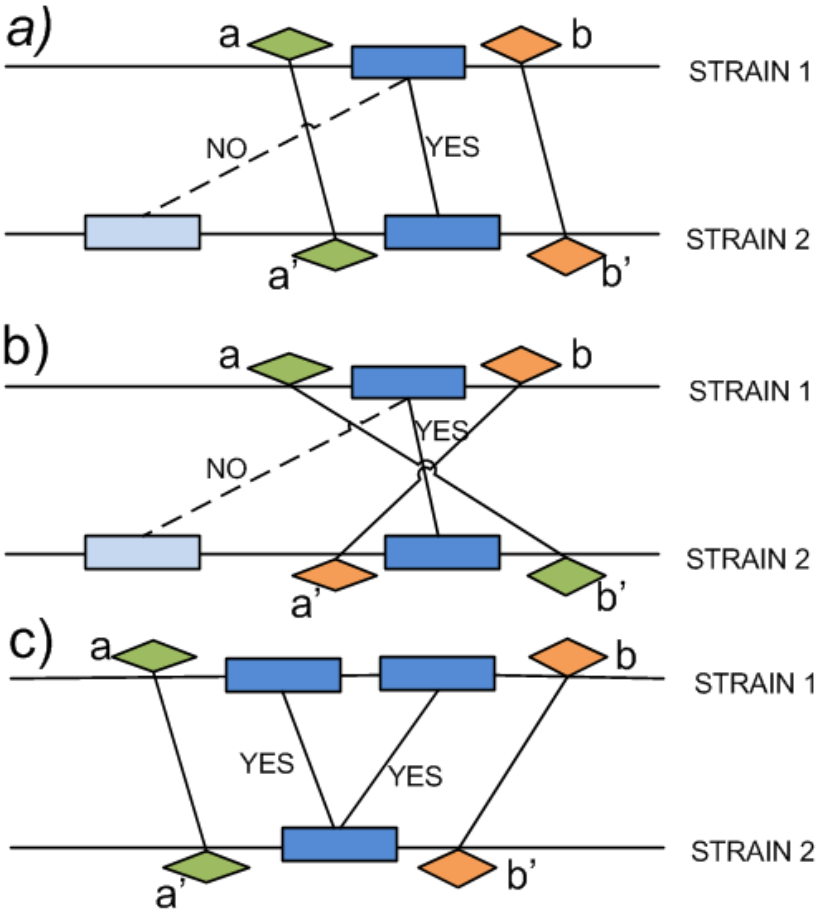

Figure 2.2: Schematic view of one step of the refinement procedure in CAMBer. Rectangles denote multigenes which belong to non-anchors in the current stage. Rhombus denotes a multigene which is already in an anchor at this stage. Edges connecting rectangles (dashed and solid) are edges of the graph of the current stage. Edges connecting rhombuses are the anchor edges. "YES" means that the edge is keep for the next stage, while "NO" means it is omitted. Parts **a)** and **b)** illustrate two situations when we can select one of the edges and leaving out the other. Part **c)** illustrates the situation when we cannot make such a decision, leading to an unresolved cluster. Both edges are kept in the graph of the next stage. Such a cluster may be resolved at a later stage. Other cases which lead to omitting the edges are possible too.

same anchors of G_M^i. If at least one of the multigenes a, a', b, b' does not exist, the edge connecting m and m' in G_M^{i+1} is not copied from G_M^i . The procedure stops when no edge is removed from the current graph. We call the resulting graph a refinement of G_M. Figure 2.2 (c) shows a situation when we have to retain two edges, leading to a cluster with unresolved one-to-one relationship. These cases may get resolved later when more anchors are obtained.

2.2.4 Time complexity

The most time-consuming operation in the closure procedure is running BLAST. We denote by blast() the BLAST running time. We note that the BLAST running time may vary due to various factors, such as the query sequence length, or the target genome size. However, since we work with genomes of closely related strains the impact of those factors is limited. Thus, we assume it to be constant.

Then, let k be the number of all considered strains and let n be the maximal number of gene annotations among all the strains. For each strain during the closure operation we use every identified or annotated ORF only once. Then, the running time of one iteration of the closure procedure is $k^2 \cdot n \cdot \text{blast}()$.

Now, we estimate time complexity of one iteration in the refinement procedure. Again, let k be the number of all considered strains and let n be the maximal number of identified multigenes among all strains. Denote by m the number of non-anchors in the consolidation graph. Additionally, let p denote the maximal number of multigenes for one strain among all non-anchor components. In order to find the nearest left and right neighbors of a multigene in linear time we first sort all of them. This takes time $k * n * \log n$. Since we have at most $p^2 \cdot \binom{k}{2}$ edges to check for support of the neighboring anchors (checking for support may take time at most n), for each of the m non-anchors, it follows that the estimated total time to resolve all of the m non-anchors is $k \cdot n \cdot \log n + m \cdot p^2 \cdot \binom{k}{2} \cdot n$.

2.3 eCAMBer: efficient support for large-scale comparative analysis of multiple bacterial strains

Here we present details on a new version of CAMBer, which we call *eCAMBer* (efficient CAMBer). It also aims to identify annotation inconsistencies and orthologous gene families. However, unlike other tools available, it has significantly better running time by taking advantage of working with highly similar genome sequences.

2.3.1 Overview

As its input, eCAMBer, requires a set of genome sequences and annotations for multiple bacterial genomes. It should be noted, however, that eCAMBer supports automatic download of bacterial annotations from the PATRIC (Gillespie

et al., 2011) database and, as an option, it allows the use of Prodigal to generate the input annotations. It works in two phases. In the first phase it uses BLAST+ (Camacho et al., 2009) to transfer each gene annotation among multiple strains. Based on the results of this procedure, homologous multigene clusters are identified. In the second phase eCAMBer applies subsequently the procedures for refinement, TIS voting and clean up. Figure 2.3 presents a schematic view of these subsequent procedures of eCAMBer.

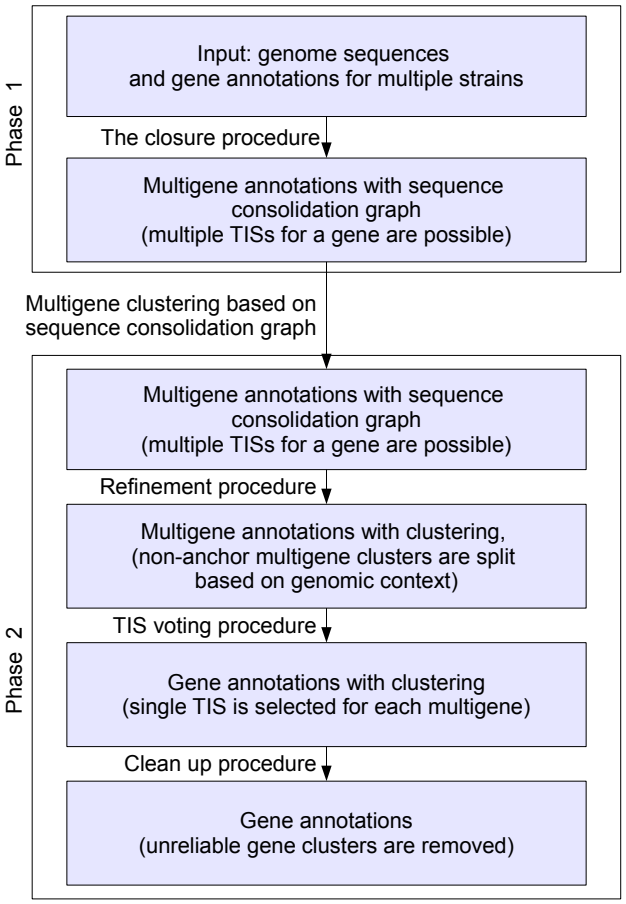

Figure 2.3: Schematic view of subsequent procedures in eCAMBer. Boxes of the chart represent the subsequent sets of genome annotations. Edges indicate application of eCAMBer procedures to process these annotations. We call a set of ORF annotations, multigene annotations, if multiple ORF annotations may share the same stop codon, indicating possible starts of translation (TISs). We use a notion of a multigene to represent multiple ORF annotations sharing the same stop codon.

The main improvements in eCAMBer, as compared to CAMBer (Woźniak et al., 2011a), are:

- significant speed up of the *closure procedure* for unifying genome annotations among bacterial strains;
- modified *refinement procedure* for splitting homologous gene families into orthologous gene clusters;
- new *TIS voting procedure* for selecting the most reliable TIS;
- new *clean up procedure* for removal of gene clusters that are likely to be gene annotation errors propagated during the closure procedure.

Here, we describe the details of the above listed procedures. The default values for parameters introduced below were chosen arbitrarily. However, based on our experiments, the program is robust for other choices of the parameters from a reasonable spectrum. eCAMBer allows users to specify values of all the parameters.

2.3.2 The closure procedure

The closure procedure is the first step of eCAMBer. The input consists of genome sequences and genome annotations for a set of closely related bacterial strains. In this procedure gene annotations are iteratively transferred among the set of considered strains, until no new ORF (open reading frames) annotations are identified.

Similarly as in CAMBer, let y' be a hit in T returned by BLAST to the query q and let y be the ORF annotation obtained from y' by extending it to the nearest in-frame stop codon (in the 3' direction on the same strand as y'). We call the BLAST hit extension y an *acceptable hit* with respect to the query sequence q, if the following five conditions are satisfied:

- the BLAST hit y' corresponds to one of the appropriate start codons: ATG, GTG, TTG, or the same start codon as in the query sequence;
- the BLAST hit y' has its beginning aligned with the beginning of the query sequence;
- the BLAST e-value score is below a given threshold e_t (in the default setting $e_t = 10^{-10}$);

- the ratio of the length of the extended hit to the query length is less than $1 + p_t$ and greater than $1 - p_t$, where p_t is a given threshold (in the default setting $p_t = 0.2$);
- the percentage of identity of the hit (calculated as the number of identities divided by the query sequence length, times 100) is above a length-dependent threshold given by the adaptation of the HSSP curve introduced in our previous work (Woźniak et al., 2011a), defined by the parameter n_t (in the default setting $n_t = 60.5$).

Algorithm 2 The closure procedure in eCAMBer (pseudocode)

Require: A set $\mathcal{S}$ of bacterial strains; and for each $S \in \mathcal{S}$, a set A_S^0 of genome annotations, a set G_S of sequences constituting the genome of S. The mapping function $\text{sequences}_S(A)$ returns the set of sequences in the genome G_S corresponding to the set of annotations A.

$Q^0 \leftarrow \bigcup_{S \in \mathcal{S}} \text{sequences}_S(A_S^0), \quad \overline{Q}^0 \leftarrow \emptyset$
$i \leftarrow 0$
while $Q^i \neq \emptyset$ **do**
  for all $T \in \mathcal{S}$ **do**
    $H_T^i \leftarrow$ acceptable hits from Q^i on the target genome G_T
    $A_T^{i+1} \leftarrow A_T^i \cup H_T^i$
  end for{The above operations are done in parallel for each target genome $T \in \mathcal{S}$. Also, for a query sequence $q \in Q^i$, if its BLAST hits are available in a database of precomputed BLAST results, eCAMBer takes results from the database instead.}
  $\overline{Q}^{i+1} \leftarrow \overline{Q}^i \cup Q^i$
  $Q^{i+1} \leftarrow \bigcup_{T \in \mathcal{S}} \text{sequences}_T(H_T^i) \setminus \overline{Q}^i$
  $i \leftarrow i + 1$
end while
return genome annotations A_S^i, for all $S \in \mathcal{S}$

First, we start with the set of input annotations A_S^0, for each strain S in the set of considered strains $\mathcal{S}$. Then, in ith iteration we compute the set of BLAST queries Q^i as the set of distinct ORF sequences among all strains, which have not been used as BLAST queries yet. Let additionally $\overline{Q}^i$ denote the set of sequences used as BLAST queries before the ith annotation. Of course $\overline{Q}^0 = \emptyset$. Next, we calculate in parallel, for each target strain $T \in \mathcal{S}$, BLAST results for all sequence queries in Q^i. For each strain $T \in \mathcal{S}$, all acceptable hits H_T^i

are added to the strain annotations, defining $A_T^{i+1} \leftarrow A_T^i \cup H_T^i$. Next, the set of newly identified sequences across all genomes $\bigcup_{T\in\mathcal{S}} \text{sequences}_T(H_T^i)$ is computed, which is then used to update the set of BLAST queries for the next iteration $Q^{i+1} \leftarrow \bigcup_{T\in\mathcal{S}} \text{sequences}_T(H_T^i) \setminus \overline{Q}^i$. We also update the set of sequences already used as queries $\overline{Q}^{i+1} \leftarrow \overline{Q}^i \cup Q^i$. The procedure stops when no new ORF sequences are identified, hence $Q^i = \emptyset$.

Thus, for every strain $T \in \mathcal{S}$, the genome annotations A_T^i in the subsequent iterations of the closure procedure in CAMBer, are defined as follows:

$$\begin{cases} H_T^i = \{x \in O_T : \exists_{q\in Q^i} q \to x\} \\ A_T^{i+1} = A_T^i \cup H_T^i \\ Q^{i+1} \leftarrow \bigcup_{T\in\mathcal{S}} \text{sequences}_T(H_T^i) \setminus \overline{Q}^i \\ \overline{Q}^{i+1} \leftarrow \overline{Q}^i \cup Q^i \end{cases} \tag{2.8}$$

The pseudocode 2 presents a more detailed view on the algorithm of computing the closure procedure in eCAMBer.

Similarly, as in CAMBer, for each strain $S \in \mathcal{S}$, we denote by A_S^c the set of gene annotations produced by the closure procedure above. We further denote by A^c the set of all ORF annotations produced by the closure procedure. For each strain $S \in \mathcal{S}$, we denote by M_S^c the set of multigenes resulting from the closure procedure.

Figure 2.4 presents a schematic view of the implementation of the closure procedure in eCAMBer.

The careful reader may notice two important differences between the closure procedure in CAMBer and eCAMBer. In particular, eCAMBer uses unique ORF sequences, rather than ORF annotations, as queries against all strain genomes and, thus, does not repeat a BLAST query when the same ORF sequence corresponds to multiple ORF annotations. In contrast, firstly, CAMBer uses all ORF sequences as queries and, thus, may repeat a query BLAST several times. Secondly, CAMBer BLASTs a query against all strains' genomes except the strain from which the query is taken. The second difference may potentially lead to different outcomes generated by these two approaches.

Since BLAST computations are the most time-consuming operations in each iteration of the closure procedure, we express the time complexity of one iteration

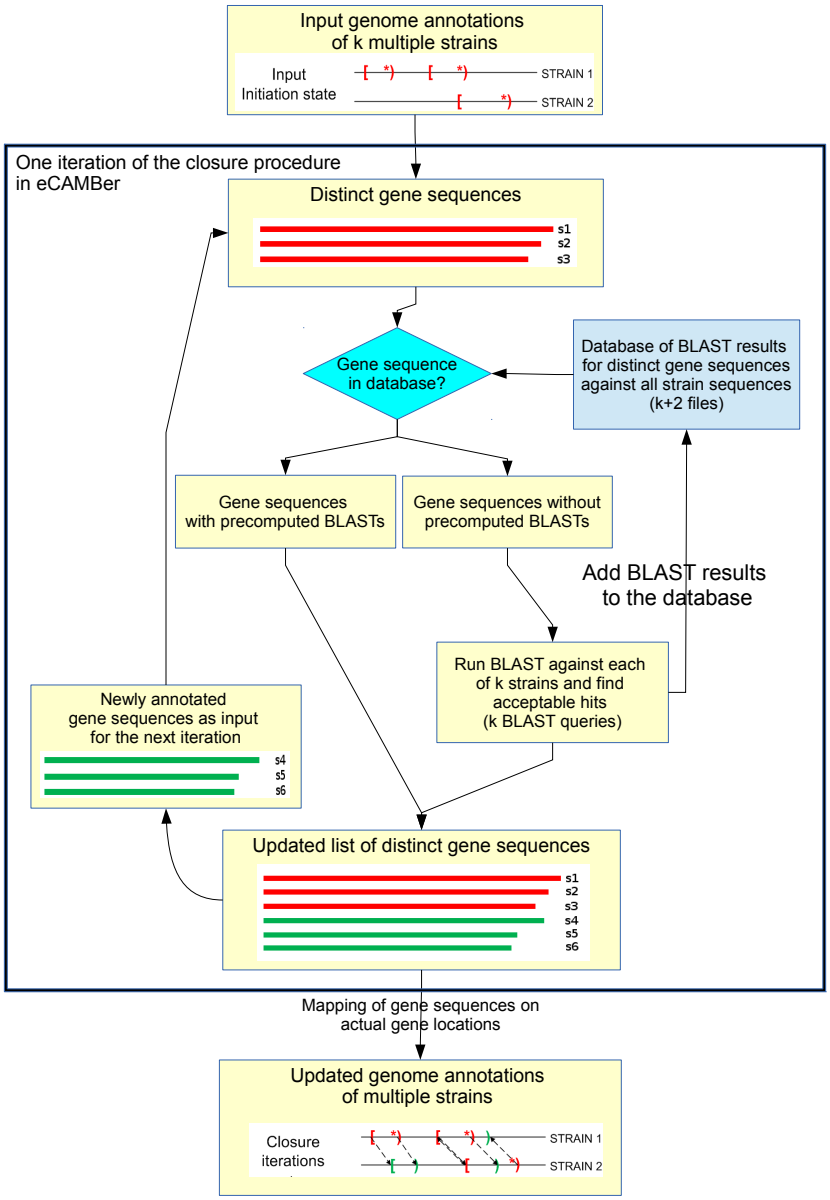

Figure 2.4: Schematic view of the closure procedure in eCAMBer. In this procedure eCAMBer, unlike CAMBer, takes advantage of working with closely related genomes. In contrast to the old approach, in each iteration, instead of using each ORF sequence as a query, it first identifies groups of ORFs with exactly identical sequences. This approach avoids use of the same ORF sequence multiple times as a BLAST query.

of the closure procedure by the number of performed BLAST computations. Let $k = |\mathcal{S}|$ denote the number of considered strains and let n be the maximal number of gene annotations per strain, in the ith iteration. Let, d denote the number of distinct gene sequences to be used as queries in the ith iteration. Then, time

complexity of one iteration of the closure procedure implemented in eCAMBer can be expressed as $O(d \cdot k)$, whereas it is $O(n \cdot k^2)$ for CAMBer.

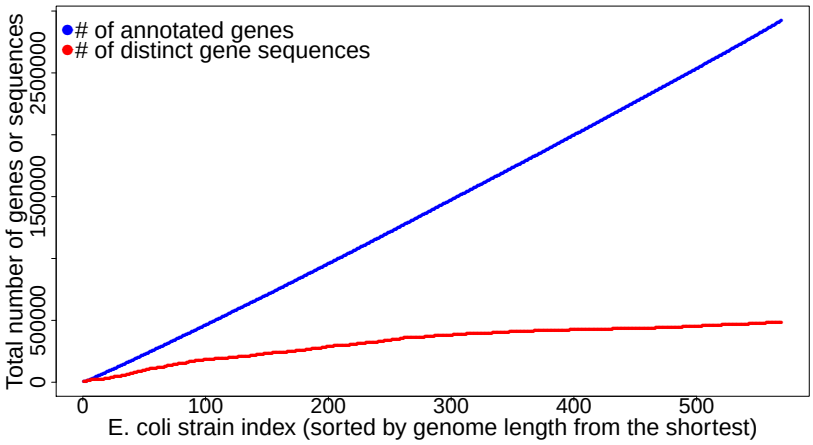

Figure 2.5: Comparison of the number of distinct gene sequences to the total number of gene annotations in original genome annotations of 569 strains of *E. coli*. Strains were included cumulatively in the order of increasing genome sizes. In the figure, the x-axis corresponds to the number of strains included. The increasing gap in the number of different gene annotations and the number of distinct gene sequences is the consequence of highly similar genome sequences, which is a typical phenomenon for bacterial strains within the same species. Based on this observation, we designed the closure procedure in eCAMBer to avoid redundant BLAST queries. Applying this strategy we have optimized eCAMBer to be able to work with datasets comprising hundreds of bacterial strains.

As our case-study experiments show, d is usually much smaller than $n \cdot k$ (see Figure 2.5). However, it should be noted that, potentially, if every annotated ORF sequence in S is different, thus $d = O(n \cdot k)$ then the complexity of eCAMBer might be as for CAMBer.

Importantly, the number of I/O operations per iteration is also significantly decreased, from $O(n \cdot k^2)$ in CAMBer to $O(k)$ in eCAMBer.

2.3.3 Sequence consolidation graph

Having the closure procedure computed we represent its results in the form of graph structures, called *consolidation graphs*. Here we introduce the *sequence consolidation graph*, which is a compact representation of the outcome of the closure procedure in eCAMBer.

In this graph $G_S = (V_S, E_S, E_B)$, nodes represent distinct sequences of annotated ORFs. There are two types of edges, E_B called *accepted-hit edges*, and E_S

called *shared-end edges*. There is an undirected *shared-end edge* $\{s_1, s_2\} \in E_S$ between a pair of sequence nodes if there is a multigene having two ORF annotations (sharing the same stop codon) with these two sequences. There is an undirected *accepted-hit edge* $\{s_1, s_2\} \in E_B$ between a pair of sequence nodes if there is an acceptable hit from sequence q_1 to an ORF with sequence q_2, or if there is an acceptable hit from sequence q_2 to an ORF with sequence q_1.

More formally, we define the graph G_S as follows:

$$\begin{cases} V_S = \bigcup_{S \in \mathcal{S}} \text{sequences}(A_S^c) \\ E_S = \{\{s_1, s_2\} : \exists_m \exists_{\substack{o_1 \in \text{elts}(m) \\ o_2 \in \text{elts}(m)}} s_1 = \text{seq}(o_1) \wedge s_2 = \text{seq}(o_2)\} \\ E_B = \{\{s_1, s_2\} : \exists_{o_2} s_1 \to o_2 \vee \exists_{o_1} s_2 \to o_1\} \end{cases} \tag{2.9}$$

Let us consider the following property: If two ORF annotations o_1 and o_2 have identical sequence, then for every query q, $q \to o_1 \Leftrightarrow q \to o_2$.

Since the BLAST e-value for a hit depends on genome length, it is possible that for two different target genomes the e-value of the corresponding hits will have different values. Thus, it is possible that one of the hits will be acceptable whereas the other not. However, in practice, such cases are very unlikely and, in our opinion, can be neglected in the analysis.

It turns out that, assuming the above mentioned property, sequence consolidation graph is a compact representation of both gene consolidation graph and multigene consolidation.

Assuming the above, the gene consolidation graph $G_O = (V_O, E_O)$ can be derived from the sequence consolidation graph $G_S = (V_S, E_S, E_B)$ following the formula:

$$\begin{cases} V_O = \bigcup_{S \in \mathcal{S}} A_S^c \\ E_O = \{\{o_1, o_2\} : \{\text{seq}(o_1), \text{seq}(o_2)\} \in E_B\} \end{cases} \tag{2.10}$$

Then, having the gene consolidation graph, we can construct the multigene consolidation graph following the 2.7 formula.

Figure 2.6 illustrates the correspondence between the consolidation graphs.

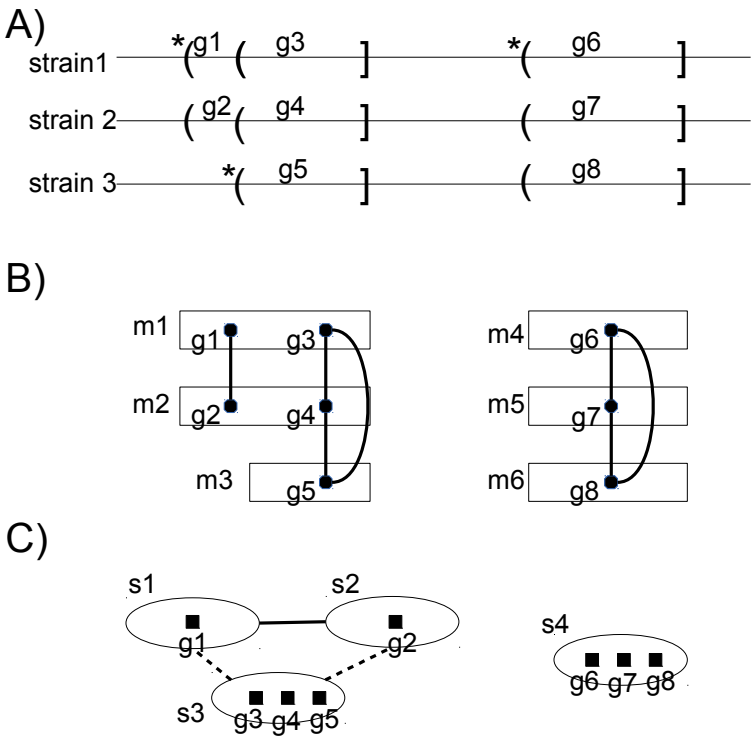

Figure 2.6: Schematic view on the correspondence between graph representations of the closure procedure results in the form of consolidation graphs; **A)** the genomes with marked ORF annotations. Round and square brackets indicate positions of the gene starts and gene ends, respectively. Round brackets with stars indicate original gene annotations, whereas those without starts indicate the transferred TIS annotations; **B)** multigene representation of the annotations with the ORF consolidation graph edges shown between multigene elements, edges of the multigene consolidation graph are not shown for the readability; **C)** the sequence consolidation graph in which nodes correspond to the distinct ORF sequences. In this graph, there is an undirected *shared-end edge* between a pair of sequence nodes if there is a multigene having two ORF annotations (sharing the same stop codon) with these two sequences. There is an undirected *accepted-hit edge* between a pair of sequence nodes if there is an acceptable hit from sequence the first sequence to an ORF with the second sequence, or vice-versa. Shared-end edges are drawn dashed, whereas accepted-hit edges are drawn solid.

2.3.4 Homologous gene clusters

The second step of eCAMBer is to determine homologous gene families as connected components of the multigene consolidation graph G_M. There is a natural one-to-one correspondence between the connected components of the multigene consolidation graph and the connected components of the sequence consolidation graph (the latter connected components are obtained by taking the union of E_S

and E_B). So, in eCAMBer, we do this using connected components of the sequence consolidation graph G_S, because it tends to be smaller for closely related genomes. The obtained set of homologous gene families is represented as a set of disjoint multigene clusters, denoted by C_M.

2.3.5 Refinement procedure

The third step of eCAMBer is the *refinement procedure*. The goal of the refinement procedure is splitting the homologous gene families, represented by multigene clusters, to obtain anchors. We call a multigene cluster an anchor, if it includes at most one multigene for every strain. Analogously, we call a multigene cluster non-anchor, if there is a strain which includes at least two multigenes in the cluster. Multigenes in the same anchor are potentially orthologous to each other, whereas a non-anchor contains at least two multigenes that are non-orthologous. Following CAMBer, we use genomic context information to decompose non-anchors into smaller multigene clusters that can emerge as anchors, as described below.

The input for the refinement procedure consists of the set of multigene clusters C_M, the sequence consolidation graph G_M, and the multigene annotations M_S^c, for each strain $S \in \mathcal{S}$. We start with classifying the set C_M of multigene clusters into two disjoint sets of anchors and non-anchors, denoted C_A and C_N, respectively. We also sort all multigenes within strain contigs by positions of their stop codons. We reconstruct the subgraph of the multigene consolidation graph, called the *refinement graph*. In this graph $G_R = (V_R, E_R)$, nodes V_R are constituted by the subset of multigenes, which belong to non-anchor clusters. There is an edge $\{m_1, m_2\} \in E_R$, between a pair of multigenes coming from different strains, if there is an edge $\{m_1, m_2\} \in E_M$, and the two multigenes belong to the same multigene cluster. By $E_R^{\{S,T\}}$ we denote the subset of edges between multigenes from a pair of strains S and T.

Then, for each unordered pair of strains $\{S, T\}$ we perform the following procedure in parallel. First, for each multigene m, we identify a pair of its nearest neighbors which belong to anchors with a multigene element present on the opposite strain. Such left and right neighbors of m are denoted as $l_m^{\{S,T\}}$ and $r_m^{\{S,T\}}$, respectively. Then, for each edge $\{m_1, m_2\} \in E_R^{\{S,T\}}$ we check whether it is *supported* in the sense that it satisfies one of the following conditions: (i) it connects multi-

genes belonging to a cluster, such that m_1 and m_2 are its only elements in strains S and T; (ii) the corresponding pairs $(l_{m_1}^{\{S,T\}}, l_{m_2}^{\{S,T\}})$ and $(r_{m_1}^{\{S,T\}}, r_{m_2}^{\{S,T\}})$ belong to the same anchor; (iii) the corresponding pairs $(l_{m_1}^{\{S,T\}}, r_{m_2}^{\{S,T\}})$ and $(r_{m_1}^{\{S,T\}}, l_{m_2}^{\{S,T\}})$ belong to the same anchor. If any of the four neighbors does not exist we substitute it with a dummy node, which virtually belongs to any anchor.

Finally, we obtain the *refined graph* G_R^* by removal of unsupported edges from G_R. Then, the set of connected components C_R of G_R^* defines the set of multigene clusters after the split. Finally, we update the set of multigene clusters as $C_M^* \leftarrow (C_M \setminus C_N) \cup C_R$.

The careful reader may notice the differences between the refinement procedures implemented in CAMBer and eCAMBer. First, the refinement procedure in CAMBer performs in iterations until no multigene clusters can be split. In eCAMBer the refinement procedure consists of only one iteration. However, since the input and output for the procedure are of the same type, it can be used multiple times, until no new clusters are split. Second, the condition for an edge to be supported in eCAMBer is more relaxed than that in CAMBer. Both approaches, for a pair of multigenes on different strains, identify pairs of their nearest left and right neighbor multigenes (belonging to anchor clusters with elements on both strains). However, CAMBer checks the actual presence of edges between the neighbors, whereas eCAMBer only checks if the identified neighbors match the same pair of clusters. This approach allows eCAMBer to avoid a costly reconstruction of the whole multigene consolidation graph.

2.3.6 TIS voting procedure

The fourth step of eCAMBer is the *TIS voting procedure*. The goal of the TIS voting procedure is to select the most reliable TIS for each multigene. To do this, we implement an approach based on the concept of majority voting. This strategy has also been used to improve genome annotation accuracy in several recent studies (Zhou and Rudd, 2013; Klassen and Currie, 2013).

In this procedure, for each multigene m in each multigene cluster $c \in C_M^*$, we try to find a TIS (originally annotated or transferred) that belongs to a connected component of the ORF consolidation graph, where the connected component satisfies the following two conditions: (i) it has TISs (originally annotated or transferred) present in at least 80% of the multigenes in c; and (ii) it has TISs

originally annotated in at least 50% of the multigenes in c, or it has TISs originally annotated in at least twice the number of multigenes in c than all other connected components in c. If such a TIS is found, it is selected as the TIS for m. If such a TIS is not found, but m has an originally annotated TIS, then the originally annotated TIS is selected as the TIS for m. If both of these two cases cannot be applied, the TIS corresponding to the longest ORF in the multigene m is selected. After the TIS voting procedure, every multigene has exactly one TIS selected. Thus, we obtain unambiguous TIS annotation for every gene.

Note that the connected components of the sequence consolidation graph — after shared-end edges have been removed — are in a natural one-to-one correspondence with the connected components in the ORF consolidation graph. So in eCAMBer, we implement the TIS voting procedure using the sequence consolidation graph, as it tends to be smaller for closely related genomes.

2.3.7 Clean up procedure

The last step of eCAMBer is the *clean up procedure*, which is designed to filter out multigene clusters which are likely due to gene annotation errors propagated during the closure procedure.

The input for this procedure consists of the set of multigene clusters C_M^* and multigene annotations M_S^c, for each strain $s \in S$. For each multigene cluster $c \in C_M^*$ we compute the following features: (i) l, the median multigene length in c; (ii) p, the ratio of the number of strains with at least one element from c to the total number of strains; (iii) r, the ratio of the number of strains with at least one originally annotated multigene to the total number of strains with at least one element from c; (iv) v, the ratio of the number of multigenes in the cluster that are overlapped by a longer multigene to the total number of multigenes in the cluster.

Then, we update the set of multigene clusters C_M^*, by removing of multigene clusters for which ($p < \frac{1}{3}$ or $r < \frac{1}{3}$) and ($l < 150$ or $v > 0.5$).

2.3.8 Other features of eCAMBer

In order to make eCAMBer more user friendly, we have added a functionality for downloading genome sequences and genome annotations from the PATRIC database, for the set of selected strains within a species. The downloaded data

is automatically formatted as input for eCAMBer. Additionally, eCAMBer integrates Prodigal to generate input gene annotations.

Notably, the current implementation of eCAMBer allows the closure procedure at any iteration. We found out that in many settings one iteration of the closure procedure gives satisfactory results.

The other features implemented in eCAMBer aim in preparation of the input genotype data for GWAMAR — the tool we have developed for detection of genotype-phenotype associations (see chapter 3 for more details).

In this order, eCAMBer employs MUSCLE — the software for computing multiple sequence alignments (Edgar, 2004). Specifically, eCAMBer, uses MUSCLE to compute multiple sequence alignments for amino-acid sequences of genes in the identified gene families. It also uses MUSCLE to compute multiple sequence alignments of promoter regions (-50bp downstream the TIS) for these gene families.

Having computed the multiple sequence alignments, eCAMBer identifies point mutations based on columns in the alignments with at least one differing character. Applying this strategy, eCAMBer identifies amino-acid point mutations inside the protein-coding genes, as well as mutations inside the gene promoter regions.

Additionally, eCAMBer identifies gene gain/loss mutations based on the structure of the corresponding gene families.

The identified point mutations and gene gain/loss mutations constitute the input genotype data for GWAMAR.

Furthermore, eCAMBer supports employing of PHYLIP (Felsenstein, 2005) and PhyML (Guindon et al., 2010) — the software for reconstruction of the phylogenetic tree based on the maximal-likelihood approach.

Finally, eCAMBer generates output compatible with CAMberVis (Woźniak et al., 2011b), a tool for simultaneous visualization of multiple genome annotations of bacterial strains. CAMBerVis also handles visualization of genome annotation inconsistencies.

2.4 CAMBerVis: visualization software to support comparative analysis of multiple bacterial strains

The amount of data that is being generated stimulates active development of visualization techniques and softwares (Nielsen et al., 2010), which are invaluable to biologists for manual curation of results in cases where standard genome annotation tools produce inconsistencies.

In order to better support this type of analysis, we have implemented CAMBerVis — a software which allows for visual comparison of the genome structure annotations of multiple bacterial strains. According to our knowledge, it is the first visualization software distinguishing original and predicted genome structure annotations. Another advantage of CAMBerVis over existing softwares is its intuitive management of plasmids, which are common in bacteria.

CAMBerVis is a standalone application written in Java, which makes it a cross-platform application, tested on Windows, Mac and Linux. Notably, it is implemented based on the Netbeans IDE platform, which makes the application flexible and easy to extend. CAMBerVis, integrated with two example datasets for *M. tuberculosis* and *S. aureus*, is freely available at the project website, `http://bioputer.mimuw.edu.pl/ecamber`.

2.4.1 Example usage

The input to CAMBerVis consists of genome FASTA files and a file with predicted genome structure annotation. The file format can be generated automatically with the use or CAMBer or eCAMBer. However, this format is generic and not dependent on these tools. A user may find more details of the format in the software documentation and learn from the integrated examples on *M. tuberculosis* and *S. aureus*.

Here we describe the main features of CAMBerVis based on a typical use case. In the first step we identify a gene family of interest with some annotation inconsistencies. CAMBerVis manages statistics for every gene family in a table in the *ComponentsStats* window. Using this table we can easily find gene families with missing gene annotations or inconsistencies among annotated TISs.

Second, the visualization is automatically focused on the selected gene family showing simultaneously its multigenes (with both annotated and predicted

TISs) in all strains. We may also see their neighborhood in different scales using intuitive genome navigation.

Third, we use on-the-fly comparative analysis supported by CAMBerVis. For example, in the case of inconsistently annotated TISs, we may compare promoter regions by multiple alignments using the integrated CLUSTALW. CAMBerVis also enables external queries via NCBI BLAST API, which can be applied to check which TIS is the most often annotated in external databases like for example NCBI Non-redundant (nr) database.

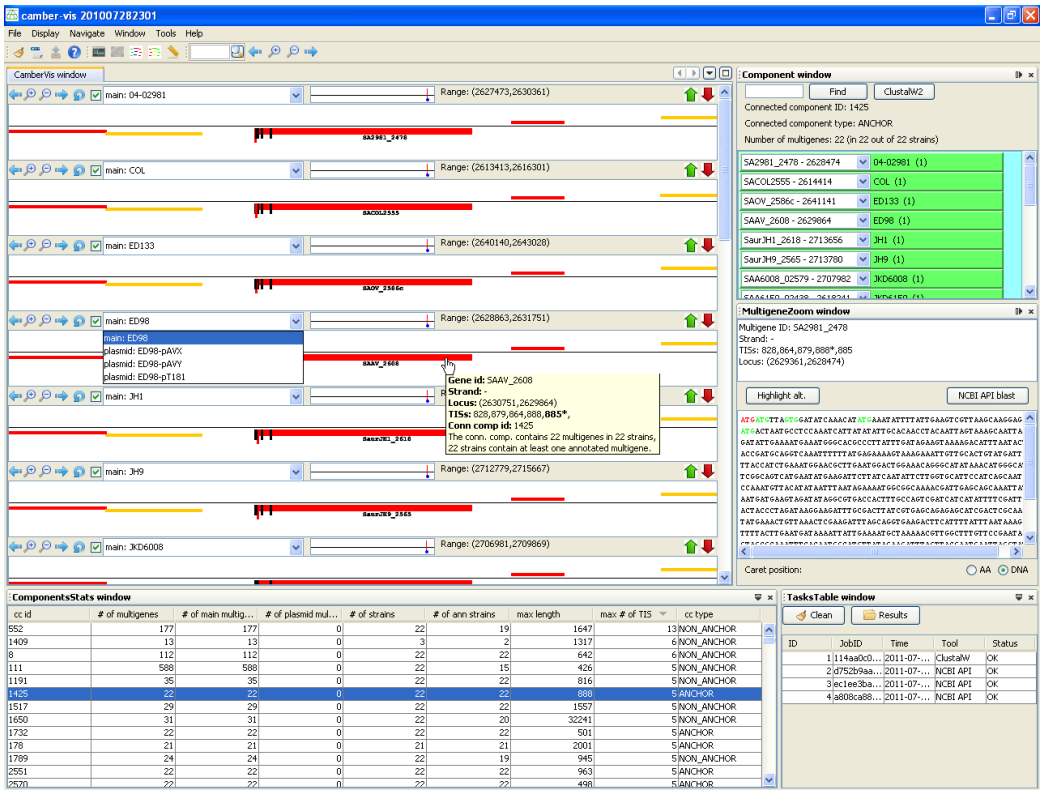

Figure 2.7: The main view of the CAMBerVis interface with loaded example data for 22 strains of *S. aureus*. The view is focused on a highly conserved gene cluster (gene family) with 5 different TISs in each multigene, selected from the list in the "ComponentsStats" window. Multigenes are visualized as horizontal rectangles, with TISs presented as vertical ticks (originally annotated TISs are red and long). The window "TasksTable" keeps track of results obtained on-the-fly by ClustalW or NCBI BLAST API.

Figure 2.7 presents a screen shot of the running application. The visualization is focused on a gene family identified by "ComponentsStats" table sorted by the number of TISs. There are 5 different TISs annotated in GenBank among the

22 fully sequenced strains of *S. aureus*, annotated with following frequencies: 2,1,7,4,8 (ordered from the TIS giving rise to the shortest gene to the TIS giving rise to the longest gene). An analysis of the multiple alignment computed by CLUSTALW showed that the gene family is highly conserved among strains and only four of the strains have SNP in the 100bp long promotor region. Queries to the NCBI Non-redundant (nr) database showed that the TIS that yields the longest gene is the most often annotated.

2.5 Results and discussion

Here we present and discuss in detail results obtained by employing the tools we developed to support comparative analysis of multiple bacterial strains. The description goes along the lines of articles introducing the tools (Woźniak et al., 2011a, 2014a).

2.5.1 Results for CAMBer

Here we present details of applying CAMBer to three datasets of 9 *M. tuberculosis*, 22 *S. aureus* and 41 *E. coli* strains. It was ran with the following parameters: $e_t = 10^{-10}$, $p_t = 0.3$ and $n_t = 30.5\%$.

The input datasets were generally taken from GenBank (Benson et al., 2013), with the exception of six *M. tuberculosis* strains. The input datasets for three of these strains came from the Broad Institute database; while the remaining three came from the supplementary material of (Ioerger et al., 2009).

2.5.1.1 Mycobacterium tuberculosis

Table 2.1 provides source information for the strains. We notice that there is substantial variance (left box plot in Figure 2.8) in the number of originally annotated genes. This is probably due to different gene-finding tools and methodologies being applied by different labs, rather than the real genomic composition.

It is quite remarkable that variance in the number of predicted multigenes after the closure procedure is much smaller (right box plot in Figure 2.8). Table 2.2 shows for each strain the distribution of multigenes with respect to the number of elements (TISs). By far the largest group in each strain are one-element

strain ID	source	resist.	# of genes	lab.
H37Rv	NC_000962	DS	3988(26)	S
H37Ra	NC_009525	DS	4034(22)	C
F11	NC_009565	DS	3941(5)	B
KZN 4207(T)	Ioerger et al. (2009)	DS	3902(47)	T
KZN 4207(B)	Broad Institute	DS	3996(4)	B
KZN 1435	Broad Institute	MDR	4059(10)	B
KZN V2475	Ioerger et al. (2009)	MDR	3893(3792)	T
KZN 605	Broad Institute	XDR	4024(26)	B
KZN R506	Ioerger et al. (2009)	XDR	3902(46)	T

Table 2.1: Details for input strains for the *M. tuberculosis* case study. The first number in column called "# of genes" corresponds to the number of annotated genes, the second (in brackets) corresponds to the number of genes excluded in the study due to unusual start or stop codons or sequence length not divisible by three. In order to avoid ambiguity in naming the same strain sequenced by two labs we introduce an additional suffix (T or B). Characters in last column, called "lab.", describe the sequencing laboratories: B - The Broad Institute, T - Texas A&M University, C - Chinese National Human Genome Center at Shanghai, S - Sanger Institute.

	# of multigenes with a given number of elements					
	5	4	3	2	1	total
F11	1	6	68	605	3475	4155
H37Ra	1	5	66	607	3488	4167
H37Rv	1	6	66	602	3483	4158
KZN 605	1	6	68	602	3457	4134
KZN 1435	1	6	69	597	3472	4145
KZN 4207(T)	1	6	70	600	3463	4140
KZN R506	1	6	70	602	3459	4138
KZN V2475	1	6	70	601	3461	4139
KZN 4207(B)	1	5	69	602	3465	4142

Table 2.2: Statistics for the number of multigene start sites after applying the closure procedure to the dataset of 9 *M. tuberculosis* strains.

multigenes. Also, Figure 2.9 shows that the predicted multigenes are quite even distributed in terms of gene length.

The careful reader may have also noticed that the same strain (*KZN 4207*) sequenced in two labs has quite different numbers of annotated genes (3902 vs. 3996); but after the closure procedure we have for these two genomes almost the same number of multigenes (4140 vs. 4142).

This case study shows that the method can also be applied to completely unannotated genomes, yielding an initial annotation of a newly sequenced genome. For example, due to a shift in annotation coordinates for the strain *KZN V2475* we removed most of the gene annotations (after the shift). Using our method,

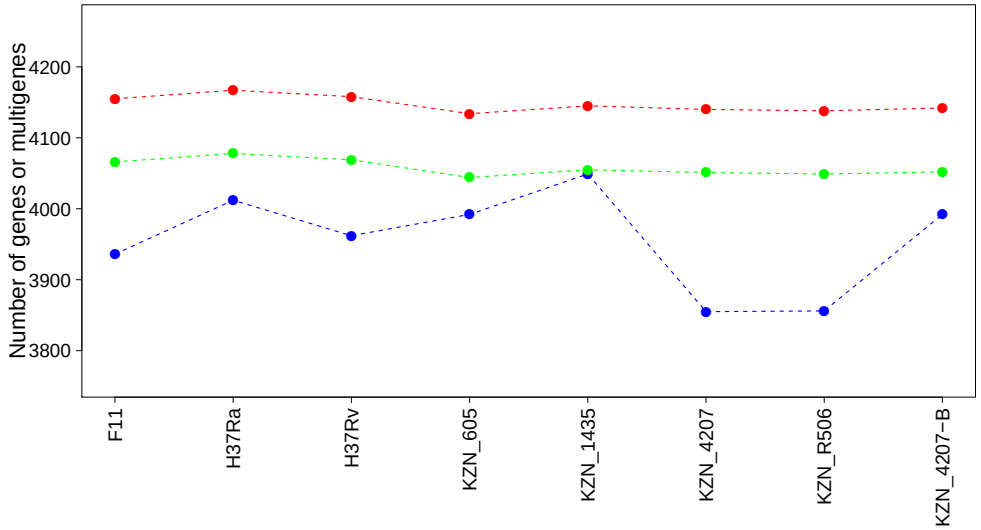

Figure 2.8: This plot presents the impact of CAMBer on the number of annotated genes in the dataset of 9 *M. tuberculosis* strains. On the x-axis strains are listed (from left to right) in descending order of their genome size. The blue points and the red points present respectively the number of originally annotated genes and the number of multigenes after the closure procedure for each strain. The green points indicate the numbers of multigenes after the closure procedure and after applied post-processing of removal multigenes shorter than 200 nucleotides length.

we were able to annotate 4139 multigenes in the genome.

After refinement of the multigene consolidation graph, the number of connected components rose from 4177 to 4287 (see Table 2.3), but size of the largest component dropped from 127 (there are two such components in the multigene consolidation graph) to 15 (only one such component after refinement). Also the maximal number of multigenes in one strain and in one non -anchor dropped from 17 in the multigne consolidation graph to 3 in the refined consolidation graph.

connected components	before refinement	after refinement
all	4178	4287
core	3986	4030
orphans	48	68
anchors	4136	4265
non-anchors	42	22
core anchors	3945	4012

Table 2.3: Statistics for the number of connected components with respect to their types, before and after the refinement procedure applied to the dataset of 9 *M. tuberculosis* strains.

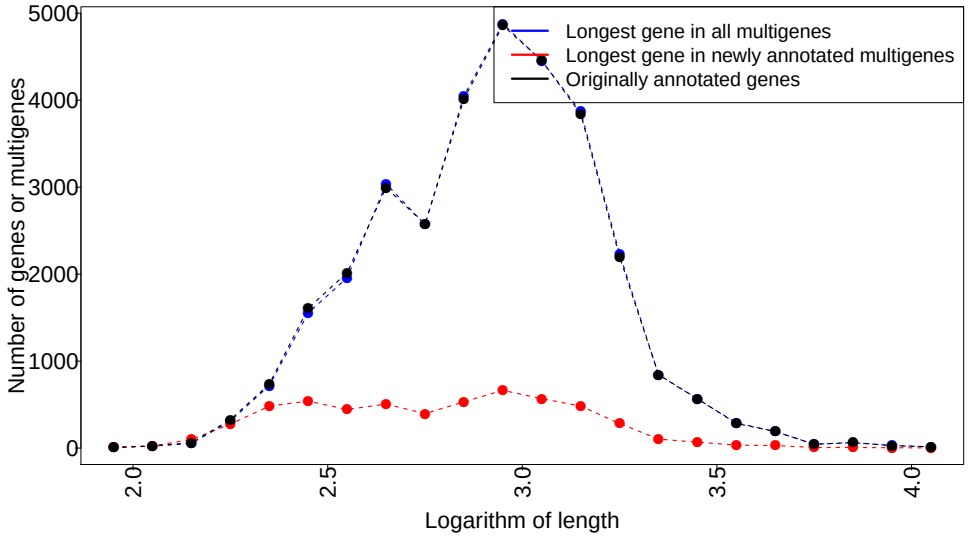

Figure 2.9: Histograms of gene lengths in logarithmic scale (base = 10) for all the *M. tuberculosis* strains taken together. The x-axis is quantified into ranges of length 0.1. Black points presents the numbers of originally annotated genes, blue points indicate the numbers of multigenes after applying CAMBer, red points indicate the numbers of multigenes formed during the closure procedure.

It is interesting to compare the two largest components of the multigene consolidation graph. As mentioned above they have in total 127 multigenes, each strain having between 12 and 17 multigenes in these non-anchors. What is remarkable here is that *H37Rv*, having 16 multigenes in each of the two components, has all of these 32 genes annotated in the *TubercuList* database, developed by Lew et al. (2011), as transposons which belong to the same insertion element (*IS6110*). Even though these two non- anchors were not successfully resolved by the refinement procedure, the resulting non-anchors (four obtained from each of the original two large non- anchors in the multigene consolidation graph) are pretty small: at most two multigenes per strain. More precisely, each of the original non-anchors was split by the refinement procedure into 34 subclusters (4 non-anchors, and 30 anchors with 9 orphans).

The multigene consolidation graph contains 4177 connected components, with only 43 components (about 1%) being non-anchors and 48 being orphans. After the refinement procedure we obtained slightly more connected components (4287), but the number of non-anchors substantially dropped to 22 (Table 2.3). Figure 2.10 gives another point of view for the refinement procedure results.

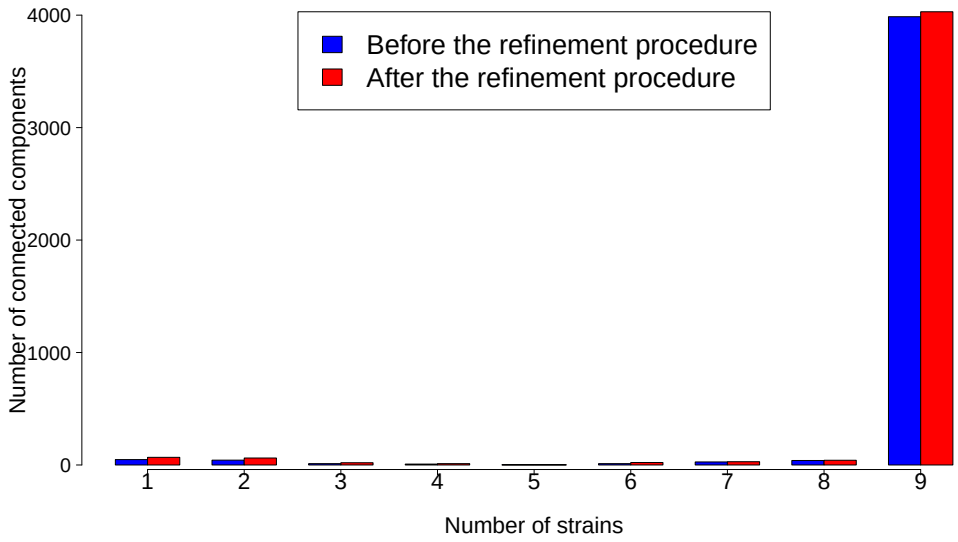

Figure 2.10: Histogram of the number of connected components (y-axis) shared by a particular number of strains (x-axis). For better clarity only numbers of connected components after the refinement procedure are shown.

With this approach we were also able to discover five cases of gene fusion/fission in the investigated genomes which seems pretty unusual for such closely related strains. We leave the analysis of this phenomenon for further study.

2.5.1.2 Staphylococcus aureus

Two methicillin-resistant strains (*N315* and *Mu50*) are the first fully sequenced *S. aureus* genomes (Kuroda et al., 2001).

Genome sequences and annotations of 22 fully sequenced strains were used in our study. At the time of writing, these were the only available *S. aureus* strains with "completed" sequencing status. Table 2.4 presents the list of the strains.

In this medium-size case study most of the results and corollaries are similar to the *M. tuberculosis* case study. However, we highlight below three interesting observations.

The first observation is that there is a much large number of short predicted multigenes compared to the number of short original annotated genes, as shown in Figure 2.11. This contrasts sharply with the situation for *M. tuberculosis* depicted in Figure 2.9. This means that in *S. aureus* many strains have short original annotations that are annotated to one of them but not to other strains, even

strain ID	GenBank ID	# of genes	genome size	lab.
TW20 0582	FN433596	2769(5)	3043210	Welcome Trust Sanger Institute
JKD6008	CP002120	2680(0)	2924344	Monash University
JH9	CP000703	2769(5)	2906700	US DOE Joint Genome Institute
JH1	CP000736	2680(0)	2906507	US DOE Joint Genome Institute
MRSA252	BX571856	2697(0)	2902619	Sanger Institute
Mu3	AP009324	2746(0)	2880168	Juntendo University
Newman	AP009351	2655(5)	2878897	Juntendo University
Mu50	BA000017	2699(63)	2878529	Juntendo University
USA300 TCH1516	CP000730	2624(0)	2872915	Baylor College of Medicine
USA300 FPR3757	CP000255	2699(61)	2872769	University of California
ST398 S0385	AM990992	2657(0)	2872582	University Medical Centre Utrecht
ED133	CP001996	2560(0)	2832478	University of Edinburgh
ED98	CP001781	2699(0)	2824404	University of Edinburgh
04-02981	CP001844	2653(2)	2821452	Robert Koch Institute
NCTC 8325	CP000253	2661(0)	2821361	University of Oklahoma
MW2	BA000033	2650(59)	2820462	NITE
N315	BA000018	2892(0)	2814816	Juntendo University
JKD6159	CP002114	2632(6)	2811435	University of Melbourne
COL	CP000046	2593(59)	2809422	TIGR
TCH60	CP002110	2555(1)	2802675	Baylor College of Medicine
MSSA476	BX571857	2672(1)	2799802	Sanger Institute
RF122	AJ938182	2673(0)	2742531	University of Minnesota

Table 2.4: Details for input strains used in the *S. aureus* case study. The first number in column called "# of genes" corresponds to the number of annotated genes, the second (in brackets) corresponds to the number of genes excluded in the study due to unusual start or stop codons or sequence length not divisible by three.

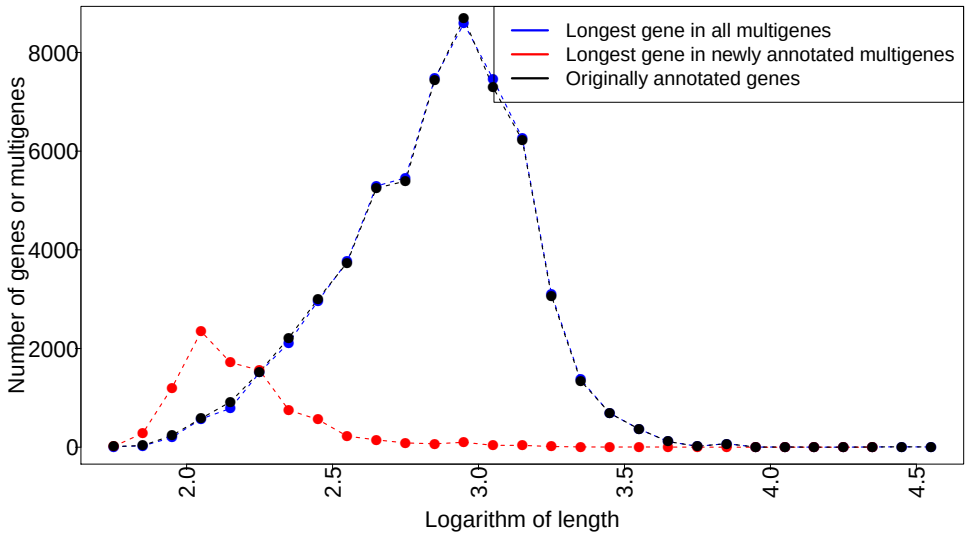

Figure 2.11: Histograms of gene lengths in logarithmic scale (base = 10) for all the *S. aureus* strains taken together. The x-axis is quantified into ranges of length 0.1. Black points presents the numbers of originally annotated genes, blue points indicate the numbers of multigenes after applying CAMBer, red points indicate the numbers of multigenes formed during the closure procedure.

though highly homologous regions exist in other strains. This suggests possible higher occurrence of annotation errors in short genes of *S. aureus*, especially in strains like *NCTC 8325*; see Figure 2.12.

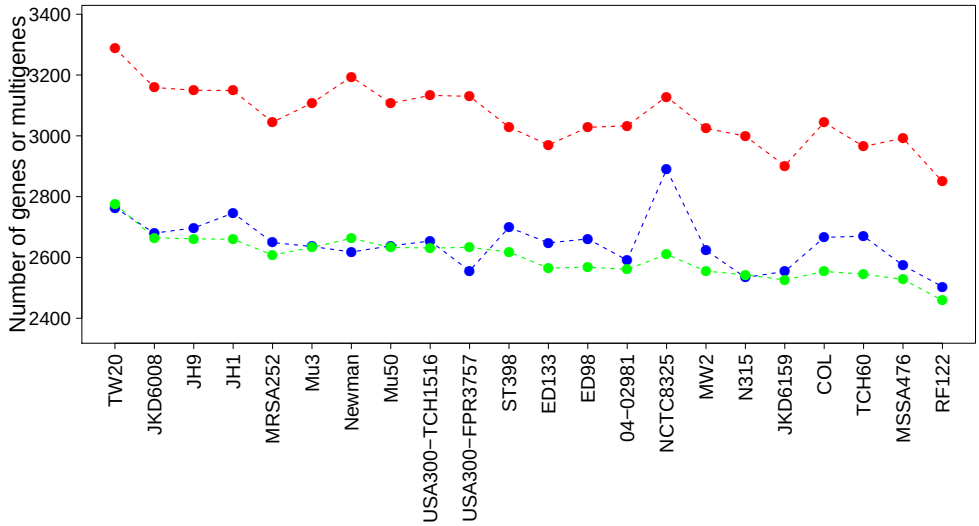

Figure 2.12: This plot presents the impact of CAMBer on the number of annotated genes in the dataset of 22 *S. aureus* strains. On the x-axis strains are listed (from left to right) in descending order of their genome size. The blue points and the red points present respectively the number of originally annotated genes and the number of multigenes after the closure procedure for each strain. The green points indicate the numbers of multigenes after the closure procedure and after applied post-processing of removal multigenes shorter than 200 nucleotides length.

The second observation is that computing of the closure procedure took 8 iterations, which is similar to the much larger study of *E. coli* (8 iterations) and more than the *M. tuberculosis* case study (3 iterations).

The third observation is that the maximal number of TISs in a multigene is 13 (see Table 2.5 for more details), where for *E. coli* it is 9 and for *M. tuberculosis* 5.

As in the other case studies, we observe uneven distribution in the number of original annotated genes; see Figure 2.12. To assess the degree of unevenness we calculated the mean absolute difference in counts coming from two neighboring strains, where strains are ordered in decreasing order of the size of their genomes. It is 78 for the original annotation curve vs. 70 for the curve constructed after the closure procedure, which further drops to 29 after post-processing removal

	# of multigenes with a given number of elements (TISs)											
	13	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**total**
TW20	0	0	1	0	1	3	9	45	224	823	2183	3289
JKD6008	0	0	1	0	1	2	8	44	218	827	2058	3159
JH9	0	0	0	1	0	0	10	42	240	805	2052	3150
JH1	0	0	0	1	0	0	11	43	241	805	2048	3149
MRSA252	9	0	0	0	1	2	8	44	207	805	1970	3046
Mu3	0	0	0	1	0	0	12	39	235	789	2032	3108
Newman	0	0	1	0	1	2	12	46	231	818	2089	3200
Mu50	0	0	0	1	0	0	12	39	234	788	2033	3107
USA300 TCH1516	0	0	1	0	1	2	12	49	237	815	2020	3137
USA300 FPR3757	0	0	1	0	1	2	12	49	239	813	2016	3133
ST398	0	0	1	0	0	0	6	39	198	768	2017	3029
ED133	0	0	1	0	0	1	9	41	212	762	1946	2972
ED98	0	0	0	1	0	0	11	38	235	769	1974	3028
04-02981	0	0	0	1	0	0	11	40	236	778	1967	3033
NCTC8325	0	0	1	0	1	2	11	44	228	799	2044	3130
MW2	0	0	0	0	0	3	11	45	230	790	1948	3027
N315	0	0	0	1	0	0	12	40	234	765	1947	2999
JKD6159	6	1	0	0	0	0	9	38	208	760	1880	2902
COL	0	0	1	0	1	2	13	49	234	785	1964	3049
TCH60	4	1	1	0	0	1	8	48	192	776	1936	2967
MSSA476	0	0	0	0	0	3	11	42	225	780	1933	2994
RF122	0	0	0	0	0	5	8	40	186	706	1905	2850

Table 2.5: Statistics for the number of multigene start sites after applying the closure procedure to the dataset of 22 *S. aureus* strains.

of multigenes shorter than 200 nucleotides.

This inconsistency was probably caused by different gene-finding methodologies applied by different labs. Curves like those presented in Figure 2.12 allow us also to estimate which labs were more conservative and which were more liberal when calling a given ORF a gene. For example, we observe a big peak in the number of original annotated genes for the strain *NCTC 8325*, suggesting that this is perhaps the case of a more liberal annotation. Indeed, we investigated the number of connected components with multigenes present in all strains but have original gene annotations in only one strain. It turned out that there are only 7 strains that contribute at least one such connected component, of which the strain *NCTC 8325* contributes the highest number (22), with the second strain being *USA300 TCH1516* (18). All other strains contributed less than 4 such components. An example of a strain with a rather conservative annotation is *USA300 FPR3757*, as can be clearly seen from a dip of the curve in Figure 2.12.

It is rather expected that most of the inconsistencies concern short genes, leading to a sudden increase in the number of short multigenes after the closure procedure; see Figure 2.11. Therefore, it is interesting to investigate the cases where new long multigenes are predicted after the closure procedure. There are

in total 31 connected components with multigenes of length at least 300 nucleotides which were originally annotated in less than half of the strains. Two of them have multigenes in all 22 strains with only one originally annotated element. More precisely, these two connected components were contributed by genes *SAOUHSC 00630* and *SAOUHSC 01489* annotated in *NCTC 8325*. Both these genes are overlapped by genes which have original annotations in all remaining strains, which suggests that these two genes were perhaps incorrectly annotated.

We also checked the structure of annotations for highly overlapping multigenes as another source of possible inconsistencies in genome annotations. For each strain, we searched for pairs of highly overlapping multigenes (after the closure procedure) belonging to core anchors (i.e., anchors with elements in every strain). Here, we define a pair of multigenes as highly overlapping when the length of the overlap is at least 50% of the length of the shorter multigene in the pair.

The number of identified pairs of multigenes in one strain varies from 17 to 20, depending on the strain. As it can be expected, strains with more liberal annotations have higher number of annotated overlapping multigene pairs. In particular, *NCTC 8325* has 7 pairs of multigenes where both multigenes in the pair have at least one original annotated element; *ST398* has 5 such pairs; and *ED98* has 4. On the other hand, *RF122*, *SA300 FPR3757*, *Newman*, *N315* and 8 other strains do not have any such highly overlapping pair of annotated multigenes.

connected components	**before refinement**	**after refinement**
all	4737	5528
core	2156	2146
anchors	4464	5421
orphans	839	1373
non-anchors	273	107
core anchors	2115	2119

Table 2.6: Statistics for the number of connected components with respect to their types, before and after the refinement procedure applied to the dataset of 22 *S. aureus* strains.

Table 2.6 presents statistics of the refinement procedure. After the closure procedure, we obtained 273 (around 5%) non-anchors in the multigene consolidation graph, of which the refinement procedure split 210 and completely resolving 175 of them. The refinement procedure yielded 4 new anchors with multigenes in all strains. Figure 2.13 gives another perspective on the refinement procedure results.

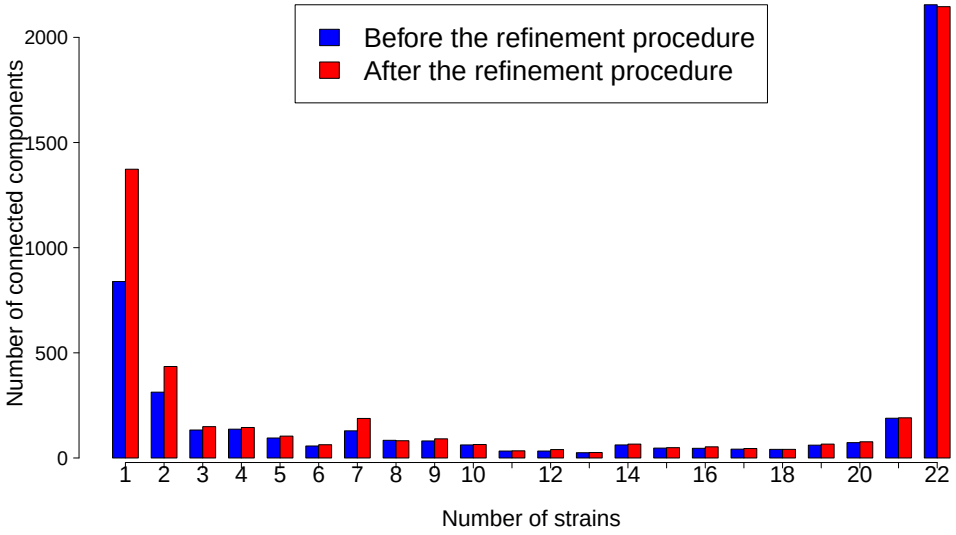

Figure 2.13: Histogram of the number of connected components (y-axis) shared by a particular number of strains (x-axis) for the dataset of 22 *S. aureus* strains.

2.5.1.3 Escherichia coli

The strain *K-12 MG1655* became the first fully sequenced *E. coli* genome in 1997 (Blattner et al., 1997).

We perform the analysis on *E. coli* to test scalability of CAMBer and check stability of the results on a large dataset. In our case study, we use genome sequences and annotations of 41 fully sequenced strains deposited in NCBI. At the time of writing, these were the only available *E. coli* strains with "completed" status. Table 2.7 presents details of the strains.

Figure 2.14 presents a distribution of gene (original annotation) and multigene (after applying our closure procedure) counts for the 41 strains. Strains in this plot occur (from left to right) in decreasing order of sizes of their genomes. We observe that the curve based on the original annotations is quite bumpy, which reflects incongruence of annotations made by different labs. This observation is supported by computing an average absolute difference in counts coming from two neighboring strains: it is 152.1 for the original annotation curve vs. 95.6 for the curve constructed after the closure procedure; and it is only 64 after post-processing removal of multigenes shorter than 200 nucleotides was applied.

We have also analyzed the distribution of sizes of the newly predicted multigene

strain ID	GenBank ID	# of genes	genome size	lab.
O26:H11 11368	AP010953	5363(4)	5697240	University of Tokyo
O157:H7 EC4115	CP001164	5315(0)	5572075	J. Craig Venter Institute
O157:H7 EDL933	AE005174	5348(10)	5528445	University of Wisconsin
O157:H7 TW14359	CP001368	5263(6)	5528136	University of Washington
O157:H7 Sakai	BA000007	5360(5)	5498450	GIRC
O103:H2 12009	AP010958	5053(4)	5449314	University of Tokyo
O55:H7 CB9615	CP001846	5014(0)	5386352	Nankai University
O111:H 11128	AP010960	4971(4)	5371077	University of Tokyo
042	FN554766	4792(18)	5241977	Welcome Trust Sanger Institute
CFT073	AE014075	5378(4)	5231428	University of Wisconsin
ED1a	CU928162	4914(4)	5209548	Genoscope
UMN026	CU928163	4825(4)	5202090	Genoscope
55989	CU928145	4762(4)	5154862	Institute Pasteur and Genoscope
ETEC H10407	FN649414	4696(3)	5153435	Welcome Trust Sanger Institute
IAI39	CU928164	4731(7)	5132068	Genoscope
ABU 83972	CP001671	4793(6)	5131397	Georg-August-University
IHE3034	CP001969	4757(3)	5108383	IGS
APEC O1	CP000468	4467(3)	5082025	Iowa State University
SMS-3-5	CP000970	4742(3)	5068389	TIGR
UTI89	CP000243	5066(13)	5065741	Washington University
S88	CU928161	4695(3)	5032268	Genoscope
UM146	CP002167	4650(0)	4993013	MBRI
E24377A	CP000800	4755(0)	4979619	TIGR
O127:H6 E2348/69	FM180568	4553(4)	4965553	Sanger Institute
536	CP000247	4629(2)	4938920	University of Goettingen
W	CP002185	4478(4)	4900968	AIBN/KRIBB
SE11	AP009240	4679(0)	4887515	Kitasato Institute for Life Sciences
O83:H1 NRG 857C	CP001855	4429(13)	4747819	Public Health Agency of Canada
ATCC 8739	CP000946	4180(7)	4746218	US DOE Joint Genome Institute
SE15	AP009378	4338(0)	4717338	Kitasato University
IAI1	CU928160	4353(4)	4700560	Genoscope
K-12 substr. DH10B	CP000948	4125(5)	4686137	University of Wisconsin-Madison
K-12 substr. W3110	AP009048	4225(9)	4646332	Nara Institute
HS	CP000802	4383(3)	4643538	TIGR
K-12 substr. MG1655	U00096	4144(7)	4639675	University of Wisconsin-Madison
DH1	CP001637	4159(4)	4630707	US DOE Joint Genome Institute
BL21-Gold(DE3)pLysS	CP001665	4208(8)	4629812	US DOE Joint Genome Institute
BW2952	CP001396	4083(5)	4578159	TEDA School
BL21(DE3) BL21	AM946981	4227(4)	4570938	Austrian Center
B REL606	CP000819	4158(6)	4558953	International *E. coli* B Consortium
BL21(DE3)	CP001509	4181(23)	4558947	Korea Research Institute

Table 2.7: Details for input strains for the *E. coli* case study. The first number in column called "# of genes" corresponds to the number of annotated genes, the second (in brackets) corresponds to the number of genes excluded in the study due to unusual start or stop codons or sequence length not divisible by three.

annotations. Figure 2.15 presents these distributions for all *E. coli* strains taken together. The striking feature is that most of the newly predicted multigenes are pretty short, around 200 nucleotides. Of course each such newly predicted multigene must have a witness coming from an original annotation in another strain. This distribution suggests that annotations of short genes may be a possible source of annotation errors. It also suggests one should remove very short multigenes from global considerations. The distribution after removal is flatter,

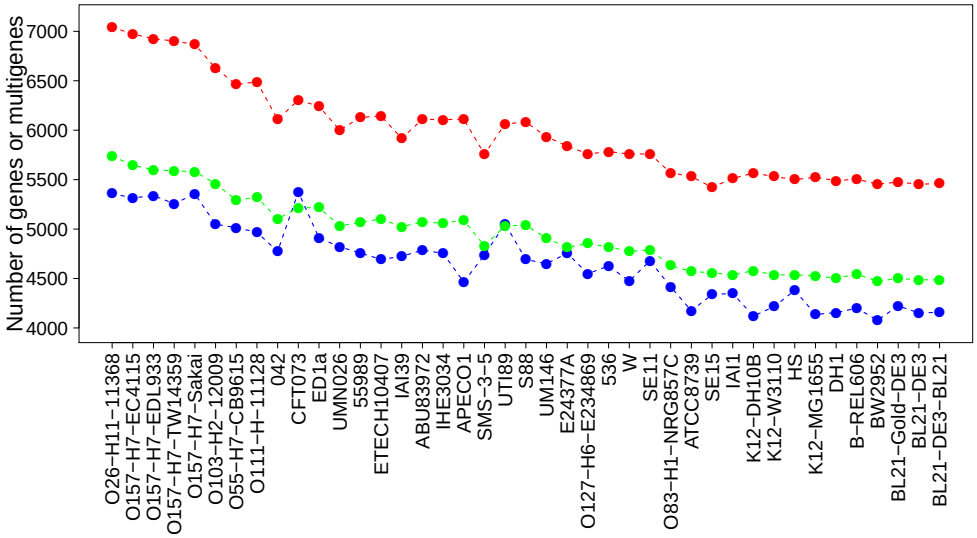

Figure 2.14: This plot presents the impact of CAMBer on the number of annotated genes in the dataset of 41 *E. coli* strains. On the x-axis strains are listed (from left to right) in descending order of their genome size. The blue points and the red points present respectively the number of originally annotated genes and the number of multigenes after the closure procedure for each strain. The green points indicate the numbers of multigenes after the closure procedure and after applied post-processing of removal multigenes shorter than 200 nucleotides length.

resembling closer to the distribution for original annotated genes, as shown in Figure 2.14.

It is also interesting to investigate which strains had the most liberal annotations of genes. This can be seen by considering connected components which have an element in each strain, but only one gene in such a component has original annotation. Such a situation suggests that the lab which was annotating this strain annotated the ORF as a gene, while other labs did not, even though the corresponding ORF was present in genomes that the other labs were working on. The top 5 most liberal annotations were obtained for *CFT073* (37 components), *E24377A* (22 components), *O157-H7 EC4115* (13 components), *UTI89* (12 components), and *IAI1* (10 components). For the rest of the strains, the number of such components was smaller than 8. In total, there were 22 strains of *E. coli* which contributed components described above.

Adopting a similar approach as in the *S. aureus* case study we performed the analysis of annotations for highly overlapping multigenes viewed as another source of inconsistencies in genome annotations. In the case of *E. coli* strains,

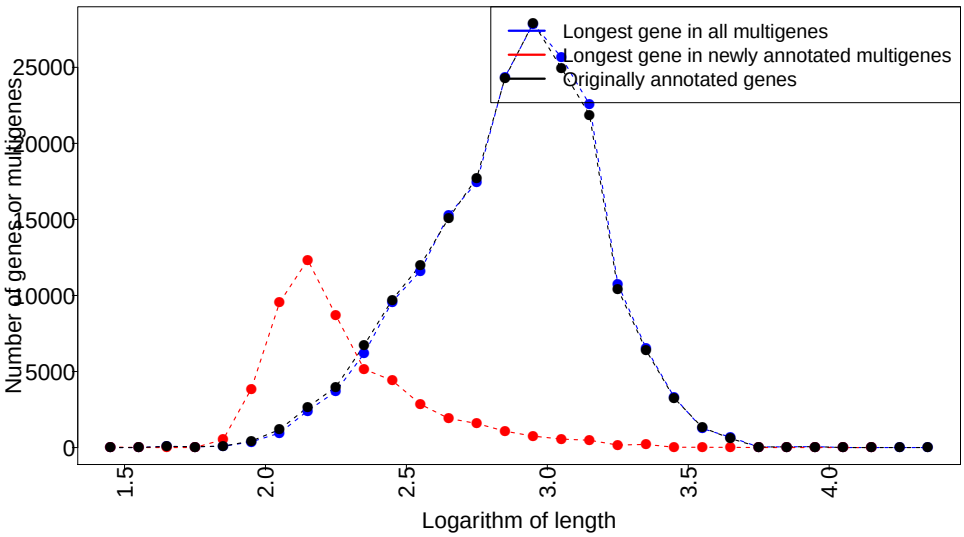

Figure 2.15: Histograms of gene lengths in logarithmic scale (base = 10) for all the *E. coli* strains taken together. The x-axis is quantified into ranges of length 0.1. Black points presents the numbers of originally annotated genes, blue points indicate the numbers of multigenes after applying CAMBer, red points indicate the numbers of multigenes formed during the closure procedure.

the number of highly overlapping pairs of multigenes varies in strains from 167 to 172.

Again, strains with local maxima on the curve of annotated genes (see Figure 2.14) tend to have a higher number of pairs of highly overlapping multigenes with both multigenes annotated. In particular, *CFT073* has 86, *UTI89* has 76, and *E24377A* has 30. On the other hand, *APECO1* has only one such pair.

Even though there are known cases of functional genes with atypical start codons, we decided to restrict our attention to the three typical start codons (ATG, GTG, CTG), hoping that it does not influence our results in a substantial way. However, it is interesting to follow the fate of genes which have atypical start codons in some strains. For example, the first fully sequenced *E. coli* strain (*K-12 MG1655*) has annotated two protein-coding genes with atypical start codons.

The first gene is *infC*, encoding IF3 translation initiation factor. As discussed by Sacerdot et al. (1982), this atypical start codon (ATT) may be in use for self-regulation. Interestingly, using CAMBer, we revealed that annotations for 25 (i.e., more than half) of the studied strains have annotated a shorter version of the gene (435 nucleotides instead of 543) with the GTG start codon. The second

	# of multigenes with a given number of elements (TISs)									
	9	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**total**
O26-H11-11368	7	7	4	20	57	213	631	1793	4310	7042
O157-H7-EC4115	13	13	7	14	62	157	624	1857	4226	6973
O157-H7-EDL933	10	13	5	18	46	143	617	1831	4242	6925
O157-H7-TW14359	14	12	7	13	58	151	616	1836	4195	6902
O157-H7-Sakai	14	8	5	16	49	152	600	1826	4198	6868
O103-H2-12009	0	28	3	16	53	162	583	1704	4078	6627
O55-H7-CB9615	0	4	10	12	44	156	564	1722	3950	6462
O111-H-11128	35	7	1	18	54	154	565	1686	3970	6490
042	0	2	2	11	28	138	538	1598	3791	6108
CFT073	6	2	4	8	33	161	534	1721	3836	6305
ED1a	1	4	0	11	24	144	524	1577	3957	6242
UMN026	0	3	7	9	29	139	539	1556	3719	6001
55989	0	2	3	11	37	146	559	1605	3766	6129
ETECH10407	1	3	2	11	36	143	549	1589	3809	6143
IAI39	22	5	2	4	43	149	508	1619	3566	5918
ABU83972	0	3	3	7	29	140	530	1662	3736	6110
IHE3034	0	1	2	9	32	144	563	1644	3712	6107
APECO1	0	1	2	12	29	145	542	1675	3705	6111
SMS-3-5	3	0	5	8	24	116	500	1515	3586	5757
UTI89	1	1	2	9	30	147	561	1655	3658	6064
S88	0	2	3	9	33	149	550	1658	3678	6082
UM146	1	1	1	8	28	137	528	1590	3640	5934
E24377A	0	1	2	6	31	125	516	1502	3656	5839
O127-H6-E234869	0	3	2	8	15	169	471	1474	3618	5760
536	1	0	2	8	21	135	510	1560	3546	5783
W	0	1	2	6	27	112	483	1492	3636	5759
SE11	0	3	0	9	32	119	505	1467	3625	5760
O83-H1-NRG857C	0	1	2	7	23	117	489	1503	3427	5569
ATCC8739	0	1	3	6	26	106	491	1431	3468	5532
SE15	0	1	1	10	22	111	467	1445	3366	5423
IAI1	0	1	1	5	29	121	484	1442	3428	5511
K12-DH10B	0	3	1	6	23	98	457	1475	3504	5567
K12-W3110	0	3	1	6	25	100	458	1467	3471	5531
HS	0	0	1	7	24	121	480	1439	3429	5501
K12-MG1655	0	3	1	6	25	97	463	1455	3473	5523
DH1	0	3	1	6	25	97	458	1453	3447	5490
B-REL606	0	3	2	5	24	99	511	1472	3389	5505
BW2952	0	3	1	7	25	97	453	1447	3421	5454
BL21-Gold-DE3	0	2	1	5	25	98	497	1460	3388	5476
BL21-DE3	0	2	1	5	25	100	497	1461	3362	5453
BL21-DE3-BL21	0	2	1	5	25	100	497	1461	3370	5461

Table 2.8: Statistics for the number of multigene start sites after applying the closure procedure to the dataset of 41 *E. coli* strains.

gene, *htgA* (synonym *htpY*), is involved in heat shock response. The possible explanation for the atypical start codon (CTG) was discussed by Missiakas et al. (1993). Using CAMBer, we identified 7 strains which annotated this gene with a different TIS. Six of them have annotated 495 nucleotides as gene length and one 486. In both cases, GTG was selected as the start codon. It is possible that some other start codons may also be used in *E. coli* (Blattner et al., 1997).

In this case study the maximal number of TIS in a multigene is 9; see Table 2.8 for more details. Interestingly, it is less than for *S. aureus*— the medium-size

case study; see Table 2.5.

connected components	before refinement	after refinement
all	13973	20257
core	3089	3084
anchors	12797	19694
orphans	3637	8380
non-anchors	1176	563
core anchors	2963	2979

Table 2.9: Statistics for the number of connected components with respect to their types, before and after the refinement procedure applied to the dataset of 41 *E. coli* strains.

Table 2.9 presents statistics of the refinement procedure. After the closure procedure we obtained 1176 non-anchors, of which we were able to split 934 using the refinement procedure, 689 of them we resolved completely into anchors. The refinement procedure produced only two new anchors with multigenes in all strains. Most of the connected components obtained were small, in particular, the number of orphans doubled; see Figure 2.16.

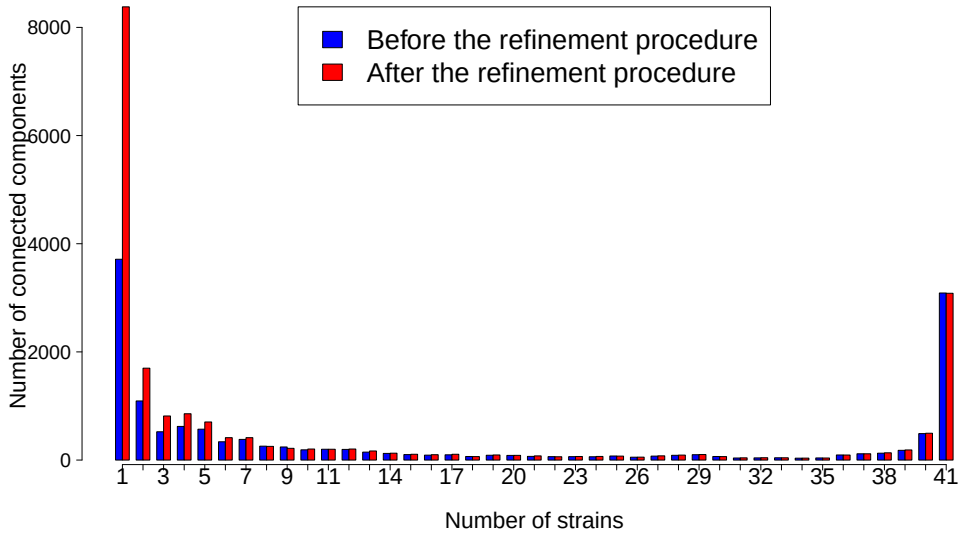

Figure 2.16: Histogram of the number of connected components (y-axis) shared by a particular number of strains (x-axis).

Core genome vs. pangenome

Finally, we computed core genome and pangenome for the family of *E. coli* strains using our concept of a multigene and compared the result to the core genome

and pangenome computed along the lines described in the work of Lukjancenko et al. (2010), where the authors considered 61 strains, many of them not having the sequencing status of "completed". Our set of strains is not a subset of the 61 strains mentioned above since there were some newly published strains (e.g., *E. coliUM146*, published in January 2011). For this reason, we had to repeat the computations for our set of strains.

Following the work of Lukjancenko et al. (2010), we call two genes homologous if the percent of identity is at least 50% covering at least 50% of the longer gene. We order all strains with respect to decreasing size of their genomes. We start with the strain having the largest genome, initializing both the pangenome and the core genome equal to the set of all genes of that strain. In the n-th step, we put a gene of the n-th strain into the pangenome if it is not homologous toany of the genes of the previously considered strains. We also remove a gene from the core genome when it not homologous to any of the genes of the n-th strain.

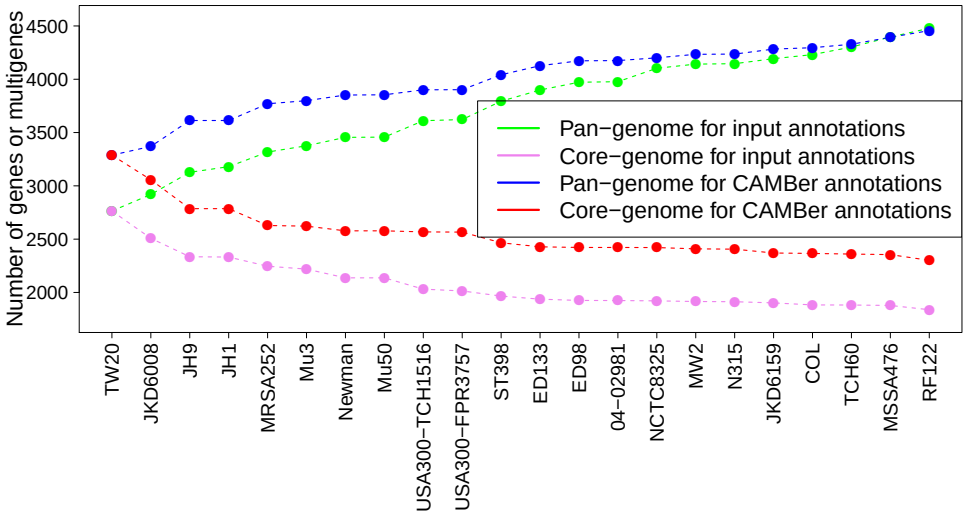

Figure 2.17: Core vs. pangenome plots of 22 *S. aureus* strains calculated using original annotations and multigene annotations, predicted by CAMBer. Strains are sorted (from left to right) in descending order of their genome sizes. Violet and green and points indicate cumulative numbers of core and pangenome sizes using annotated genes, while red and blue and points indicate cumulative numbers of core and pangenome sizes using multigenes after the closure procedure. The proportion of core genome to pangenome size has risen from 42% to 52% after the closure procedure.

We run two experiments on our set of strains: one which relies on the original genome annotations, as it was done in the work of Lukjancenko et al. (2010),

and another one which relies on previously pre-computed multigene annotations. Figure 2.18 shows the dynamics of change in gene numbers both for pangenome and core genome. It shows that as the number of strains increases both methods asymptotically converge to a pangenome size of around 13 000 genes. This suggests that the notion of a pangenome is quite robust when considering a large number of strains. On the other hand, there is a consistent difference between sizes of the core genome computed for the original annotations vs. pre- computed multigene annotations. For the latter method the core genome is substantially larger than for the former, resulting in an increase of the percentage with respect to pangenome from 18% to 25%. The analogous percentage for the 61 strains considered by Lukjancenko et al. (2010) was reported in that work as only 6%, but the computation was relying on original annotations.

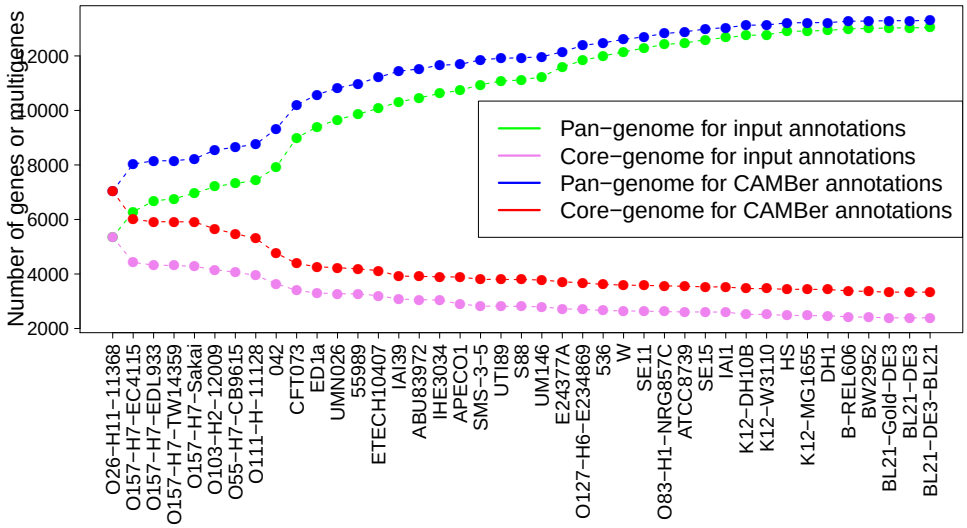

Figure 2.18: Core vs. pangenome plots for 41 *E. coli* strains calculated using original annotations and multigene annotations, predicted by CAMBer. Strains are sorted (from left to right) in descending order of their genome sizes. Violet and green and points indicate cumulative numbers of core and pangenome sizes using annotated genes, while red and blue and points indicate cumulative numbers of core and pangenome sizes using multigenes after the closure procedure. The proportion of core genome to pangenome size has risen from 18% to 25% after switching to multigene annotations resulting from CAMBer.

We also performed the analogous computations for *M. tuberculosis* and *S. aureus*. Figures 2.19 and 2.17 present results for *M. tuberculosis* and *S. aureus*, respectively. The conclusions are similar as for *E. coli*. The size of pangenome

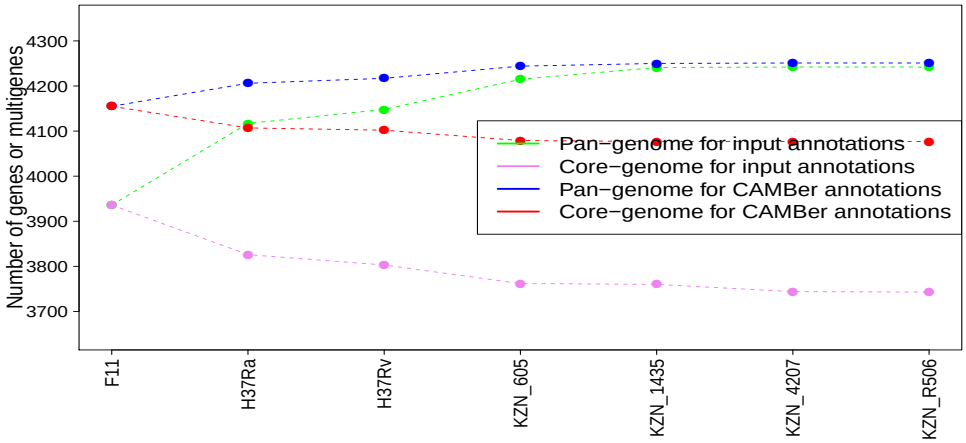

Figure 2.19: Core vs. pangenome plots of 9 *M. tuberculosis* strains calculated using original annotations and multigene annotations, predicted by CAMBer. Strains are sorted (from left to right) in descending order of their genome sizes. Violet and green and points indicate cumulative numbers of core and pangenome sizes using annotated genes, while red and blue and points indicate cumulative numbers of core and pangenome sizes using multigenes after the closure procedure. Here, the strain *KZN V2475* was excluded due to wrong annotation, caused by a shift in gene coordinates. The proportion of core genome to pangenome size has risen from 88.5% to 96.1% after switching to multigene annotations resulting from CAMBer.

computed using both methods converges, as the number of considered strains increases. On the other hand, size of the core genome shows a consistent difference for both methods. As a result, the proportion of the size of core genome with respect to the pangenome substantially depends on the chosen method, yielding higher score for the method based on pre-computed multigene annotations. The increase is from 42% to 52% for *S. aureus* and from 88% to 96% for the *M. tuberculosis* dataset.

2.5.2 Results for eCAMBer

In this section we present the results of our experiments, which demonstrate that: (i) eCAMBer is much more efficient than CAMBer, Mugsy-Annotator and the GMV pipeline; (ii) it scales well to large datasets; (iii) it improves annotation consistency; (iv) it improves annotation accuracy; and (v) eCAMBer outperforms Mugsy-Annotator and the GMV pipeline in terms of accuracy.

2.5.2.1 Comparison of running times

First, we compare the efficiency of eCAMBer and CAMBer by running the closure procedure for both tools on four datasets from our previous work on CAMBer (Woźniak et al., 2011a). All computations in the first experiment were performed on the same desktop machine with 4 processor cores being used. In this experiment, eCAMBer significantly outperforms CAMBer (Table 2.10). For example, the running time of the closure procedure on 9 strains of *M. tuberculosis* was reduced from about 1 hour 27 minutes to only 41 seconds.

	CAMBer		eCAMBer	
Dataset	**BLASTs**	**closure**	**BLASTs**	**closure**
2 strains of S. aureus	1 m 47 s	2 m 5 s	8 s	18 s
9 strains of M. tuberculosis	1 h 22 m	1 h 27 m	27 s	41 s
22 strains of S. aureus	6 h	6.5 h	3 m 15 s	4 m
41 strains of E. coli	42 h	48.5 h	22 m	25 m

Table 2.10: Comparison of running times between eCAMBer and CAMBer on four datasets considered in our precious work on CAMBer. Here, all the computations were performed on the same desktop machine with 4 processor cores being used.

Second, we also compare the running time of eCAMBer against CAMBer, Mugsy-Annotator and the GMV pipeline by running them on the four datasets from our previous work on CAMBer (Woźniak et al., 2011a). Since Mugsy-Annotator does not support multi-thread processing, in this experiment we use only one processor core for the computations. Table 2.11 presents running times in this experiment. It is clear from this table that the running time speedup achieved by eCAMBer is much more pronounced for larger datasets. This is an expected phenomenon since the other tools have quadratic running times with respect to the number of strains included.

The above results also suggest that eCAMBer scales well to larger datasets.

2.5.2.2 Large case studies

We examine the scalability of eCAMBer to large datasets by running it on 10 datasets for the 10 species with the highest number of sequenced strains in the PATRIC database (Gillespie et al., 2011), in the 16 March 2013 release. All datasets consist of genome sequences and annotations for the sets of strains

Dataset	CAMBer	eCAMBer	Mugsy-Ann.	GMV
2 strains of S. aureus	7 m 31 s	26 s	2 m	21 m
9 strains of M. tuberculosis	4 h 12 m	2 m 37 s	1 h 25 m	13 h 53 m
22 strains of S. aureus	37 h 5 m	16 m 30 s	4 h 11 m	28 h 36 m
41 strains of E. coli	273 h 22 m	1 h 48 m	19 h 21 m	368 h 31 m

Table 2.11: Comparison of running times between eCAMBer, CAMBer, Mugsy-Annotator and the GMV pipeline on four datasets from our previous work on CAMBer. All computations were executed on a machine with 1 processor core being used. The machine used in this computational experiment was different than the one used in the previous experiment. Columns correspond, in left-to-right order, to: short detaset description, total time consumed by the closure procedure in CAMBer, total time consumed by eCAMBer, total time consumed by Mugsy-Annotator, total time consumed the GMV pipeline.

within the same species. Experiments for all of these datasets were conducted on a machine with 24 processor cores, out of which 20 were used.

Table 2.12 shows a distribution of running times of all procedures of eCAMBer. The reader may observe that the running times are not necessarily monotonically increasing with the number of strains. For example, the closure procedure computations for the dataset of 162 strains of *H. pylori* took longer than the larger dataset of 195 strains of *S. aureus*. This may be explained by the fact that the total number of distinct sequences for annotated genes in *S. aureus* (98562) is much smaller than in *H. pylori* (208790).

In order to further investigate the scalability of eCAMBer, we check how the number of distinct gene sequences increases, when more strains are included. For this experiment, we chose the largest dataset of 569 strains of *E. coli*. Next, we sorted all genomes from the smallest to the largest. The plots (Figure 2.5) present the number of annotated genes and the number of gene sequences in a cumulative manner. We observe that the total number of distinct sequences grows much slower than the total number of gene annotations, suggesting sub-linear growth of the number of distinct gene sequences. Thus, according to our theoretical considerations, the algorithm implemented in eCAMBer for computing the closure procedure is sub- quadratic with respect to the number of strains included.

This experiment also shows that the strategy applied in eCAMBer to work with unique ORF sequences, rather than ORF annotations, leads to a sequence consolidation graph that is significantly smaller than the corresponding ORF consolidation graph. For example, in the largest dataset for 569 strains of *E. coli*, there are about 12.4mln nodes (ORF annotations) and 2.8bln edges in the ORF

Detaset description				Running times				
Species name	**Strains**	**Genes**	**Distinct seq.**	**Closure**	**Graph**	**Refine.**	**TIS v.**	**Clean up**
E. coli	569	2923165	487141 (0.17)	12 h	59 m	2 h 51 m	14 m	10 m
S. enterica	293	1366439	244450 (0.18)	3 h 56 m	18 m	36 m	4 m	4 m
S. agalactiae	250	517648	56215 (0.11)	29 m	2 m	5 m	37 s	53 s
S. pneumoniae	238	529076	99578 (0.19)	2 h 29 m	5 m	9 m	1 m 30 s	1 m 10 s
S. aureus	195	523557	98562 (0.19)	1 h 7 m	3 m	4 m	1 m50 s	1 m
H. pylori	163	267302	208790 (0.78)	1 h 42 m	12 m	5 m	5 m 10 s	2 m 10 s
L. interrogans	139	649916	175899 (0.27)	1 h 30 m	4 m	7 m	1 m 30 s	1 m 50 s
V. cholerae	130	467413	97258 (0.21)	24 m	2 m	2 m 20 s	35 s	51 s
A. baumannii	131	487775	129089 (0.27)	34 m	3 m	2 m 30 s	52 s	58 s
B. cereus	104	602986	395477 (0.66)	1 h 13 m	6 m	3 m 50 s	2 m 57 s	1 m 52 s

Table 2.12: Running times of eCAMBer on the 10 large datasets. All experiments were performed on the same machine with 24 processor cores, where 20 of them were used. The columns correspond in left-to-right order to: the species name, the number of sequenced strains within the species, the total number of annotated genes, the number of distinct sequences for the set of annotated genes (in the brackets we also provide the ratio between the number of distinct sequences to the total number of annotated genes), running time to compute all BLASTs for the closure procedure, total running time to compute the closure procedure (including BLAST computations), the running time to construct the sequence consolidation graph, the running time to compute the refinement procedure, the running time for the TIS voting procedure, and the running time for the clean up procedure.

consolidation graph, whereas there are only about 1.6mln nodes (unique ORF sequences), 1.3mln shared-end edges, and 55.9mln BLAST-hit edges in the sequence consolidation graph.

2.5.2.3 Annotation consistency

We also investigate ability of eCAMBer to identify annotation inconsistencies and to improve the consistency of annotations. As a case-study, we use the set of 20 *E. coli* strains with manually curated annotations, deposited in the ColiScope database (Touchon et al., 2009), available through the web-based interface MaGe (Vallenet et al., 2006). Pseudogenes were excluded from the analysis. On this dataset we run the closure procedure, followed by: the refinement procedure, the TIS voting procedure, and the clean up procedure. For comparison we also include annotations for the same set of strains, but downloaded from the PATRIC database (Gillespie et al., 2011).

In order to assess the improvement of annotation consistency, after running eCAMBer, we calculated the mean absolute difference in the number of annotated multigenes between two neighbour strains. It is 311 for the original annotations from ColiScope vs. 159 after applying eCAMBer. Analogous statistics on the

dataset from PATRIC are 409 for the original annotations and 311 after applying eCAMBer.

In the dataset of 20 *E. coli* strains from ColiScope database, after the closure procedure, eCAMBer identifies 73 gene families which have the following property: each family has a member in every strain, and for each family exactly one strain has a missing original annotation in that family. The top three strains with the highest number of missing gene annotations of that type are: *Sd197* (13), *2a 2457T* (8) and *536* (7). The most well-studied strain *K-12 MG1655* has four missing annotations of the above described type. These annotations were added by eCAMBer during the closure procedure.

Based on this case-study, we also investigate how eCAMBer improves consistency of TISs. There are 8038 pairs of originally annotated genes with different TISs, but with identical sequence (including 100bp. upstream region from the TIS of the longer annotation). This number was reduced to 4230 after applying the TIS majority voting procedure and the clean up procedure.

This case study also shows that inconsistencies, which come from annotation errors, are present even for a very well-studied bacterial organism like *E. coli*. Note also that the discussed annotation inconsistencies were identified among strains with annotations curated by the same laboratory.

2.5.2.4 Comparison of other features

CAMBer, eCAMBer, Mugsy-Annotator and the GMV pipeline aim to improve annotation consistency and accuracy. But there are some important differences between these approaches and their features (Table 2.13). For example, CAMBer and Mugsy-Annotator require gene annotations to be provided, whereas the GMV pipeline generates the input annotations using Prodigal and there is no straightforward way to substitute these annotations with any other. Thus, in all computational experiments involving the GMV pipeline were run only on Prodigal annotations. eCAMBer also integrates Prodigal as a tool to generate input annotations; however, it also allows the user to provide any other annotations as the input. All the tools require genome sequences at the input.

Different tools also aim at solving different annotation problems. For example, the GMV pipeline only identifies and solves TIS annotation inconsistencies, whereas Mugsy-Annotator also tries to identify missing genes. Our new tool,

	CAMBer	eCAMBer	Mugsy-Annotator	GMV
Input data	GS, GA	GS, optional GA	GS, GA	GS
Mapping of similar sequences	BLAST	BLAST	Multiple WGA	BLAST
Detection of gene presence inconsistencies	Yes	Yes	Yes	No
Detection of gene start inconsistencies	Yes	Yes	Yes	Yes
Correction of gene presence annotations	No	Yes (add. and rem.)	Yes (only add.)	No
Correction of gene start annotations	No	Yes	Yes	Yes
Multithreading	Partial	Yes	No	Partial

Table 2.13: Qualitative comparison of different tools. Columns correspond to the tools, whereas rows correpond to different qualitative features of these tools. Acronyms "GS" and "GA" denote genome sequences and genome annotations, respectively. Acronym "WGA" stands for whole genome alignment. Both CAMBer and the GMV pipeline have partial support for multithreading computations since only BLAST computations can be executed in parallel.

eCAMBer, is capable of resolving TIS inconsistencies, as well as removal of over-annotated genes and addition of missing genes (Table 2.13). Our previous tool only identifies annotation inconsistencies, but it does not propose corrections.

Notably, Mugsy-Annotator, GMV pipeline, CAMBer and eCAMBer do not make any assumption about the reference strain. However, Mugsy-Annotator, GMV pipeline and CAMBer suffer from the quadratic time complexity with respect to the number of strains since they use pairwise all-against-all comparisons. However, unlike the other tools, eCAMBer avoids redundant BLAST queries. This strategy gives especially good results when working with highly similar genome sequences.

Support for multithreading is a valuable feature for computationally demanding problems. Thus, it should be noted that eCAMBer has the most comprehensive support for multithreading among the tools considered. It allows the use of multiple threads for each of its steps. The GMV pipeline and CAMBer support multithreading only for BLAST computations. Mugsy-Annotator does not support it (Table 2.13).

2.5.2.5 Evaluation of annotation accuracy

In order to evaluate accuracy of annotations produced by eCAMBer, Mugsy-Annotator and the GMV pipeline, we apply the tools to annotations produced by the automatic annotation pipeline in PATRIC (Gillespie et al., 2011) for the set of 20 *E. coli* strains with manually curated annotations in the ColiScope database (Touchon et al., 2009). As an alternative dataset of input annotations

for the same set of strains we use annotations generated using Prodigal (Hyatt et al., 2010).

In all our comparative experiments we run Mugsy-Annotator and the GMV pipeline with default parameters. It should also be mentioned that both Mugsy-Annotator and the GMV pipeline output lists of suggestions of changes to input annotations, rather than actually output the corrected annotations. We post-processed these proposed lists of changes to generate the output annotations used for the comparative experiments.

First, we assess the correctness of the changes introduced to the input annotations based on the dataset of gene annotations with experimental support available in the EcoGene 3 database, developed by (Zhou and Rudd, 2013). This dataset consists of 922 gene annotations for the *K-12 MG1655* strain. From this set we excluded four genes: *fdhF*, *prfB*, *rph'*, *insN'*; since their sequences corresponding to the annotated coordinates are disrupted (the length of the sequence from the start codon to the stop codon is not divisible by 3). Additionally, we ran one iteration of the eCAMBer closure procedure to transfer the set of 918 gene annotations on the remaining 19 strains. The transferred gene annotations share at least 80% of sequence identity with original annotations for strain *K-12 MG1655*.

Table 2.14 presents statistics for the TIS changes introduced by different tools compared against the dataset described above. There are three different scenarios: (i) a correct TIS annotation is changed to an incorrect one (orange); (ii) an incorrect TIS annotation is changed to another incorrect TIS (yellow); (iii) an incorrect TIS is changed to the correct one (green). Since for each gene, there is only one TIS annotation considered as correct, there is no possible change from one correct TIS to another one. For each strain the majority of TIS changes introduced by eCAMBer is correct. In this experiment eCAMBer made 89 TIS changes from incorrect to correct and only 12 TIS changes from correct to incorrect on the dataset of Prodigal annotations. For comparison, GMV made 47 incorrect-to-correct TIS changes and 8 correct-to-incorrect TIS changes, on the same dataset. Thus, the number of correct TIS annotations has increased by 77 in case of eCAMBer and by 39 in case of GMV. Application of Mugsy-Annotator made more wrong changes than correct.

Since the extended dataset of annotations from Ecogene 3 constitutes only

Statistic	PATRIC		Prodigal		
	MA	eCAMBer	GMV	MA	eCAMBer
# of incorrect→correct TIS changes	839	392	47	132	89
# of incorrect→incorrect TIS changes	215	50	5	96	8
# of correct→incorrect TIS changes	892	92	8	672	12

Table 2.14: Overall statistics for TIS changes introduced by eCAMBer, Mugsy-Annotator (MA) and the GMV pipeline. The tools were run on the dataset of 20 *E. coli* with annotations from the PATRIC database (columns 2 to 3) and generated using Prodigal (columns 4 to 6). Correctness of the changes introduced was assessed by comparison them against the set of experimentally verified gene annotations available in the EcoGene 3 database for the *K-12 MG1655* strain. Gold standard annotations for the remaining 19 strains were obtained by homology transfer of that set of 918 annotations. Statistic presented in this table include only that subset of genes which share the same stop codon as any of the genes in the gold standard.

about 20% of all genes in the 20 strains of *E. coli*, it is not sufficient for direct assessment of overall quality of changes introduced by eCAMBer and other tools. In particular, we cannot conclude if a gene annotation is correct or not based on its absence in this dataset (so that there is no gene annotations in the dataset sharing the same stop codon). Thus, we perform further assessment of the quality of changes introduced relying on manually curated annotations for the set of 20 *E. coli* strains in the ColiScope dataset (Touchon et al., 2009). It is a reasonable choice as a gold standard, since many of the annotations have experimental support. In particular, the annotation for the strain *K-12 MG1655* contains 901 out of 918 gene annotations present in the dataset described previously. For comparison, for this strain, there are only 841 and 883 such gene annotations for PATRIC and Prodigal, respectively.

Next, Figure 2.20 presents the assessment of TIS changes introduced during the TIS voting procedure based on the ColiScope dataset. It shows the assessment of the TIS changes introduced to the input PATRIC annotations, with respect to each of the 20 *E. coli* strains. Statistic presented in this figure distinguishes three different scenarios: (i) a correct TIS annotation is changed to an incorrect one (orange); (ii) an incorrect TIS annotation is changed to another incorrect TIS (yellow); (iii) an incorrect TIS is changed to the correct one (green). Since for each gene, there is only one TIS annotation considered as correct, there is no possible change from one correct TIS to another one. For each strain the majority

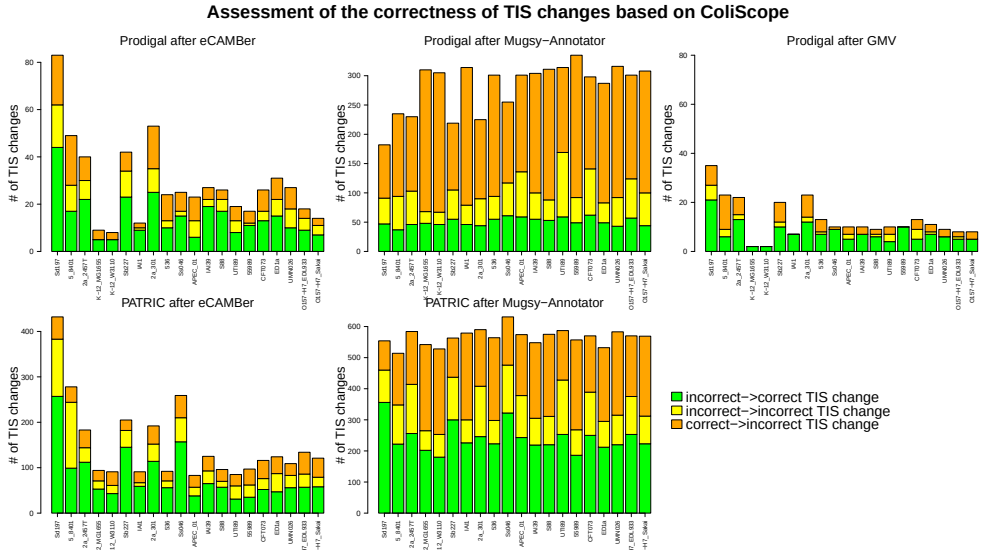

Figure 2.20: Impact of the TIS voting procedure of eCAMBer on annotations from the PATRIC database. Annotations from the ColiScope database were used to assess correctness of TIS changes. Note, that since for each gene, there is only one TIS annotation considered as correct, thus there is no possible change from one correct TIS to another one.

	PATRIC			Prodigal			
Statistic	**Input**	**MA**	**eCAMBer**	**Input**	**GMV**	**MA**	**eCAMBer**
# of incorrectly removed genes	NA	0	1224	NA	0	0	388
# of incorrectly added genes	NA	1177	792	NA	0	344	331
# of correctly removed genes	NA	0	3993	NA	0	0	1185
# of correctly added genes	NA	410	701	NA	0	210	1447
# of incorrect→correct TIS changes	NA	4812	1591	NA	149	1015	290
# of incorrect→incorrect TIS changes	NA	2223	747	NA	28	1018	113
# of correct→incorrect TIS changes	NA	4279	669	NA	78	3618	170
Precision for gene starts	0.665	0.663	0.699	0.764	0.764	0.734	0.775
Recall for gene starts	0.695	0.702	0.703	0.752	0.753	0.727	0.765
f1 for gene starts	0.680	0.682	0.701	0.758	0.759	0.731	0.770
Precision for gene ends	0.892	0.882	0.920	0.931	0.931	0.928	0.940
Recall for gene ends	0.931	0.935	0.926	0.917	0.917	0.919	0.927
f1 for gene ends	0.911	0.908	0.923	0.924	0.924	0.923	0.934

Table 2.15: Overall statistics for accuracy of changes introduced by eCAMBer, Mugsy-Annotator (MA) and the GMV pipeline. The tools were run on the dataset of 20 *E. coli* with annotations from the PATRIC database (columns 2 to 4) and generated using Prodigal (columns 5 to 8). Correctness of the changes introduced was assessed by comparison with annotations from the Coliscope database. Columns Input correspond to the original annotations. "NA" stands for not applicable. Rows correspond to different statistics of running each tool.

of TIS changes introduced by eCAMBer is correct.

Rows 5 to 8 of Table 2.15 summarize the overall impact of eCAMBer and

Mugsy-Annotator on TIS annotations. Remarkably, 70% (1591 out of 2260) of TIS changes introduced by eCAMBer to PATRIC annotations were correct. For comparison, only 43% of the TIS changes introduced by Mugsy-Annotator were correct.

Figure 2.21 presents the assessment of gene additions and removals introduced during the closure and the clean up procedures, respectively. It shows the assessment of the changes introduced to the input PATRIC annotations, with respect to each of the 20 *E. coli* strains. Statistic presented in this figure distinguishes four different scenarios: (i) a missing genome annotation is correctly added during the closure procedure (blue); (ii) a wrong gene annotation is correctly removed during the clean up procedure (green); (iii) a wrong gene annotation is incorrectly added during the closure procedure (red); and (iv) a correct gene annotation is incorrectly removed during the clean up procedure (orange). It can be seen that, for each strain, the majority of changes introduced by eCAMBer is correct.

The first four rows of Table 2.15 summarize the overall impact of eCAMBer and Mugsy-Annotator on gene presence. The results show that eCAMBer outperforms Mugsy-Annotator in this aspect. For example, 70% of the changes introduced by eCAMBer to PATRIC annotations were correct, whereas it was only 26% for Mugsy-Annotator.

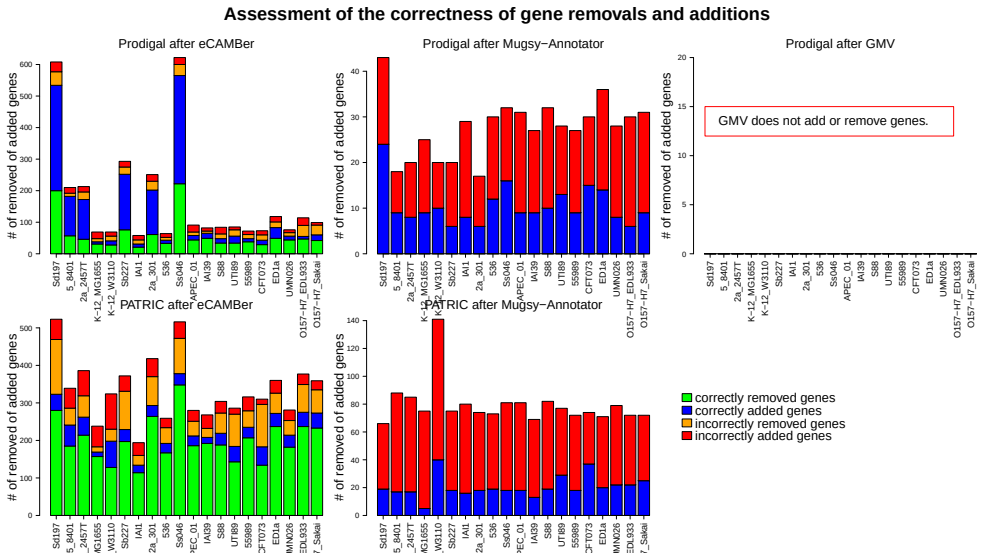

Figure 2.21: Impact of the closure and clean up procedures of eCAMBer on the annotations from the PATRIC database. Annotations from the ColiScope database were used to assess correctness of gene removals and additions introduced by eCAMBer.

Finally, we investigate how the whole pipelines implemented in eCAMBer, Mugsy-Annotator and GMV improve the overall annotation accuracy. Here, the accuracy is measured by f_1 statistic, defined as $2 \cdot \frac{precision \cdot recall}{precision + recall}$, where $precision = \frac{TP}{TP+FP}$ and $recall = \frac{TP}{TP+FN}$. Here, TP, FP and FN denote true positive, false positive and false negative prediction, respectively. Since a pair of gene annotations may have the same stop codon, but different TISs, we keep track on the results for both stop codon predictions and for the TIS predictions.

Results of eCAMBer on PATRIC annotations in this experiment are presented in Figure 2.22. Note that each correctly identified TIS determines also its correctly identified stop codon, but not the other way round. Thus, the accuracy for the TIS prediction is lower than for the stop codons. As the figure shows, eCAMBer improves annotation accuracy, for each strain, both in terms of TIS annotations and stop codon annotations.

Rows 9 and 12 of Table 2.15 summarize the change in accuracy when running different tools on PATRIC and Prodigal annotations. It is clear from this table that eCAMBer outperforms other tools. For example, eCAMBer increased the f1 statistic of initial annotations of Prodigal (for gene starts) from 0.764 to 0.775, whereas the application of GMV improved it only by 0.001 and the application of Mugsy-Annotator decreased it by 0.027. In the case of PATRIC annotations, application of Mugsy-Annotator improved the accuracy from 0.680 to 0.682. However, the accuracy of annotations after eCAMBer increased to 0.703.

2.6 Summary

In this chapter we presented our work and tools we have developed to support comparative analysis of multiple bacterial strains. In particular, in section 2.2 we presented details of CAMBer — the first tool we have developed to support comparative analysis of multiple bacterial strains. In section 2.3 we presented details on eCAMBer, which is a highly optimized version of CAMBer. Finally, in section 2.4 we presented CAMBerVis.

The presented results for CAMBer suggest, that it can be successfully used to improve consistency of the input annotations for small datasets.

However, despite its usefulness on small datasets, the main drawback of CAMBer is efficiency. Similarly as Mugsy-Annotator and the GMV pipeline, it suffers

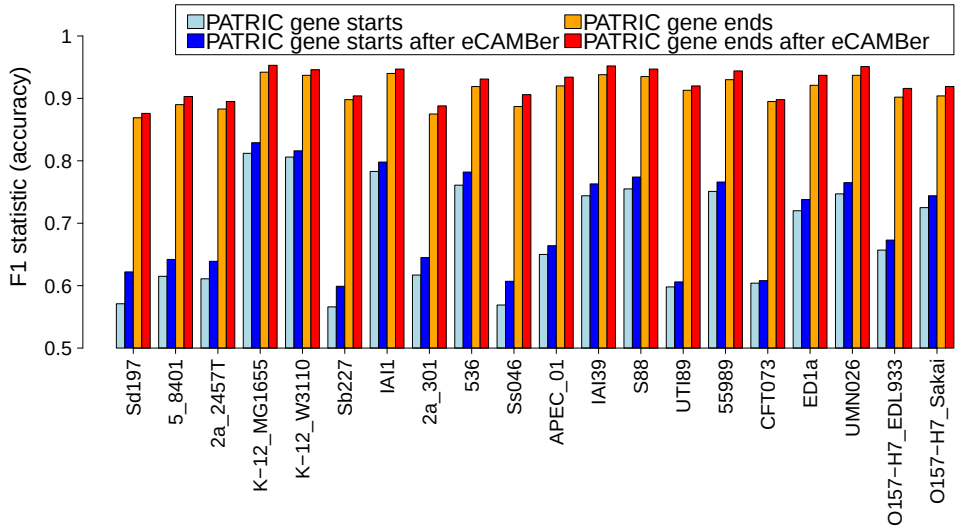

Figure 2.22: Comparison of annotation accuracy before and after applying eCAMBer on the dataset of 20 *E. coli* strains with annotations from PATRIC. Manually curated annotations from ColiScope were used as a gold standard.

from the quadratic time complexity with respect to the number of strains. This makes the tools rather unusable for datasets comprising hundreds of bacterial strains.

This was the main problem we addressed by developing eCAMBer. The underlying idea behind the efficient implementation of the procedure is to avoid redundant BLAST queries. This approach greatly reduces the computational complexity, thus leading to much shorter running time than other tools.

For example, on the dataset of 41 strains of *E. coli*, computations took less than two hours (using only one processing thread), whereas Mugsy-Annotator (the fastest competitor) took more than 19 hours. Moreover, eCAMBer supports multithreading for all its procedures. This allows eCAMBer to be used on much larger datasets comprising hundreds of bacterial strains. A dramatic speed up offered by eCAMBer can be seen when working with a large number of bacterial strains. The running time is reduced (for 41 strains of *E. coli*) from 2 days, in the case of CAMBer, to less than half an hour, in the case of eCAMBer.

Since the approach of avoiding redundant BLAST queries for identical gene sequences turned out so useful, it might also be fruitful to investigate the idea of relaxing the condition of avoiding redundancy from identical sequences to just highly similar sequences. This approach could further lower the running time

of the tool without significant drop in accuracy. However, we leave this as a potential future direction of research.

Furthermore, eCAMBer tries to resolve annotation inconsistencies in order to produce more accurate annotations. For this purpose, it implements a majority voting-like approach for selecting the most reliable TISs and implements a procedure for identification and removal of gene families which are likely to be propagated annotation errors.

The presented results show, that eCAMBer outperforms its competitors, Mugsy-Annotator and the GMV pipeline, in terms of improving quality of annotations. In particular, when run on genome annotations generated by Prodigal for the set of 20 *E. coli* strains, eCAMBer increased the f1 statistic of initial annotations from 0.764 to 0.775, whereas the application of GMV improved it only by 0.001 and the application of Mugsy-Annotator even decreased it.

Of course, eCAMBer also has some limitations. One is that it purely relies on the quality of original annotations. Thus, for example, eCAMBer cannot identify genes, whose annotations are missing for all strains. Another limitation of eCAMBer is that pseudogenes and non-protein coding genes are excluded from the analysis. This follows from the assumption that eCAMBer considers only genes of length divisible by 3, start with start codon, and end with stop codon.

Lastly, the tools we developed to improve the overall accuracy and consistency of bacterial genome annotations are of general applicability and can be used for other purposes than studying drug resistance.

The tools, case-study input data and the obtained results are available at the website of this project, `http://bioputer.mimuw.edu.pl/ecamber`.

"Owing to this struggle for life, any variation, however slight and from whatever cause proceeding, if it be in any degree profitable to an individual of any species, in its infinitely complex relations to other organic beings and to external nature, will tend to the preservation of that individual, and will generally be inherited by its offspring. (...) I have called this principle, by which each slight variation, if useful, is preserved, by the term of Natural Selection, in order to mark its relation to man's power of selection."

Charles Darwin, On the Origin of Species, 1859

3

Drug resistance-associated mutations

In this chapter we present our work on identifying drug resistance-associated mutations based on comparative analysis of whole-genome sequences of closely related bacterial strains. In particular, we present GWAMAR, the tool we have developed to support this type of analysis. In section 3.1, we describe the idea behind our approach and review some related work. We also introduce the basic concepts and notations. In section 3.2, we describe in detail the methodology of GWAMAR. Notably, it uses eCAMBer, described in chapter 2, for identification of genetic variations (mutations) among the set of considered strains, which constitute the genotype data. As a part of this section, we also present *weighted support* (WS) and *tree-generalized hypergeometric* (TGH) score — two statistics we propose for identifying of drug resistance associations. Additionally, we propose a Rank-based metascore (RBM) for combining multiple scores into one in order to compromise between different approaches used to define different scores. In section 3.3, we present and discuss results obtained by applying GWAMAR to three datasets — one for *S. aureus* and two for *M. tuberculosis*. The presented results show that GWAMAR can be successfully used for identification of drug resistance-associated mutations.

3.1 Introduction

Genome-Wide Association Studies (GWAS) have been successfully applied to associate human mutations with phenotype of various human diseases and traits (Manolio, 2010; Stadler et al., 2010; Davies et al., 2011).

The recent progress in genome-sequencing technologies, continuously decreasing the cost of sequencing of bacterial genomes (Loman et al., 2012), enables the use of similar approaches for genotype-phenotype mapping in bacteria.

The potential of the use of whole-genome comparative approaches to study drug resistance and host-pathogen interactions in bacteria has been recently proposed (Khor and Hibberd, 2012; Read and Massey, 2014).

3.1.1 Genotype data

The input genotype data for these studies usually comes from in-house sequencing, rather than publicly available data. This might be caused by the problematic use of the publicly available data. First, as we noticed in the previous chapter, the inconsistent and poor-quality annotations of publicly available strains may complicate that analysis. Second, the phenotype data with respect to drug susceptibility tests are spread throughout the literature and are not easy to collect.

In the previous chapter of this work, we presented CAMBer and eCAMBer — the tools to support comparative analysis of multiple bacterial strains — thus addressing the first issue. In order to overcome the second issue, we have to perform a careful search of the literature for results of drug susceptibility tests of the strains considered.

3.1.2 Phenotype data

Minimum Inhibitory Concentration (MIC) is the most commonly used measure to quantify drug resistance in bacteria. It is the lowest concentration of an antibiotic which inhibits visible growth of a colony of bacteria after overnight incubation. The detailed guidelines for the procedure of drug susceptibility testing are published by bodies such as Clinical and Laboratory Standards Institute (CLSI), British Society for Antimicrobial Chemotherapy (BSAC), and The European Committee on Antimicrobial Susceptibility Testing (EUCAST). The guidelines also contain information on MIC breakpoints to assign drug resistance or drug

susceptibility. Sometimes also the third class of intermediate resistance is distinguished.

Most of the sources reporting results of drug susceptibility testing provide only information on the outcome status, rather than particular MIC values. Thus, in our study we use only three classes of drug resistance: drug susceptible, intermediate drug resistant and drug resistant.

For the purpose of this work, we have collected the phenotype data for drug resistance from the following sources: (i) publications issued together with the fully sequenced genomes; (ii) NARSA project (`http://www.narsa.net`); (iii) email exchange with the authors of some publications; and (iv) other publications found by searching of the related literature.

3.1.3 Gold standard associations

One problem we faced during the project was caused by the relatively small number of positive associations available in the databases, which would constitute the gold standard data to assess the accuracy of our method.

Nevertheless, there are known genes and point mutations responsible for some of the drug resistance mechanisms. However, these are spread over various studies and are therefore not easy to gather.

One attempt to collect the information on genetic changes associated with drug resistance into a database is the Antibiotic Drug Resistance Database (ARDB) developed by Liu and Pop (2009). However, this database focuses on genes associated with drug resistance rather than particular point mutations within them. We use data available in this database as our gold standard for the case study on the *S. aureus* dataset, presented in the results section of the chapter.

Another species-specific database of drug resistance-associated mutations in *M. tuberculosis* is the Tuberculosis Drug Resistance Mutation Database (TBDReaMDB) developed by Sandgren et al. (2009). This database provides detailed information on a set of 1230 associations between drugs and point mutations. Furthermore, it distinguishes a subset of *high-confidence* mutations which were often reported in the literature. We use data available in this database as our gold standard for the case study on the two *M. tuberculosis* datasets, presented in the results section of the chapter.

3.1.4 Phylogenetic information

In this work we investigate the potential of the use of phylogenetic information in identifying drug resistance-associated mutations. In particular, we propose two association scores, called TGH and WS, based on the phylogenetic information.

The rationale for our approach is based on two known phenomena. First, the bacteria isolated from close-distance locations of each other tend to have similar genome sequences. As a result, subtrees of the phylogenetic trees tend to correspond to geographic locations (Daubin et al., 2003).

Second, although the phenomenon of genomic convergence is unlikely in general, it is rather common in case of mutations which are subject to evolutionary pressure caused by drug treatment (Hazbón et al., 2008; Farhat et al., 2013). Thus, drug resistance-associated mutations tend to be independent of geographic location and therefore more widely distributed over the tree, as opposed to mutations driven by other environmental factors which tend to concentrate in small subtrees.

Hence, mutations predicted to occur independently multiple times in the evolutionary history of the bacterial strains are more likely to be associated with drug resistance, rather than with other environmental factors (Hazbón et al., 2008). A conceptually similar approach has been taken by Dutheil (2012) to identify co-evolving mutations in protein sequences.

We note however, it is only an approximation to represent the evolutionary history of bacteria as a tree. It has been debated that, in the presence of HGT mechanisms in bacteria, their evolutionary history may be better represented as a network rather than a tree (Philippe and Douady, 2003). On the other hand, some estimations show that the effect of HGT on the overall evolution is limited and does not preclude the use of phylogenetic trees (Daubin et al., 2003; Boto, 2010). We leave the possibility of using other representations of the evolutionary history of bacteria as a subject of further research.

3.1.5 Basic definitions

In this work, we consider a set $\mathcal{S}$ of closely related bacterial genomes. Typically, this is a set of strains within the same species of bacteria.

Then, we represent the available drug resistance information as a set of *drug resistance profiles* $\mathcal{R}$, where each drug resistance profile $r \in \mathcal{R}$ is represented as

a vector:

$$r : \mathcal{S} \to \{\text{'S'}, \text{'I'}, \text{'R'}, \text{'?'}\}. \tag{3.1}$$

Here, 'S', 'I', 'R' denote that a given strain is known to be drug susceptible, intermediate-resistant, or resistant, respectively. We indicate, using question mark '?', that the drug resistance status of a strain is unknown. We call a drug resistance profile *complete* if it does not contain question marks.

The genotype data consists of a set of genetic mutations of three types:

- point mutations (in amino-acid sequences),
- gene gain/losses,
- promoter mutations.

In our approach we exclude synonymous SNPs as, according to our knowledge, there are no known examples of synonymous mutations associated with drug resistance.

Each mutation is represented as a piece of information adequate for the type of the mutation (such as gene identifier of the corresponding gene family) and a vector called *mutation profile*:

$$v : \mathcal{S} \to \Sigma. \tag{3.2}$$

Here, for each point mutation, we keep the information on its position in the multiple alignment of its corresponding gene family and the information on the gene family identifier. The mutation profile for each point mutation is determined based on its corresponding column in the multiple alignment. In that case $\Sigma = \Sigma_{AA}$ denotes the set of twenty amino acids. We also assume Σ_{AA} contains the '-', symbol for the gap in the corresponding multiple alignment and the '?' symbol if the gene sequence is unknown for a given strain. We take into account only columns which contain at least two different characters (ignoring '?').

Next, for each gene gain/loss, we keep the information on its corresponding gene family identifier. For such a mutation, its mutation profile is determined

based on the presence or absence of a gene in the corresponding gene family for a given strain. Thus, $\Sigma = \{\text{'L'}, \text{'G'}\}$, where $v(S) = \text{'L'}$ means that the gene is absent in strain S, whereas $v(S) = \text{'G'}$ means that the gene is present in strain S.

Finally, for each promoter mutation, we keep the information on its position in the multiple alignment of promoter sequences for the corresponding gene family and the information on the gene family identifier. The mutation profile for each promoter mutation is determined based on its corresponding column in the multiple alignment. In that case $\Sigma = \Sigma_{NT}$ denotes the set of four different nucleotides together with the '-' symbol for gaps in the corresponding multiple alignment and the '?' symbol if the gene promoter sequence is unknown for a given strain.

Analogously, we call a mutation profile *complete* if it does not contain question marks.

It should be noted that potentially multiple mutations (for example point mutations at different positions in the genome) may have identical mutation profiles. In that situation the mutations would essentially carry the same information about their mutation profiles. Thus, we also introduce an auxiliary concept called *binary mutation profile*. Let $S^* \in \mathcal{S}$ denote the reference strain and $S \in \mathcal{S}$ denote any strain. Then, for a given *mutation profile* v, its corresponding binary mutation profile

$$b_v : \mathcal{S} \to \{\text{'0'}, \text{'1'}, \text{'?'}\}, \tag{3.3}$$

is defined as follows:

$$b_v(S) = \begin{cases} \text{'?'} & \text{if } v(S) = \text{'?'} \\ \text{'0'} & \text{if } v(S) = v(S^*) \\ \text{'1'} & \text{otherwise} \end{cases} \tag{3.4}$$

Analogous to mutation profiles, we call a binary mutation profile *complete* if it does not contain question marks.

We say that a genetic change (mutation) m is *present* in strain $S \in \mathcal{S}$ if for its corresponding mutation profile v, $b_v(S) = \text{'1'}$; otherwise we say that the mutation m is *absent* in strain S.

3.1.6 Problem setting

Finally, we define the problem which we address here: given a list of mutations and a list of drug resistance profiles, produce an ordered list of associations between the phenotype and genotype data (represented as drug resistance and mutation profiles) such that the top-scored associations are the most likely to be real.

3.2 GWAMAR: Genome-wide assessment of mutations associated with drug resistance in bacteria

In this section, we present details of GWAMAR, the tool we have developed for genome-wide assessment of mutations associated with drug resistance. The presentation includes the preprocessing of input data; computation of the association scores and results obtained by applying the tool to datasets for *M. tuberculosis* and *S. aureus*.

3.2.1 The pipeline of GWAMAR

GWAMAR is designed as a pipeline. It first employs eCAMBer, the tool described in the previous chapter, to perform three preliminary steps: (i) downloading of genome sequences and annotations for the set of multiple bacterial strains in question, (ii) consolidation of the genome annotations, (iii) identification of homologous gene families; see Figure 3.1.

In the next step eCAMBer identifies the set of genetic variations and represents them as mutations profiles. As described in section 3.1.5, three types of mutations are considered: (i) point mutations in amino-acid sequences, (ii) point mutations in promoter regions (-50bp downstream the corresponding TIS), (iii) gene gain/losses.

Here, each gene gain/loss mutation profile is determined based on the presence/absence of elements of the corresponding gene family among the strains.

For each identified gene family, eCAMBer employs MUSCLE (Edgar, 2004), to compute its multiple sequence alignment for the set of corresponding amino-acid sequences. Similarly, it uses MUSCLE to compute a multiple sequence alignment for the set of corresponding promoter sequences.

Next, eCAMBer transforms each column in the computed multiple alignment into a mutation profile, as long as at least one character in that column differs (there is a mutation present); see Figure 3.1.

Also, eCAMBer supports use of PHYLIP (Felsenstein, 2005) and PhyML (Guindon et al., 2010) — the software for reconstruction of the phylogenetic tree based on the maximal-likelihood approach.

In the next step, for the selected reference strain, GWAMAR computes binary mutation profiles for each mutation profile, based on formula 3.4. Since multiple mutation profiles may correspond to a binary mutation profile, this step significantly reduces the number of pairs of profiles (resistance and mutation profiles) to be scored.

Finally, GWAMAR computes several statistical scores to associate drug resistance profiles to the mutation profiles, including mutual information (MI), odds ratio (OR), hypergeometric (H) score, weighted support (WS), and the tree-generalized hypergeometric (TGH) score. Additionally, it implements a score we called Rank-based metascore (RBM) which for combining multiple scores into one in order to compensate for weaknesses of different individual scores.

Figure 3.1 illustrates the overall data-processing flow implemented in GWAMAR.

3.2.2 Association scores

Here we present the association scores implemented in GWAMAR to score pairs of binary mutation and drug-resistance profiles. These scores include statistics commonly used in various associations studies, such as mutual information (Wu et al., 2012), odds ratio (Clarke et al., 2011), hypergeometric test (Cabrera et al., 2012). It also computes weighted support and tree-generalized hypergeometric score — the newly proposed statistics to incorporate the phylogenetic information. Moreover, it implements the Rank-based metascore for combining multiple scores into one.

For a given binary mutation profile $b\mathcal{B}$ and a given drug resistance profile $r\mathcal{R}$, we introduce the following auxiliary notations:

- $\mathcal{S}_1^R = \{S \in \mathcal{S} : b(S) = \text{'1'} \wedge r(S) = \text{'R'}\}$,
- $\mathcal{S}_0^R = \{S \in \mathcal{S} : b(S) = \text{'0'} \wedge r(S) = \text{'R'}\}$,

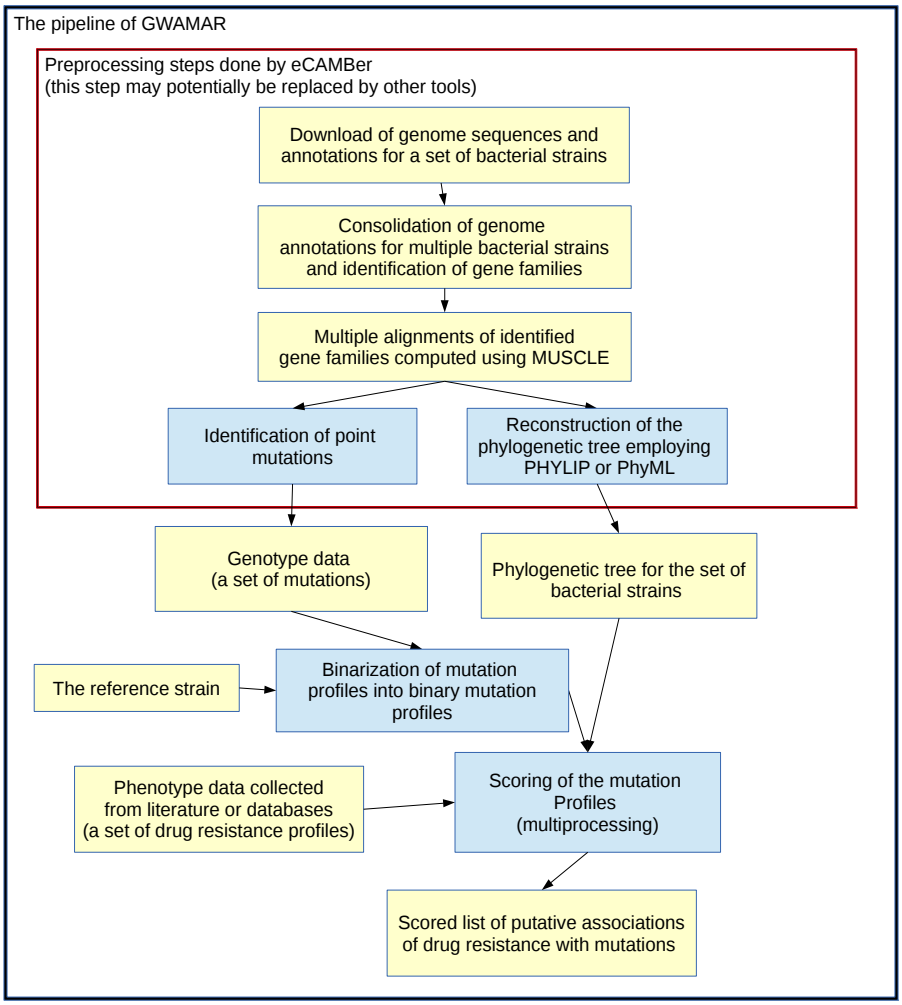

Figure 3.1: Schema of the pipeline of GWAMAR. For a set of considered bacterial strains, the input data for GWAMAR consists of (i) a set of mutations; (ii) a set of drug resistance profiles; and (iii) optional, phylogenetic tree for the set of bacterial strains. Typically the set of mutation profiles is generated using eCAMBer, which is able to download the genome sequences and annotations for the set of bacterial strains, identify point mutations based on multiple alignments, and reconstruct the phylogenetic tree of the considered bacterial strains. Assuming the genotype data is preprocessed, the first step of GWAMAR is to compute binary mutation profiles for all the mutations. This step significantly reduces the number of profiles considered. Finally, GWAMAR implements several statistical scores to associate drug resistance profiles with mutation profiles. These include: mutual information, odds ratio, hypergeometric score, weighted support, tree-generalized hypergeometric and the Rank-based metascore. As a result, we obtain ordered lists of drug resistance associations, where the top-scored associations are the most likely to be real.

- $\mathcal{S}_1^I = \{S \in \mathcal{S} : b(S) = \text{'1'} \wedge r(S) = \text{'I'}\}$,

- $\mathcal{S}_0^I = \{S \in \mathcal{S} : b(S) = \text{'0'} \wedge r(S) = \text{'I'}\}$,
- $\mathcal{S}_1^S = \{S \in \mathcal{S} : b(S) = \text{'1'} \wedge r(S) = \text{'S'}\}$,
- $\mathcal{S}_0^S = \{S \in \mathcal{S} : b(S) = \text{'0'} \wedge r(S) = \text{'S'}\}$,
- $\mathcal{S}^S = \{S \in \mathcal{S} : r(S) = \text{'S'}\}$,
- $\mathcal{S}^R = \{S \in \mathcal{S} : r(S) = \text{'R'}\}$.
- $\mathcal{S}^I = \{S \in \mathcal{S} : r(S) = \text{'I'}\}$.
- $\mathcal{S}_0 = \{S \in \mathcal{S} : b(S) = \text{'0'}\}$,
- $\mathcal{S}_1 = \{S \in \mathcal{S} : b(S) = \text{'1'}\}$.

Note that, instead of using mutation profiles, we use binary mutation profiles.

3.2.2.1 Odds ratio

For a given binary mutation profile b and drug resistance profile r, we calculate *odds ratio* (OR) score using the following formula:

$$\text{OR}(b, r) = \frac{|\mathcal{S}_1^R| \cdot |\mathcal{S}_0^S|}{max(1, |\mathcal{S}_0^R|) \cdot max(1, |\mathcal{S}_1^S|)} \tag{3.5}$$

Here, we use the *max* function in the denominator to ensure there is no problem with divisibility by 0.

3.2.2.2 Mutual information

For a given binary mutation profile b and a given drug resistance profile r, we calculate *mutual information* (MI) score using the following formula:

$$\text{MI}(b, r) = \sum_{x \in \{\text{'0'},\text{'1'}\}} \sum_{y \in \{\text{'S'},\text{'I'},\text{'R'}\}} \frac{|\mathcal{S}_x^y|}{|\mathcal{S}|} \cdot log(\frac{|\mathcal{S}_x^y| \cdot |\mathcal{S}|}{|\mathcal{S}_x| \cdot |\mathcal{S}^y|}) \tag{3.6}$$

3.2.2.3 Hypergeometric score

For a given binary mutation profile b and a given drug resistance profile r, we calculate hypergeometric (H) score using the following formula:

$$\mathrm{H}(b,r) = -log\Big(\sum_{i=|\mathcal{S}^R|}^{|\mathcal{S}|} H(|\mathcal{S}|, |\mathcal{S}^R|, |\mathcal{S}_1|, i)\Big) \tag{3.7}$$

where:

$$H(N,K,n,k) = \frac{\binom{K}{k} \cdot \binom{N-K}{n-k}}{\binom{N}{n}} \tag{3.8}$$

Here, we define the hypergeometric score as a minus logarithm of the value typically used in the definition of the hypergeometric test. We use this approach in order to have consistent property for all considered scoring methods, such that the higher the score the more likely drug resistance profile is associated with binary mutation profile.

3.2.2.4 Support

For a given binary mutation b and a given drug resistance profile r, we define *support* (S) as the number of drug-resistant strains with the mutation present minus the number of drug-susceptible strains with the mutation present:

$$\mathrm{S}(b,r) = |\mathcal{S}_1^R| - \alpha(r)|\mathcal{S}_1^S|, \tag{3.9}$$

where:

$$\alpha(r) = \frac{|\mathcal{S}^R|}{|\mathcal{S}^S|} \tag{3.10}$$

Here $\alpha(r)$ is a weight which we use to punish mutations for their presence in drug-susceptible strains. It is defined as the proportion of the number of drug-resistant to the number of drug-susceptible strains, so that occurrences of a mutation are given equal emphasis in drug-resistant and drug-susceptible strains.

3.2.2.5 Weighted support

Although the support is a simple and intuitive score, it does not incorporate any phylogenetic information. For example, let us assume there are two point mutations with the same support 3, where the first mutation covers only drug-resistant strains within one subtree of the phylogenetic tree, whereas the second mutation covers the same number of strains but spread throughout the whole tree. The first mutation is likely to be associated with the phylogeny, driven by some environmental changes. This suggests that the second mutation should have a greater score as it has to be acquired a few times independently during the evolution process.

We propose weighted support (WS) as a score to account for the above situation. For a given phylogenetic tree T, drug resistance profile b, and binary mutation profile b, WS is defined as follows:

$$\mathrm{WS}_T(b, r) = \sum_{S \in \mathcal{S}} w_T(b, r, S)[b(S) = \text{'1'}] \tag{3.11}$$

where $w_T(b, r, S)$ is a weight assigned to each cell in a given drug resistance profile.

The weights are assigned in the following way: all drug-susceptible strains are assigned weight $-\alpha(r)$ (defined as above); each drug-resistant strain S is assigned a weight $\frac{1}{n}$, where n is the number of drug-resistant strains in the subtree (containing strain S) determined by its highest parental node, such that the subtree does not contain any drug-susceptible strain in its leaves. All strains without drug resistance information are assigned weights 0.

Note that the support score can also be expressed as weighted support, where $w_T(b, r, S)$ are assigned as $-\alpha(r)$, 1, 0 for drug-susceptible, drug-resistant and strains without drug resistance information, respectively.

Figure 3.2 illustrates the concept of support and weighted support.

In order to make the support scores more comparable between drugs, we introduce normalized versions of the scores, *normalized support* and *normalized weighted support* which denote the respective support value divided by the maximal possible support or weighted support, respectively.

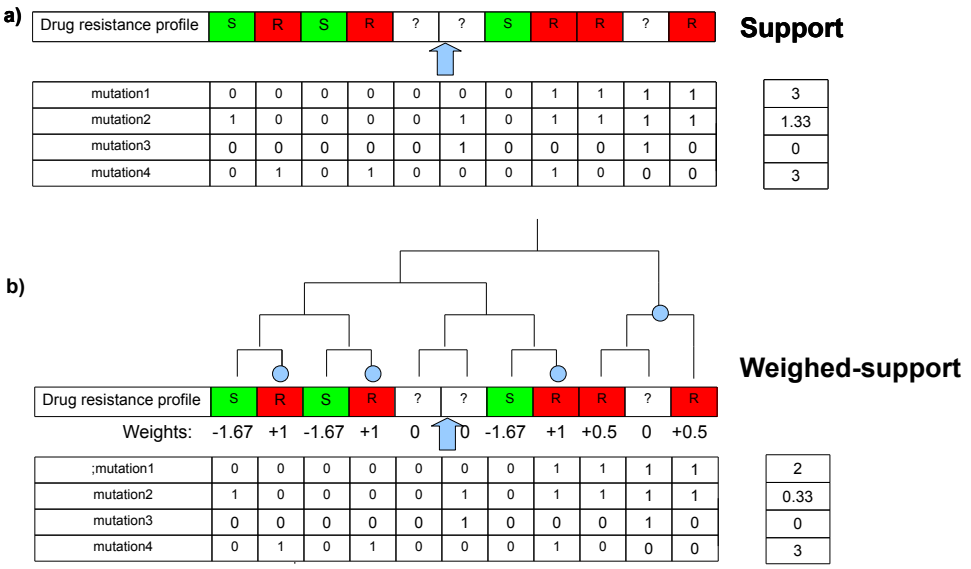

Figure 3.2: A schematic example of several mutation profiles and computation of their supports. Light blue circles mark nodes which appear in the definition of weighted support. These are nodes the highest parental nodes (for the leaf nodes corresponding to drug-resistant strains), that their subtrees do not contain any drug-susceptible strains in leaves. The scores (a) support and (b) weighted support are assigned to these mutations. For this drug-resistance profile, the ratio $\alpha(r)$ equals $\frac{5}{3}$.

STATISTICAL SIGNIFICANCE FOR WS In order to assess statistical significance of the associations we calculate their *p-values*.

More precisely, for a given drug resistance profile v, let X be the random variable giving support of a random mutation. Then, for a given observed mutation with $Support = c$, its p-value is defined by the following formula:

$$\mathbb{P}(X \geq c) = \sum_{n=1}^{|\mathcal{S}|} \mathbb{P}(X \geq c | N = n) \cdot \mathbb{P}(N = n) \tag{3.12}$$

Here, N is a random variable which denotes the number of mutated strains in a random mutation. For each n the probability $\mathbb{P}(N = n)$ of observing a mutation present in n strains is estimated (as the number of mutations present in n strains to the total number of considered mutations) from the data for point mutation and gene gain/loss profiles separately. The details follow. Assume that weights, for a given drug resistance profile v, take k different values: $l_1, l_2, \ldots, l_k$. For $1 \leq j \leq k$, let m_j be the number of strains which take value l_j. Clearly we have $m_1 + m_2 + \ldots + m_k = |\mathcal{S}|$. Then, the probability $\mathbb{P}(X \geq c | N = n)$ (from

the equation 3.12) is given by the formula:

$$\sum_{\substack{0\leq n_1\leq m_1\\ 0\leq n_2\leq m_2\\ \dots\\ 0\leq n_k\leq m_k\\ n_1+n_2+\ldots+n_k=n}} \frac{\prod_{j=1}^{k}\binom{m_j}{n_j}}{\binom{|\mathcal{S}|}{n}}\Big[\sum_{j=1}^{k} n_j\cdot l_j \geq c\Big] \tag{3.13}$$

Here we describe our algorithm for calculating the p-value. It should be clear that the problem reduces to computing $\mathbb{P}(X \geq c|N = n) = \frac{t_c(n)}{\binom{|\mathcal{S}|}{n}}$ for each $0 \leq n \leq |\mathcal{S}|$, where $t_c(n)$ denotes the number of ways for distributing n ones over $|\mathcal{S}|$ strains, such that the corresponding sum of weights is greater or equal than c. The term $\binom{|\mathcal{S}|}{n}$ is the total number of possible ways for distributing n ones over $|\mathcal{S}|$ strains. Thus, the problem reduces to calculating $t_c(n)$ for each $0 \leq n \leq |\mathcal{S}|$. Additionally, without any loss of generality, we may assume that the weight levels are strictly decreasing: $l_1 > l_2 > \ldots > l_k$, where $l_k < 0$ and $l_{k-1} \geq 0$.

The algorithm iteratively generates partial combinations (without n_k) starting from the partial combination $(n_1 = m_1, \ldots, n_{k-1} = m_{k-1})$ in the following manner: if j is the highest index of the non-zero n_i in the current partial combination, the next partial combination will be $(n_1, \ldots, n_j - 1, n_{j+1} = m_{j+1}, \ldots, n_{k-1} = m_{k-1})$. The algorithm terminates generating partial combinations when two 1following partial combinations have their corresponding sum of weights below the level of c. At each step of the algorithm, all possible full combinations $(n_1, \ldots n_{k-1}, n_k)$ are generated from the current partial combination $(n_1, \ldots n_{k-1})$. If for the full combination its corresponding sum of weights is greater or equal c $(\sum_{i=1}^{k} n_i \cdot l_i \geq c)$, then we increment the value $t_c(n)$ by $\prod i = 1^k \binom{m_i}{n_i}$, where $n = n_1 + \ldots + n_k$. As the outcome, we obtain $t_c(n)$ and, thus, also $\mathbb{P}(X \geq c|N = n)$ for each n.

The last step is to calculate formula 3.12 using these calculated probabilities.

Note that, since support is a special case of weighted support, the same formula and algorithm can be used to compute its corresponding p-values.

3.2.2.6 Tree-generalized hypergeometric score

As a part of this work, we also introduce a new association score, called tree-generalized hypergeometric (TGH) score, which is conceptually similar to the

CCTSWEEP score proposed by Habib et al. (2007).

We consider a set of bacterial strains $\mathcal{S}$ with its rooted phylogenetic tree T, whose leaves correspond to the strains in $\mathcal{S}$. Let V_T denote the set of all nodes (internal and leaves) in T. Let additionally, function $P_T : V_T \Rightarrow V_T \cup \{\text{null}\}$, for a given $\omega \in V_T$, return its parent node; or null for the root node. Let also function C_T, for a given node $\omega \in V_T$, return the set of its immediate child nodes.

We also introduce function L_T which, for each node ω in T, returns the subtree of descendants of the node, including the node itself. We say these nodes are visible from ω. Additionally, the function L_T applied to any subset c of V_T returns the union of all nodes visible from nodes in the set. More formally, $L_T(c) = \bigcup_{\omega \in V_T} L_T(\omega)$.

In order to present the formal definition of TGH, we first define some auxiliary concepts.

Let $\overline{r} : V_T \to \{\text{'?'}, \text{'S'}, \text{'R'}\}$ denote the *tree-extended resistance profile* defined recursively as follows:

$$\overline{r}(\omega) = \begin{cases} r(S) & \text{if } \omega \text{ is a leaf node corresponding to strain } S \\ \text{'S'} & \exists_{\omega' \in C_T(\omega)} \overline{r}(\omega') = \text{'S'} \\ \text{'R'} & \neg\exists_{\omega' \in C_T(\omega)} \overline{r}(\omega') = \text{'S'} \wedge \exists_{\omega' \in C_T(\omega)} \overline{r}(\omega') = \text{'R'} \\ \text{'?'} & \text{otherwise} \end{cases} \tag{3.14}$$

Analogously, let $\widehat{b} : V_T \to \{\text{'?'}, \text{'0'}, \text{'1'}\}$ denote the *tree-extended binary mutation profile* defined recursively as follows:

$$\widehat{b}(\omega) = \begin{cases} b(S) & \text{if } \omega \text{ is a leaf node corresponding to strain } S \\ \text{'0'} & \exists_{\omega' \in C_T(\omega)} \widehat{b}(\omega') = \text{'0'} \\ \text{'1'} & \neg\exists_{\omega' \in C_T(\omega)} \widehat{b}(\omega') = \text{'0'} \wedge \exists_{\omega' \in C_T(\omega)} \widehat{b}(\omega') = \text{'1'} \\ \text{'?'} & \text{otherwise} \end{cases} \tag{3.15}$$

For a given tree T, we call a subset c of its nodes a coloring, if it satisfies the following two conditions:

(A) each path from a leaf to the root contains at most one node from c,

(B) each internal node in T has at least one immediate child node which does not belong to c.

We call a coloring $\overline{c}$ *induced* by a given drug resistance profile r, if it contains the set of nodes in which drug resistance was acquired. More formally, we define a coloring induced by a drug resistance profile r, using its corresponding tree-extended resistance profile $\overline{r}$, as:

$$\overline{c} = \{\omega \in V_T : \overline{r}(\omega) = \text{'R'} \wedge \big(P_T(\omega) = \text{null} \vee \overline{r}(P_T(\omega)) = \text{'S'}\big)\}. \tag{3.16}$$

Analogously, we call a coloring $\widehat{c}$ induced by a given binary mutation profile b, if it contains the set of nodes in which the mutation was acquired. More formally, we define a coloring induced by a binary mutation profile b, using its corresponding tree-extended mutation profile $\widehat{b}$, as:

$$\widehat{c} = \{\omega \in V_T : \widehat{b}(\omega) = \text{'1'} \wedge \Big(P_T(\omega) = \text{null} \vee \widehat{b}(P_T(\omega)) = \text{'0'}\Big)\}. \tag{3.17}$$

Figure 3.3 (A) presents an example of colorings induced by a given drug resistance profile (large red nodes) and a given binary mutation profile (small orange nodes) for a flat tree. Figure 3.3 (B) presents another example of colorings induced by the same pair of profiles, but for a tree which is not flat. In this model the dependencies between different strains are captured by the topology of the tree.

$$W_\omega(n) = \#\{c \in \mathcal{C}_T(\omega) : |c| = n\} \tag{3.18}$$

Here, $\mathcal{C}_T(\omega)$ denotes the set of all colorings of $L_T(\omega)$. We denote by $W_T(n)$, the value of $W_\omega(n)$ for the root node ω in T.

We also define $B_{\omega,\overline{c}}(k,n)$ as the number of colorings of size n, such that exactly k nodes of that coloring are visible from nodes of coloring $\overline{c}$. More formally,

$$B_{\omega,\overline{c}}(k,n) = \#\{c \in \mathcal{C}_T(\omega) : |L_T(\overline{c}) \cap c| = k \wedge |c| = n\} \tag{3.19}$$

We denote by $B_{T,\overline{c}}(k,n)$ the value of $B_{\omega,\overline{c}}(k,n)$ for the root node ω in T.

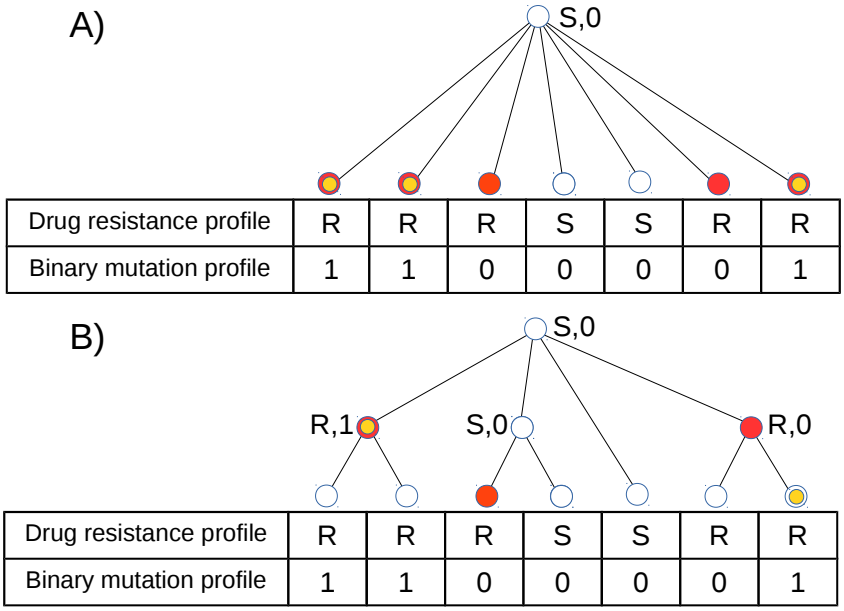

Figure 3.3: **(A)** an example of a pair of a drug resistance profile and a binary mutation profile. Values of the corresponding tree-extended binary mutation profile, and the corresponding tree-extended drug resistance profile are shown next to the nodes. Nodes belonging to the coloring induced by the drug resistance profile $\overline{c}$ are indicated by large red nodes, whereas nodes belonging to the coloring induced by the binary mutation profile $\widehat{c}$ are indicated by small orange nodes. In this example $|\overline{c}| = 5$, $|\widehat{c}| = 3$ and $|L_T(\overline{c}) \cap \widehat{c}| = 3$. **(B)** colorings $\overline{c}$ and $\widehat{c}$ induced by the same pair of profiles but for a different tree. In this example $|\overline{c}| = 3$, $|\widehat{c}| = 2$ and $|L_T(\overline{c}) \cap \widehat{c}| = 2$.

Finally, for a drug resistance profile r and a binary mutation profile b, we denote the colorings induced by the profiles as $\overline{c}$ and $\widehat{c}$, respectively. Let additionally, $n = |\overline{c}|$ and $k = |L(\overline{c}) \cap \widehat{c}|$. Then, we finally define the TGH score, as follows:

$$\mathrm{TGH}_T(r, b) = -log\Big(\frac{\sum_{i=k}^{n} B_{T,\overline{c}}(i, n)}{W_T(n)}\Big). \tag{3.20}$$

We take the negative logarithm to have consistent property, with other scoring methods, such that the higher the score the more likely drug resistance profile r is associated with binary mutation profile b.

THE ALGORITHM FOR TGH Here we describe the algorithm we use to compute the TGH score for a set of pairs of drug resistance profiles and binary mutation profiles.

Naturally, for each leaf node ω in T, two colorings exist: $c_1 = \{\omega\}$, $c_2 = \emptyset$. The following lemma 1 characterizes colorings for internal nodes of T.

Lemma 1. *Let ω be an internal node in T with l immediate child nodes $(\omega_1, \dots \omega_l)$. Let c be a subset of V_T. Then, c is a coloring of $L_T(\omega)$ if and only if $c = \{\omega\}$ or $\big(\omega \notin c$ and $c \neq \{\omega_1, \dots \omega_l\}$ and $c \cap L_T(\omega_i)$ is a coloring of $L_T(\omega_i)$, for each $\omega_i\big)$.*

Proof. $\Rightarrow$: Proof by contradiction. Let us assume $c \neq \{\omega\}$. If $c = \{\omega_1, \dots \omega_l\}$, then c contradicts with the (B) condition of the definition of a coloring. Thus, there exists ω_i, such that, $c \cap L_T(\omega_i)$ does not satisfy (A) or (B). Since $c \cap L_T(\omega_i)$ is a subset of c, c also violates the corresponding (A) or (B) condition. Hence, it contradicts with our assumption that c is a coloring.
$\Leftarrow$: naturally, $\{\omega\}$ satisfies both (A) and (B). Otherwise, ince $\omega \notin c$, $c = \bigcup_{\omega_i} c \cap L_T(\omega_i)$. Thus, c satisfies (A). The condition (B) is satisfied unless $c \cap L_T(\omega_i) = \{\omega_i\}$, but this case is excluded as a separate case. □

Based on the proposition 1 we can derive the following recursive formulas for $W_\omega(n)$. If ω is a leaf node in T, then:

$$W_\omega(n) = [n = 0] + [n = 1] \tag{3.21}$$

If ω is an internal node in T, then:

$$W_\omega(n) = \underbrace{[n = 1]}_{c=\{\omega\}} - \underbrace{[n = l]}_{\{\omega_1,\dots,\omega_l\} \text{ is not a coloring}} + \sum_{\substack{0 \le n_1 \le n, \dots, 0 \le n_l \le n \\ n_1 + \dots + n_l = n}} \prod_{i=1}^{l} W_{\omega_i}(n_i) \tag{3.22}$$

Similarly, we can derive the recursive formulas for $B_{\omega,\overline{c}}(k, n)$. If ω is a leaf node in T, then:

$$\begin{aligned} B_{\omega,\overline{c}}(k, n) = \; & [n = 1 \wedge k = 1 \wedge \overline{c} = \{\omega\}] \\ & + [n = 1 \wedge k = 0 \wedge \overline{c} \neq \{\omega\}] \\ & - [n = l \wedge k = |L(\overline{c}) \cap \{\omega_1, \dots, \omega_l\}|] \end{aligned} \tag{3.23}$$

If ω is an internal node in T, then:

$$\begin{aligned} B_{\omega,\overline{c}}(k,n) = \; & [n = 1 \wedge k = 1 \wedge \overline{c} = \{\omega\}] \\ & +[n = 1 \wedge k = 0 \wedge \overline{c} \neq \{\omega\}] \\ & -[n = l \wedge k = |L(\overline{c}) \cap \{\omega_1, \ldots, \omega_l\}|] \\ & + \sum_{\substack{0 \leq n_1 \leq n, \ldots, 0 \leq n_l \leq n \\ n_1 + \ldots + n_l = n \\ 0 \leq k_1 \leq n_1, \ldots, 0 \leq k_l \leq n_l \\ k_1 + \ldots + k_l = k}} \prod_{i=1}^{l} B_{\omega_i,\overline{c}}(k_i, n_i) \end{aligned} \tag{3.24}$$

The pseudocode 3 presents the following steps of the algorithm to compute the TGH score for each pair of drug resistance profile and binary mutation profile. These steps, for a given drug resistance profile r, comprise: (i) simplification of the input tree T' to T by removal of the leaves corresponding to the strains with unknown drug resistance status (according to r); (ii) computation of the tree-extended resistance profile $\overline{r}$ and its corresponding coloring $\overline{c}$; (iii) computation of the values of $W_\omega(n)$ for each n and $\omega \in V_T$, following the recursive formulas 3.21 and 3.22 from the leaves to the root (dynamic programming technique); (iv) computation of the values of $B_{\omega,\overline{c}}(k,n)$ for each k, n and $\omega \in V_T$, following the recursive formulas 3.23 and 3.24, from the leaves to the root (dynamic programming technique); (from leaves to the root); (v) for each binary mutation profile $b \in \mathcal{B}$, computation of the tree-extended binary mutation profile $\widehat{b}$ and its corresponding coloring $\widehat{c}$; and finally (vi) computation of the TGH score based on formula 3.20.

Additionally, in order to speed up the computations of the and $W_T(n)$ and $B_{T,\overline{c}}(k,n)$ values we use the memorization technique to cache results depending on the topology of a subtree. The subtree topologies, used as hashes, are represented as strings in the Nawick tree format enriched by the additional information of belonging to $\overline{c}$, for each node.

Also, due to high time complexity of the score with respect to the maximal number of immediate children of a node, in all computational experiments we calculate the actual TGH score as an average over TGH scores obtained for trees generated by randomly binarizing the input tree.

Algorithm 3 Pseudocode for computing the TGH score

Require: A set $\mathcal{S}$ of bacterial strains; with a phylogenetic tree T', a set of binary resistance profiles $\mathcal{R}$, and a set of binary mutation profiles $\mathcal{B}$. The function $simplify$ removes a node ω from the tree T' if the strains corresponding to the set of leaves visible from ω have all unknown drug resistance status in r. After this step, it removes all internal nodes of degree one.

for all $r \in \mathcal{R}$ **do**
  $T \leftarrow simplify(r, T')$
  compute the tree-extended resistance profile $\overline{r}$ for r in T
  compute the coloring $\overline{c}$ induced for $\overline{r}$ in T
  compute $W_\omega(n)$ bottom-up for every n and $\omega \in V_T$, following the 3.21 and 3.22 formulas
  compute $B_{\omega,\overline{c}}(k,n)$ bottom-up for every k, n and $\omega \in V_T$, following the 3.23 and 3.24 formulas
  for all $b \in \mathcal{B}$ **do**
    compute the tree-extended mutation profile $\widehat{b}$ for b in T
    compute the coloring $\widehat{c}$ for $\widehat{b}$ in T
    $n \leftarrow |\widehat{c}|$
    $k \leftarrow |L_T(\overline{c}) \cap \widehat{c}|$
    $\text{TGH} \leftarrow -log\Big(\frac{\sum_{i=k}^{n} B_{T,\overline{c}}(i,n)}{W_T(n)}\Big)$
  end for
end for{These computations are done in parallel for each drug resistance profile $r \in \mathcal{R}$}
return TGH score for each pair $r \in \mathcal{R}$ and $b \in \mathcal{B}$.

3.2.2.7 Rank-based metascore

Finally, we introduce an association score, called *Rank-based metascore* (RBM), which combines a set of scores into a new score. This approach is based on the natural assumption that each individual score has its own good and weak points. Thus, RBM tries to compromise between the different approaches used to define different scores. This score is based on rankings after sorting with accordance to the scores being combined, rather than the absolute values of the scores.

Let $S_1, S_2, \ldots, S_k$ denote the set of different scores to be combined with RBM. Then, for a given binary mutation profile $b \in \mathcal{B}$ and resistance profile $r \in \mathcal{R}$, the score is defined as the sum of average rankings of b with accordance to scores in

question. More formally,

$$\mathrm{RBM}(S_1, \ldots, S_k)(b, r) = \sum_{i=1}^{k} \frac{\mathrm{rank}_u^{S_i}(b, r) + \mathrm{rank}_d^{S_i}(b, r)}{2}. \tag{3.25}$$

Here, $\mathrm{rank}_u^{S_i}(b, r)$ denote the highest ranking of the binary mutation profile with the same S_i score as b, which is the number of binary mutation profiles with the S_i score higher than b plus 1, more formally:

$$\mathrm{rank}_u^{S_i}(b, r) = \#\{b' \in \mathcal{B} : S_i(b', r) > S_i(b, r)\} + 1. \tag{3.26}$$

Analogously, we define $\mathrm{rank}_d^{S_i}(b, r)$ as the lowest ranking of the binary mutation profile with the same S_i score as b, which is the number of binary mutation profiles with the S_i score higher or equal than b, more formally:

$$\mathrm{rank}_d^{S_i}(b, r) = \#\{b' \in \mathcal{B} : S_i(b', r) \geq S_i(b, r)\}. \tag{3.27}$$

Note that, if each binary mutation profile has a different score, the formula $\frac{\mathrm{rank}_u^{S_i}(b,r)+\mathrm{rank}_d^{S_i}(b,r)}{2}$ simplifies to return the ranking of b on the sorted list of binary mutation profiles with respect to the score S_i.

In order to compute the RBM, assuming all the individual scores are already computed, we sort the lists of mutations for each individual score and drug resistance profile r, separately. Then we compute the rank_u and rank_d mappings. Finally, we compute the actual RBM.

Note that, unlike the other scores presented in this work, here, the lower the value of the score the higher the chance the association is real. This definition of RBM is consistent with the current implementation of the score.

In the thesis we consider three versions of the score: (i) combining all the tree-ignorant scores, denoted: RBM (MI,OR,H); (ii) combining WS and TGH, denoted RBM (WS,TGH); and combining (iii) all the five individual scores, denoted RBM (MI,OR,H,WS,TGH) and also shortly RBM (ALL). Note that RBM (MI,OR,H) can be categorized as tree-ignorant score, whereas RBM (WS,TGH) and RBM (ALL) as tree-aware.

3.2.3 Time complexity

Let D denote the number of drug resistance profiles considered. Additionally, let N denote the number of considered strains and M denote the number of binary mutation profiles. Finally, let K denote the maximal number of children of an internal node in the tree. Then, the time complexity of the algorithms we implemented to compute the hypergeometric score, the mutual information, odds ratio, and weighted support is $O(D \cdot N \cdot M)$.

In order to compute the TGH score for the input tree T, based on the formulas 3.23 and 3.24, we implement the dynamic programming algorithm to compute bottom-up the values $B_{\omega,\bar{c}}(k, n)$ for each internal node ω in T, k and n. The time complexity of computing these values for all the nodes is $O(\cdot N^{2\cdot(K-1)} \cdot N)$. Similarly, based on the recursive formulas 3.21 and 3.22,we implement the dynamic programming algorithm to compute bottom-up the values $W_\omega(n)$ for all nodes in T and n. The time complexity of this step is $O(\cdot N^{\cdot(K-1)} \cdot N)$.

This strategy gives the algorithm to compute the TGH score with time complexity $O(D \cdot N^{2\cdot(K-1)} \cdot N + D \cdot N \cdot M)$ which simplifies to $O(D \cdot N \cdot (M + N^2))$ for binary trees.

The time complexity of the algorithm to compute the RBM for a set of E individual scores, assuming the scores are already computed, is $O(D \cdot E \cdot M)$. Note that the time complexity does not depend on the number of strains N.

3.3 Results and Discussion

Here we present the results of applying GWAMAR to three datasets. One for *S. aureus* and two for *M. tuberculosis*.

3.3.1 S. aureus dataset

We first discuss the computational experiment on the dataset of 100 *S. aureus* strains. We use this case study to show the usability of GWAMAR to identify genes associated with drug resistance.

3.3.1.1 Genotype data

We collected genotype data (genome sequences and annotations) for 100 fully sequenced strains of *S. aureus* from the GenBank (Benson et al., 2013) and PATRIC

databases (Gillespie et al., 2011). Additionally, genotype data for strain *EMRSA-15* were downloaded from the Wellcome Trust Sanger Institute website. At the time of writing, 31 out of the 100 *S. aureus* strains had the sequencing status "completed". For the remaining strains whose genomes were still being assembled, contig sequences (covering around 90% of the genomes) and annotations were used.

We unified the original genome annotations employing CAMBer. However, in order to determine gene families we additionally extended the multigene consolidation graph by edges coming from BLAST amino-acid queries. More formally, we added an edge between a pair of genes to the consolidation graph if the percent of identity (calculated as the number of identities over the length of the longer gene) of the BLAST hit between them exceeded a threshold $P(L)$ given by the HSSP curve formula (Rost, 1999):

$$P(L) = \begin{cases} 100 & L \leq 11 \\ c + 480 \cdot L^{-0.32 \cdot (1 + e^{-L/1000})} & 11 < L \leq 450 \\ c + 19.5 & L > 450 \end{cases} \tag{3.28}$$

Here, c was set to 40.5 and L is the number of aligned amino-acid residues.

Then, each connected component in the multigene consolidation graph corresponds to a gene family. We computed multiple alignments using MUSCLE (Edgar, 2004) for all these gene families.

In this work, unlike in the current version of GWAMAR, we considered two kinds of genetic variations (mutations):

- gene gain/losses,
- point mutations (in amino-acid sequences).

In comparison to the current version of GWAMAR, we did not take into account mutations in gene promoter regions. Here, point-mutation profiles are transformed from columns in multiple alignments computed for gene families with elements present in at least $|\mathcal{S}| - 1$ strains.

3.3.1.2 Phylogenetic tree of the strains

We computed the phylogenetic tree of the input strains using a consensus method with majority rule implemented in the PHYLIP package, developed by Felsenstein (2005). We applied the consensus method to trees constructed for all gene families with exactly one element in each strain. The trees were constructed using the maximum likelihood approach implemented in the PHYLIP package.

3.3.1.3 Phenotype data (drug susceptibility)

We performed a careful search of the literature for results of drug susceptibility tests of the strains considered. The drug susceptibility data were collected from the following sources: (i) 25 publications issued together with the fully sequenced genomes; (ii) NARSA project (`http://www.narsa.net`); (iii) email exchange with the authors of publications related to strains *ST398* and *TW20*; and (iv) other publications found by searching related literature. In total we used 71 publications to retrieve the drug resistance information.

3.3.1.4 Assessment of accuracy

We verified the usability of our approach by trying to re-identify known drug resistance determinants. In this experiment, we compared the proposed scoring — support and weighted support— to odds ratio, which is a popular measure used in genome-wide association studies. Table 3.1 shows rankings of the gene-gain/-loss profiles for genes which are known drug resistance determinants. The experiment suggests that weighted support outperforms both: support and odds ratio. The latter two scores do not incorporate additional information about phylogeny

3.3.1.5 Prediction of resistance

This experiment also reveals that the amount of the collected drug resistance information is not sufficient to correctly identify drug resistance-associated genes. However, the high consistency of drug resistance profiles corresponding to the collected information and the presence of drug resistance determinants (summing over drugs, there are 117 drug-resistant strains, where only 4 of them do not

		Rankings before prediction			Rankings after prediction		
gene id.	drug name	S	WS	OR	S	WS	OR
tet	tetracycline	54.5	2.5	43.7	1.5	1.5	1.5
tetM	tetracycline	14.5	11.5	7.5	4	4	4
mecA	methicillin	1	1	1	1	1	1
mecA	oxacillin	3	4	2	1	2	1
ermA1	clindamycin	5.5	5.5	5.5	1	1	1
ermC	clindamycin	907	471	907	414.5	11	191.5
ermA1	erythromycin	3	3	4	1	1	1
ermC	erythromycin	1527	3994.5	1006.5	413.5	28	214.5
aacA-aphD	gentamicin	72	34	34	1	1	1
blaZ	penicillin	163	66	223	1.5	1	2.5
mecA	penicillin	163	8	223	11	5	52
Average ranking (excluding ermC):		53.27	**15.05**	60.411	2.55	**1.94**	7.22

Table 3.1: Rankings of the known drug resistance determinants obtained by employing three different methods to score gene-gain/-loss profiles: support (S), weighted support (WS) and odds ratio (OR). Since some of the gene-gain/-loss profiles are assigned with the same score, we calculate their rankings as the arithmetic mean of positions of the profiles with the same score on the list sorted according to the scores; thus some of the rankings are not round numbers. The rankings were computed before and after prediction of drug resistance, which is based on the presence of drug resistance determinants. We excluded the gene *ermC* from the calculations of average rankings since none of the methods were able to pull it out into the top 100 before prediction.

have any known drug resistance determinants; and there are 112 drug-susceptible strains, where only 8 of them have at least one drug resistance determinant) suggests that we can use the determinants to predict drug resistance in the strains without drug resistance information available.

It is perhaps questionable to predict drug resistance in those strains for which the whole-genome sequence is not determined yet. So we did prediction only for those strains with completed sequencing or at least information on their plasmids (which often carry the drug resistance determinants). Nevertheless, we predicted drug resistance also for those strains that were not yet fully sequenced, provided the presence of drug resistance-determining genes had been confirmed for them. Moreover, we predicted drug resistance to rifampicin and ciprofloxacin for all 100 strains, as the drug resistance for rifampicin and ciprofloxacin is determined by point mutations in genes *rpoB*, *gyrA* and *grlA* (synonymous name to *parC*), which were sequenced in all strains. More precisely, we predicted as rifampicin-resistant all strains with any mutation present in the rifampicin resistance determining region (RRDR). We defined the RRDR as the amino-acid range from 463 to 530 in the *rpoB* gene sequence (according to (O'Neill et al., 2006)). Analogously,

we predicted as ciprofloxacin-resistant all strains with any point mutation in the quinolone resistance determining region (QRDR). We defined QRDR as the amino-acid ranges from position 68 to 107 and from position 64 to 103 in the *grlA* and *parC* gene sequences, respectively (according to (Ferrero et al., 1995)). Figure 3.4 shows the complete information about drug susceptibility after prediction.

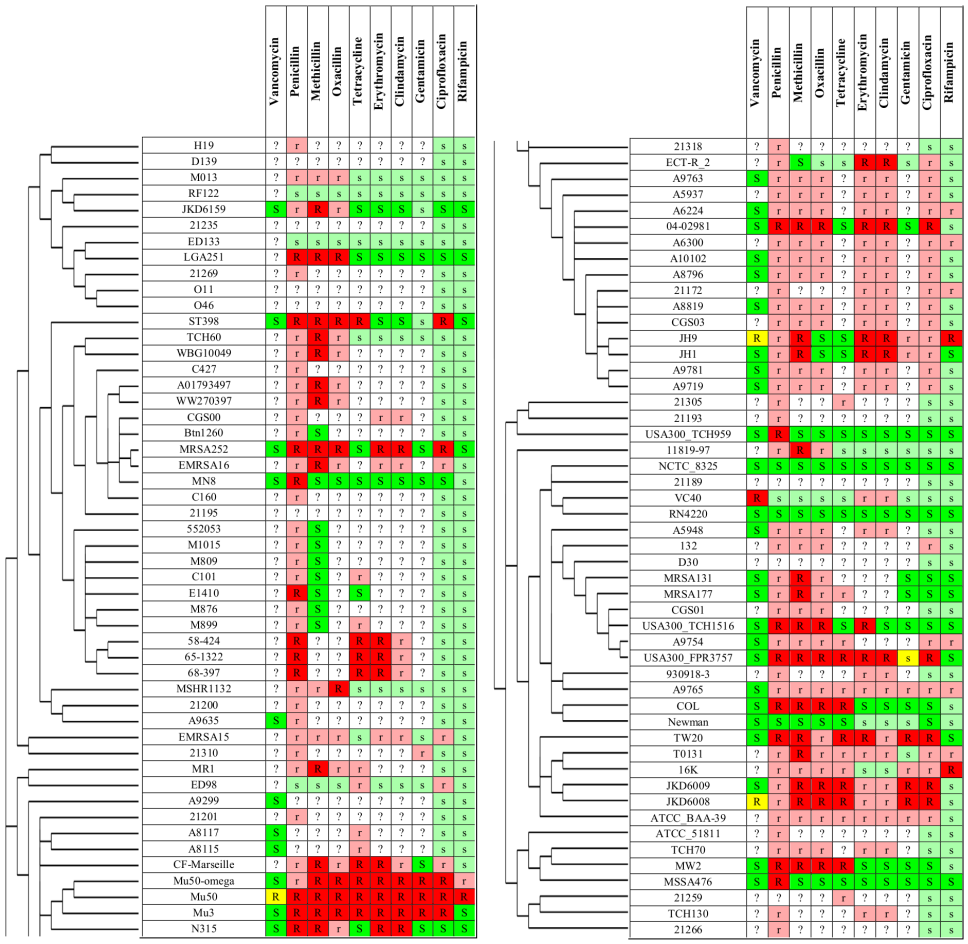

Figure 3.4: The collected dataset of phenotypes put together with results of our drug resistance predictions based on the presence of known drug resistance determinants. Due to the high number of strains the table is split into two panels. Columns represent drugs, rows represent *S. aureus* strains included in the study in the order corresponding to the reconstructed phylogenetic tree of strains. Green, yellow and red cell colors represent susceptible, intermediate resistant and resistant phenotypes, respectively. Analogously, light green and light red cell colors represent predicted susceptible and resistant phenotypes, respectively. White cell color represents unknown (not determined by experiments or prediction) drug resistance phenotypes.

3.3.1.6 Essential mutations

Here, we distinguish two categories of gene-gain/-loss and point-mutation profiles depending on how they correspond to a given drug resistance profile. We categorize a given mutation profile m as:

- *Essential* mutation, when m is absent in all drug-susceptible strains,
- *Conflict* mutation, when m is present in at least one drug-susceptible strain.

Further, we distinguish *neutral* mutations as a subclass of essential mutations, these are essential mutations that are not present in any of drug-resistant strains. Thus, neutral mutations may only be present in strains with unknown drug-resistance status.

Analogously, we transfer the above introduced concepts to gene-gain/-loss profiles, defining essential, neutral and conflict gene-gain/-loss profiles.

3.3.1.7 Detection of drug resistance-associated mutations

Then, we applied our approach to the dataset supplemented by the predicted information about drug susceptibility for the following drugs: tetracycline, β-lactams (penicillin, oxacillin, methicillin), erythromycin, gentamicin, vancomycin, ciprofloxacin and rifampicin.

Below we discuss the results of our approach applied separately to the following drugs: tetracycline, β-lactams (penicillin, methicillin), erythromycin, gentamicin, vancomycin, ciprofloxacin. We do not discuss here results for oxacillin and clindamycin, since they have very similar drug resistance profiles to methicillin and erythromycin, respectively. All other drugs were excluded from the analysis due to low number of strains with available drug resistance information on these drugs.

Tables 3.2 and 3.3 present the top-scored gene-gain/-loss, and point-mutation profiles for the discussed drugs, respectively. The genes presented in the tables were selected according to the following procedure: for each drug we construct a function, which gives for each gene (listed in descending order with respect to normalized weighted support) minus logarithm of p-value ($-log$(p-value)) of this score. Then, we report genes which correspond to the portion of the graph of this function before it gets flattened.

Gene identifier	NS	NWS	OR	p-value	Gene functional annotation
Penicillin (NWS-threshold: 0.58)					
⋆ SAR1831(blaZ)	0.84	0.81	37.15	1.15e-06	beta-lactamase
SAR1829(blaI)	0.84	0.74	37.15	5.24e-06	transcriptional repressor
SAR1830(blaR1)	0.82	0.73	31.27	7.09e-06	beta-lactamase regulatory protein blar1
SAR0056	0.63	0.71	12.13	1.03e-05	conserved hypothetical protein
⋆ SAR0039(mecA)	0.61	0.70	10.94	1.28e-05	methicillin resistance determinant mecA
SAR0060(ccrA)	0.61	0.63	10.94	4.40e-05	resolvase, n-terminal domain protein
SAR0061(yycG)	0.61	0.63	10.94	4.40e-05	putative membrane protein
NWMN 0025	0.57	0.63	9.40	4.41e-05	conserved domain protein
SAR0037(ugpQ)	0.60	0.63	10.39	5.08e-05	glycerophosphoryldiester phosphodiesterase
SAR0038(maoC)	0.60	0.63	10.39	5.08e-05	dehydratase
SAR0057	0.57	0.59	9.40	9.78e-05	conserved hypothetical protein
Methicillin (NWS-threshold: 0.68)					
⋆ SAR0039(mecA)	1.00	1.00	950.00	4.48e-20	methicillin resistance determinant mecA
SAR0037(ugpQ)	0.98	0.94	931.00	6.77e-15	glycerophosphoryldiester phosphodiesterase
SAR0038(maoC)	0.98	0.94	931.00	6.77e-15	dehydratase
SAR0056	0.95	0.85	900.00	7.55e-12	conserved hypothetical protein
SAR0036	0.64	0.80	33.78	5.77e-11	putative membrane protein
SAR0057	0.85	0.75	162.00	6.47e-10	conserved hypothetical protein
SAR0060(ccrA)	0.91	0.73	432.00	1.40e-09	resolvase, n-terminal domain protein
SAR0061(yycG)	0.91	0.73	432.00	1.40e-09	putative membrane protein
MW0028(ebpS)	0.54	0.71	22.30	2.76e-09	hmg-coa synthase
Tetracycline (NWS-threshold: 0.32)					
SAAV b3(repC)	0.54	0.64	27.69	5.70e-08	plasmid replication protein
⋆ SATW20 00660(tet)	0.54	0.64	27.69	5.70e-08	tetracycline resistance protein
SATW20 00670(pre)	0.50	0.50	24.00	3.51e-06	plasmid recombination enzyme type 3
⋆ SATW20 04620(tetM)	0.46	0.37	20.80	7.54e-05	tetracycline resistance protein tetM
SATW20 08990(virE)	0.42	0.37	19.93	7.67e-05	pathogenicity island protein
SATW20 09000	0.42	0.37	19.93	7.67e-05	pathogenicity island protein
SATW20 09010(lipA)	0.42	0.37	19.93	7.67e-05	superantigen-encoding pathogenicity islands
SATW20 04610(thiI)	0.43	0.35	18.00	1.32e-04	putative transcriptional regulator
MW0745(int)	0.25	0.32	8.00	2.28e-04	site-specific recombinase, phage integrase
MW0747	0.25	0.32	8.00	2.28e-04	DNA-binding helix-turn-helix protein
Erythromycin (NWS-threshold: 0.27)					
⋆ SAR0050(ermA1)	0.80	0.58	76.00	1.36e-06	rRNA adenine n-6-methyltransferase
CGSSa03 12660	0.47	0.44	17.19	2.98e-05	conserved hypothetical protein
SAR0054(tnpA1)	0.75	0.39	72.00	8.12e-05	transposase for transposon
SAR1734	0.75	0.39	72.00	8.12e-05	methylase
SAR1736(spc2)	0.75	0.39	72.00	8.12e-05	spectinomycin 9-o-adenylyltransferase
SaurJH9 1711(radC)	0.72	0.38	62.00	8.83e-05	predicted protein
SAUSA300 pUSA030006	0.20	0.35	4.75	1.65e-04	replication and maintenance protein
SAR1737(tnpC2)	0.72	0.34	62.00	1.89e-04	Unknown
SAR1529	0.33	0.33	9.15	2.43e-04	conserved hypothetical protein
SATW20 04860(recF 1)	0.23	0.30	5.52	3.67e-04	recombinational DNA repair ATPase
SAR1738(tnpB2)	0.70	0.29	54.00	4.39e-04	transposase B from transposon Tn554
SauraJ 010100009720	0.23	0.27	5.52	6.60e-04	conserved domain protein
Gentamicin (NWS-threshold: 0.83)					
⋆ SaurJH1 2806(aacA-aphD)	0.83	0.90	150.00	9.38e-11	bifunc. acetyltransferase/phosphotransferase
SaurJH1 2805	0.75	0.83	90.00	2.95e-09	GNAT family acetyltransferase
Ciprofloxacin (NWS-threshold: 0.4)					
SATW20 04610(thiI)	0.35	0.45	36.00	1.33e-07	putative transcriptional regulator
SATW20 04650(cap8J)	0.32	0.40	31.57	8.25e-07	lipoprotein
SATW20 04670(capL)	0.32	0.40	31.57	8.25e-07	putative ATP/GTP-binding protein
SATW20 04780	0.32	0.40	31.57	8.25e-07	conjugation related protein
SATW20 04800	0.32	0.40	31.57	8.25e-07	replication initiation factor
SATW20 04810	0.32	0.40	31.57	8.25e-07	DNA segregation ATPase FtsK/SpoIIIE
SATW20 04830	0.32	0.40	31.57	8.25e-07	conjugative transposon protein

Table 3.2: Summarizing information for the top scored gene-gain/-loss profiles. The consequent columns refer to: gene identifier of the corresponding gene family; normalized support (NS); normalized weighted support (NWS); odds ratio (OR); p-value and the gene functional annotation. Thresholds for weighted support are provided in brackets for each drug.

TETRACYCLINE Tetracycline acts by binding to the 30S ribosomal subunit (16S rRNA and the protein encoded by the gene *rpsS* are its direct targets), preventing

Gene identifier	desc.	NS	NWS	OR	p-value	Gene functional annotation
Penicillin (NWS-threshold: 0.4)						
SAR0023(sasH)	$G_{723}D$	0.55	0.63	8.51	1.87e-05	virulence-associated cell-wall-anchored protein
SAR0023(sasH)	$T_{725}A$	0.54	0.62	8.11	2.23e-05	virulence-associated cell-wall-anchored protein
SAR0304	$V_{295}I$	0.39	0.49	4.48	3.25e-04	acid phosphatase
SAR2791	$V_{182}M$	0.46	0.46	6.05	5.41e-04	transcriptional regulator, Xre family
SAR2700	$N_{493}KD$	0.52	0.45	7.72	6.16e-04	ABC transporter permease protein
SAR0233(hmp)	$Q_{333}K$	0.44	0.44	5.48	7.21e-04	flavohemoprotein
SAR0318(sbnA)	$N_{25}HK$	0.44	0.43	5.48	8.36e-04	alpha/beta family hydrolase
SAR2664	$V_{282}AT$	0.44	0.43	5.48	8.36e-04	probable monooxygenase
SAR2779	$S_{48}G$	0.44	0.43	5.48	8.36e-04	n-hydroxyarylamine o-acetyltransferase
SAR0318(sbnA)	$T_{138}IM$	0.43	0.43	5.21	8.36e-04	alpha/beta family hydrolase
SAR0318(sbnA)	$T_{139}AQ$	0.43	0.43	5.21	8.36e-04	alpha/beta family hydrolase
SAR0023(sasH)	$A_{749}TG$	0.41	0.43	4.96	8.44e-04	virulence-associated cell-wall-anchored protein
SAR0318(sbnA)	$R_{130}CG$	0.41	0.43	4.96	8.72e-04	alpha/beta family hydrolase
SAR0322(folC)	$H_{201}YQE$	0.41	0.43	4.96	8.72e-04	possibly adp-ribose binding module
SAR0233(hmp)	$K_{323}ET$	0.40	0.42	4.71	9.08e-04	flavohemoprotein
SAR2750(icaC)	$I_{21}V$	0.40	0.42	4.71	9.46e-04	polysaccharide intercellular adhesin biosynthesis
SAR0233(hmp)	$S_{309}RN$	0.39	0.42	4.48	9.46e-04	flavohemoprotein
Methicillin (NWS-threshold: 0.25)						
SAR0198(oppF)	$T_{287}IK$	0.10	0.29	2.11	1.41e-04	putative glutathione transporter, ATP-binding
SAR0420	$I_{72}F$	0.10	0.29	2.11	1.41e-04	membrane protein
SAR2508(sbi)	$S_{219}AT$	0.10	0.29	2.11	1.41e-04	IgG-binding protein Sbi
SAR2508(sbi)	$N_{222}QK$	0.10	0.29	2.11	1.41e-04	IgG-binding protein Sbi
SAR2508(sbi)	$K_{224}SDN$	0.10	0.29	2.11	1.41e-04	IgG-binding protein Sbi
Tetracycline (NWS-threshold: 0.2)						
SAR1840	$D_{291}YS$	0.18	0.23	5.22	7.09e-04	NAD(FAD)-utilizing dehydrogenases
SAR2336(rpsJ)	$K_{57}M$	0.29	0.23	9.60	7.32e-04	SSU ribosomal protein S10P (S20E)
SAR0550(rpsL)	$K_{113}R$	0.36	0.20	13.33	1.14e-03	SSU ribosomal protein S12P (S23E)
Erythromycin (NWS-threshold: 0.2)						
SAR0576	$A_{68}EV$	0.07	0.21	1.54	8.89e-04	phosphoglycolate phosphatase
Gentamicin (NWS-threshold: 0.21)						
SAR1840	$L_{289}IW$	0.33	0.29	15.00	1.43e-03	NAD(FAD)-utilizing dehydrogenases
SAR1840	$D_{291}YS$	0.33	0.29	15.00	1.43e-03	NAD(FAD)-utilizing dehydrogenases
SAR1840	$H_{327}RF$	0.33	0.29	15.00	1.43e-03	NAD(FAD)-utilizing dehydrogenases
SAR1167(ylmH)	$K_{215}N$	0.25	0.29	10.00	1.43e-03	RNA-binding S4 domain-containing protein
SAR1167(ylmH)	$R_{216}V$	0.25	0.29	10.00	1.43e-03	RNA-binding S4 domain-containing protein
SAR1167(ylmH)	$V_{217}L$	0.25	0.29	10.00	1.43e-03	RNA-binding S4 domain-containing protein
SAR0547(rpoB)	$D_{471}YG$	0.17	0.21	6.00	4.61e-03	DNA-directed RNA polymerase beta subunit
SAR1833(trmB)	$T_{54}IK$	0.17	0.21	6.00	4.61e-03	tRNA (guanine46-n7-)-methyltransferase
Ciprofloxacin (NWS-threshold: 0.12)						
SAR1367(grlA)	$S_{80}YF$	1.00	1.00	2244.00	6.03e-30	topoisomerase IV subunit a
SAR0006(gyrA)	$S_{90}AL$	0.94	0.88	1056.00	1.92e-18	DNA gyrase subunit a
SAR2449(lytT)	$V_{45}I$	0.21	0.20	17.11	2.06e-04	transcriptional regulator
SAR1840	$L_{289}IW$	0.12	0.20	8.80	4.56e-04	NAD(FAD)-utilizing dehydrogenases
SAR1793(thiI)	$A_{92}ET$	0.09	0.20	6.39	2.06e-04	thiamine biosynthesis protein thiI
SAR2212(murA2)	$A_{102}T$	0.06	0.20	4.12	2.06e-04	UDP-n-acetylglucosamine 1-carboxyvinyltransferase
SAR1367(grlA)	$E_{84}KG$	0.26	0.15	23.76	9.40e-04	topoisomerase IV subunit a
SAR0235(pstG_1)	$F_{401}LV$	0.09	0.13	6.39	2.21e-03	PTS system, maltose and glucose-specific
SAR0400(nfrA)	$R_{194}H$	0.09	0.13	6.39	2.21e-03	nitroreductase family protein

Table 3.3: Summarizing information for the top scored point mutation profiles, only for essential mutations. The conflict mutations were removed from the table for: tetracycline, erythromycin and gentamicin (for the rest of drugs there were no conflict mutations above the set thresholds). The consequent columns refer to: gene identifier of the corresponding gene family; corresponding position in the multiple alignment and changed amino acids; normalized support (NS); normalized weighted support (NWS); odds ratio (OR); p-value (computed as described in section 3.2.2.5) and the gene functional annotation. Thresholds for weighted support are provided in brackets for each drug.

binding of tRNA to the mRNA-ribosome complex, and thus inhibiting protein synthesis (Knox et al., 2011).

The most common drug resistance mechanism to tetracycline in *S. aureus* is

mediated by ribosome protection proteins (RPPs) such as *tet* and *tetM*, which bind to the ribosome complex, thus preventing the binding of tetracycline (Chopra and Roberts, 2001; Connell et al., 2003).

Genes *tet* and *tetM*, mediating this mechanism, cover all tetracycline-resistant strains, except *MW2*. The drug resistance status pf *MW2* may be caused by errors in the drug susceptibility test, errors in sequencing, or by some not-yet-known drug resistance mechanism. The inconsistent information about strain *MW2*'s tetracycline susceptibility and the lack of identified drug resistance determinants suggest that the strain is possibly drug susceptible. In our experiment we initially assumed that the tetracycline resistance information is not available for strain *MW2*.

Our method shows that, besides *tet* and *tetM*, there are a few more genes that have highly scored gene-gain/-loss profiles. Especially interesting are the following genes which are not gained by any of the drug susceptible strains: *repC*, *pre*, *thiI*, *int*, *clfB* (see Table 3.2). There are studies reporting the significance of these *clfB* and *repC* genes in drug resistance (McAleese and Foster, 2003; Werckenthin et al., 1996). Interestingly, the gene *repC* seems to co-evolve with *tet* (highly correlated gene-gain/-loss profiles).

Applying our method to point mutations, we have identified two highly scored (and essential) point mutations in ribosomal complex proteins: $K101R$ in *rpsL* and $K57M$ in *rpsJ*. According to our knowledge, this is the first report on the significance of the point mutations for drug resistance in *S. aureus*. However, mutations in *rpsJ* have been associated with tetracycline resistance in another bacteria *Neisseria gonorrhoeae* (Hu et al., 2005).

Beta-lactams Beta-lactams are a broad class of antibiotics, which possess (by definition) the β-lactam ring in their structure. The ring is capable of binding transpeptidase proteins (also known as Penicillin Binding Proteins — PBPs) (Knox et al., 2011), which are important for synthesis of the peptidoglycan layer of bacterial cell wall. PBPs with attached drug molecules are no longer able to synthesize peptidoglycan, leading to bacterial death (Sabath, 1982). In our case study, we consider three β-lactam antibiotics: penicillin, oxacillin and methicillin. However, since the drug resistance profile and drug resistance mechanisms for oxacillin and methicillin are very similar we discuss results only for methicillin.

There are two common resistance mechanisms to β-lactams in *S. aureus* (Sabath,

1982; Drawz and Bonomo, 2010). The first one is mediated by β-lactamase enzymes, which bind drug molecules and break the β-lactam ring, thus deactivating the drug molecules. This mechanism is effective against penicillin (which is β-lactamase sensitive) and not effective against methicillin and oxacillin (which are β-lactamase resistant) (WHO, 2010). The second β-lactam resistance mechanism is mediated by proteins which are capable of functionally substituting for PBPs, but have much smaller affinity to β-lactam molecules. This mechanism is effective against penicillin, methicillin and oxacillin.

PENICILLIN In our dataset, all strains resistant to penicillin possess proteins responsible for one of the two mechanisms. More precisely, there are 69 drug-resistant strains (with available drug resistance information), which possess *blaZ*—the standard β-lactamase protein (note that its regulators *blaR1* and *blaI* do not always co-occur). All the remaining penicillin-resistant strains have *mecA*, which is an altered PBP. Table 3.2 provides information about the top-scoring gene-gain/-loss profiles.

Applying our method we, have also identified the uncategorized putative protein, *SAR0056*, as putatively associated with penicillin resistance (see Table 3.2).

METHICILLIN Applying our approach to gene-gain/-loss profiles we identified (beside *mecA*) genes *ugpQ* and *maoC*. The correlation of gene profiles to the profile of *mecA* and their close proximity on the genomes suggests that these genes co-evolve (see Figure 3.5 for more details). This co-evolution may reflect some important role played by these genes in methicillin resistance. This calls for further study of the role of these two genes in methicillin resistance.

We have also identified a few point mutations that are putatively associated with methicillin resistance. Interestingly, two of the mutations in the top 10 essential mutations according to weighted support ($I72F$ in *SAR0420* and $E208Q/K/D$ in *SAR0436*) are present in cell membrane proteins. This suggests some compensatory mechanism to the presence of *mecA*.

CIPROFLOXACIN Ciprofloxacin belongs to a broad class of antibiotics, called fluoroquinolones, which are functional against bacteria by binding DNA gyrase subunit A (encoded by *gyrA*) and DNA topoisomerase 4 subunit A (encoded by

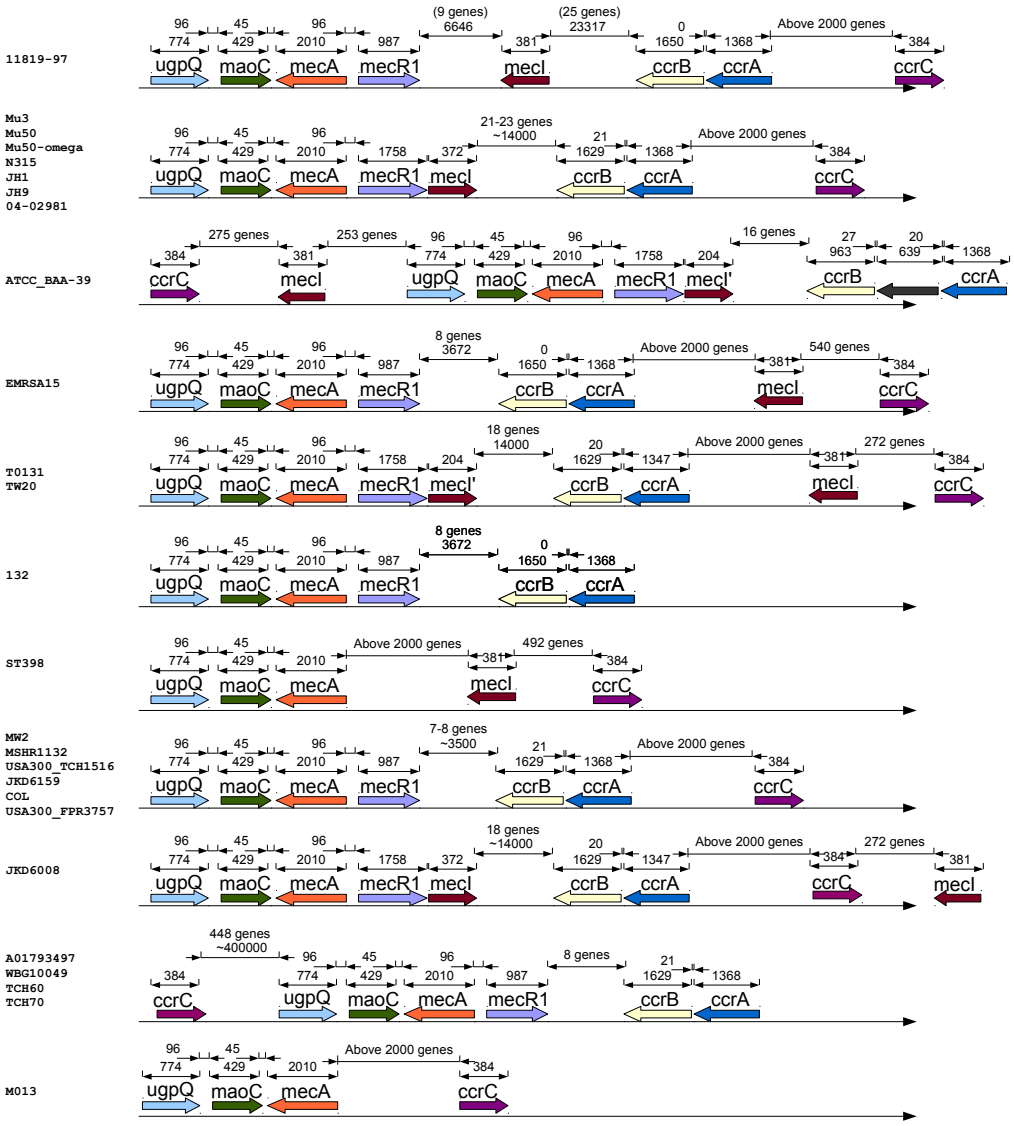

Figure 3.5: Presence and relative genome coordinates of genes related to methicillin resistance (*mecA*, *mecR1*, *mecI*, *ccrA*, *ccrB*, *ccrC*), put together with the identified genes: *ugpQ* and *maoC*. The gene presence profiles are clustered with respect to the genes order. In this figure we include only these methicillin-resistant strains for which all the genes where located on the main genome and within the same sequence contig (in order to determine the relative positions).

parC), which are enzymes necessary to separate bacterial DNA, thereby inhibiting cell division (Knox et al., 2011). The most common ciprofloxacin-resistance mechanism is mediated by point mutations in the drug targets, *parC* and *gyrA*.

Applying our approach we identified (by highest weighted support) two point mutations in ciprofloxacin target genes — $S80F/Y$ in *parC* and $S90A/L$ in *gyrA*— which are located in QRDR and known to be responsible for the first mechanism of ciprofloxacin resistance (Ferrero et al., 1995). The presence of these mutations is correlated with the ciprofloxacin resistance profile for strains with available drug resistance information. However, they differ for two strains *ED98* and *16K* (only the mutation in *parC* is present). This may suggest intermediate drug resistance level for these strains. Unfortunately ciprofloxacin resistance information is not available for these strains.

Erythromycin Erythromycin acts by binding the 23S rRNA molecule (in the 50S subunit) of the bacterial ribosome complex, leading to inhibition of protein synthesis (Knox et al., 2011).

There are three known erythromycin resistance mechanisms (Schmitz et al., 2000). First — the most common mechanism — is by methylation (addition of two residues to the domain V of 23S rRNA) of the 23S rRNA molecule, which prevents the ribosome from binding with erythromycin. This methylation is mediated by enzymes from the *erm* gene family, the most common are *ermA* and *ermC*. The second mechanism is mediated by the presence of macrolide efflux pumps (encoded by *msrA* and *msrB*). The third mechanism is the inactivation of drug molecules by specialized enzymes such as *ereA* or *msrB* (Schmitz et al., 2000).

We found that none of the strains in our case study possess genes *ereA* or *ereB*. Genes encoding efflux pumps (*msrA* and *msrB*) are present also in drug-susceptible strains (for example, *NCTC 8325* and *Newman*), which may suggest that the mechanism is inactive for the considered strains of *S. aureus* or the enzyme production rates are too small, which we are not able to account by our method. Using our approach we identified (by the highest support) the gene *ermA* responsible for the most common drug resistance mechanism.

Here, there is one erythromycin-susceptible strain, *USA300 TCH959*, which harbours the *ermA* gene. This may suggest disruption of the drug resistance mechanism in that strain, errors in drug susceptibility testing or errors in sequencing.

Interestingly, we identified gene *SAR1736(spc2)* (which is a known spectinomycin resistance determinant) as potentially associated with erythromycin re-

sistance. This suggests that drug resistance to spectinomycin and erythromycin co-evolved, despite these two drugs belonging to different classes according to the ATC drug classification system (WHO, 2010).

GENTAMICIN Gentamicin works by inhibition of protein synthesis by binding the 30S subunit of the ribosome complex (Shakil et al., 2008).

Interestingly, strain *USA300 FPR3757* exhibits intermediate drug resistance, which is correlated with the absence of *aacA-aphD* gene in its genome sequence. Since our method requires binary information on drug susceptibility, we marked this strain as drug-susceptible for experiments.

The most common resistance mechanism responsible for high levels of gentamicin resistance is mediated by the drug-modifying enzyme *SaurJH1 2806(aacA-aphD)*. Applying our methodology we identified the gene encoding it as likely to be associated with drug resistance (maximal support). Moreover, we identified also the gene *SaurJH1 2805* as putatively associated with gentamicin resistance. The close proximity of these two genes in the genomes and their highly correlated gene-gain/-loss profiles suggest co-evolution. We hypothesize that the gene *SaurJH1 2805* plays some role in drug resistance for gentamicin.

3.3.2 M. TUBERCULOSIS DATASETS

Here we present results obtained by applying GWAMAR to two large datasets for *M. tuberculosis*. We use these case studies to present the usability of GWAMAR to identify chromosomal mutations associated with drug resistance. The first dataset is prepared for the set of 173 strains with genome sequences and annotations publicly available in the PATRIC database, developed by Wattam et al. (2014). For this set of strains, we collected drug resistance information from over 20 publications. The genotype and phenotype data for the second dataset comes from the *M. tuberculosis* Drug Resistance Directed Sequencing Database at `http://www.broadinstitute.org/annotation/genome/mtb_drug_resistance`.

3.3.2.1 FIRST CASE STUDY

The first case study is based on the set of 173 fully sequenced strains of *M. tuberculosis* with publicly available data.

The preprocessing steps of preparing the genotype data were performed using eCAMBer, our tool to support comparative analysis of multiple bacterial strains (Woźniak et al., 2012).

In particular, first, we used eCAMBer to download the genome sequences and annotations from the PATRIC database (Wattam et al., 2014). Next, we applied eCAMBer to unify the genome annotations of protein-coding genes and to identify the clusters of genes with high sequence similarity. Then, for the subset of 4379 such identified gene clusters with genes present in at least 90% of the strains, we computed multiple alignments using MUSCLE (Edgar, 2004). The multiple alignments were computed for amino acid sequences of protein coding genes, as well as nucleotide sequences of their promoter regions (-50bp upstream), and rRNA genes. In total, based on the computed multiple alignments, we identified 118913 mutations, which constituted the input genotype data for GWAMAR. After the procedure of binarization in GWAMAR we ended up with 18635 binary mutation profiles.

The input phenotype data was collected from over 20 publications issued together with the fully sequenced genomes. Based on the drug resistance information collected for ciprofloxacin and ofloxacin, we introduced a new drug resistance profile for the drug family of fluoroquinolones. A strain was categorized as susceptible to fluoroquinolones if it was susceptible to at least one of the drugs, but not resistant to any of them. Similarly, a strain was categorized as resistant to fluoroquinolones if it was resistant to at least one of the drugs, but not susceptible to any of them. If none of the cases was satisfied for a strain, then the drug resistance status of the strain was categorized as unknown. We restrict analysis to the set of six drugs or drug families: fluoroquinolones, ethambutol, isoniazid, pyrazinamide, rifampicin and streptomycin.

The input phylogenetic tree was reconstructed using the maximum likelihood approach implemented in the PhyML, developed by (Guindon et al., 2010). As for the input for the tool we used the set of all the identified point mutations concatenated into one multiple alignment file. This input was prepared by an additional feature of eCAMBer.

Having prepared the set of binary mutation profiles and drug resistance profiles, together with the phylogenetic tree, we applied GWAMAR to compute MI, OR, H, WS, TGH and RBM association scores. However, in order to compute TGH score efficiently, we averaged three TGH scores obtained over three random

binary trees we obtained from the original tree by splitting its nodes with multifurcations. This step of computations, ran using 6 processors, took around 6s for MI, OR, H and WS; around 34s for TGH; and around 3s for all the considered variants of the RBM score.

drug name	gene id	gene name	mutation	all	h.c.	TGH
Fluoroquinolones	Rv0006	gyrA	$D94H_1A_5N_2Y_2G_{12}$	Y	Y	14.184
Isoniazid	Rv1908c	katG	$S315N_1G_2T_{75}$	Y	Y	9.045
Rifampicin	Rv0667	rpoB	$S450L_{71}$	Y	Y	8.602
Streptomycin	Rv0682	rpsL	$K43R_{15}$	Y	Y	8.323
Ethambutol	Rv3795	embB	$M306L_1I_{32}V_{18}$	Y	Y	8.250
Isoniazid	Rv1483	fabG1	$C\text{-}15T_{30}$	Y	Y	5.845
Rifampicin	Rv0667	rpoB	$D435Y_2F_5V_{11}G_3A_1$	Y	Y	5.040
Streptomycin	Rv0682	rpsL	$K88R_5M_1$	Y	Y	4.164
Ethambutol	Rv3795	embB	$E504G_1D_1$	N	N	3.331
Pyrazinamide	Rv2043c	pncA	$H51P_1$	Y	Y	2.708
Pyrazinamide	Rv2043c	pncA	$W68L_1$	Y	Y	2.708
Rifampicin	Rv0667	rpoB	$H445D_8Y_2R_1$	Y	Y	2.530
Streptomycin	Rvnr01	rrs	$G1108C_2$	N	N	1.717
Ethambutol	Rv3795	embB	$D869G_1$	N	N	1.688
Ethambutol	Rv3795	embB	$A505T_1$	N	N	1.688
Ethambutol	Rv3795	embB	$D1024N_1$	Y	N	1.688
Fluoroquinolones	Rv0005	gyrB	$N538T_1$	Y	Y	1.685
Fluoroquinolones	Rv0006	gyrA	$S91P_1$	Y	Y	1.685
Fluoroquinolones	Rv0005	gyrB	$T539I_1$	N	N	1.685
Streptomycin	Rvnr01	rrs	$A1401G_{17}$	Y	N	1.288
Ethambutol	Rv3795	embB	$Y334H_2$	Y	N	1.054
Ethambutol	Rv3795	embB	$Q497R_2$	Y	Y	1.054
Rifampicin	Rv0667	rpoB	$E250G_3$	N	N	1.047
Fluoroquinolones	Rv0006	gyrA	$A90V_6G_3$	Y	Y	1.035
Streptomycin	Rvnr01	rrs	$C517T_{33}$	Y	Y	0.915

Table 3.4: 25 top-scoring associations between drug resistance profiles and point mutations in the case study on 173 fully sequenced *M. tuberculosis* strains, when restricted to only these genes which are associated with drug resistance to the corresponding drugs
. Each row corresponds to one association, whereas the consecutive columns describe: drug name, gene identifier, gene name, mutation, association presence in the TBDReaMDB database, status indicating whether the association is categorized as high-confidence in TBDReaMDB, TGH score. Lower indexes in the mutation descriptions indicate the numbers of strains possessing the corresponding amino acid or nucleotide variant.

We took a closer look at the top-scoring mutations returned by the scores, but restricting our analysis to only these associations which involve genes which are known to be associated with drug resistance for the corresponding drug — possessing at least one point mutation annotated as high-confidence in the TBDReaMDB database. Table 3.4 presents the list of top 25 associations ordered according to TGH score. In the set of 25 top-scored associations, 19 are present in the TBDReaMDB database and 16 of them are categorized as high-confidence mutations. A closer look at the mutations which are not present in TBDReaMDB revealed that some of them can be supported by literature. In particular, muta-

tion $E504G/D$ in *embB* has recently been reported as associated with resistance to ethambutol (Bakuła et al., 2013). The close proximity of this mutation to $A505T$ in *embB* may also suggest that the latter is associated with ethambutol resistance. Similarly, the mutation $T539I$ has already been associated with resistance to fluoroquinolones (Malik et al., 2012).

Literature search did not provide us any additional support for the remaining three mutations ($D869G$ in *embB* and $G1108C$ in *rrs*), which haven't been also reported in TBDReaMDB.

3.3.2.2 Second case study

The second case study, *mtu_broad*, is based on the data available in the Broad Institute database `http://www.broadinstitute.org/annotation/genome/mtb_drug_resistance`. This database contains sequencing data and drug resistance information for 1398 strains of *M. tuberculosis*. However, it should be noted that only genes of interest were sequenced; Table 3.5 presents the list of 28 sequenced genes for each strain. Additionally 12 promoter sequences were sequenced. In total, this database contains 1067 mutations (non-synonymous amino-acid changes or nucleotide changes in promoters), which constituted the input genotype data for GWAMAR. After the procedure of binarization in GWAMAR we ended up 850 binary mutation profiles.

Similar to the previous case study, based on the drug resistance information available in the database for ciprofloxacin, ofloxacin, levofloxacin and moxifloxacin, we introduced a new drug resistance profile for the drug family of fluoroquinolones. A strain was categorized as susceptible to fluoroquinolones if it was susceptible to at least one of the drugs, but not resistant to any of them. Similarly, a strain was categorized as resistant to fluoroquinolones if it was resistant to at least one of the drugs, but not susceptible to any of them. If none of the cases was satisfied for a strain, then the drug resistance status of the strain was categorized as unknown. We restrict further analysis to the set of six drugs or drug families: fluoroquinolones, ethambutol, isoniazid, pyrazinamide, rifampicin and streptomycin.

In these experiments the phylogenetic tree was reconstructed using the maximum likelihood approach implemented in the PhyML package, developed by Guindon et al. (2010). As an input for the tool we used the set of all available mutations

gene id	gene name	description	prom. sequenced?
Rv0005	gyrB	DNA gyrase subunit B	yes
Rv0006	gyrA	DNA gyrase subunit A	yes
Rv0341	iniB	isoniazid inductible gene protein	yes
Rv0342	iniA	isoniazid inductible gene protein	yes
Rv0343	iniC	isoniazid inductible gene protein	yes
Rv0667	rpoB	DNA-directed RNA polymerase beta chain	yes
Rv0682	rpsL	30S ribosomal protein S12	yes
Rv1483	fabG1	3-oxoacyl-[acyl-carrier protein] reductase	yes
Rv1484	inhA	NADH-dependent enoyl-[acyl-carrier-protein] reductase	yes
Rv1694	tlyA	cytotoxin|haemolysin	no
Rv1854c	ndh	NADH dehydrogenase	yes
Rv1908c	katG	catalase-peroxidase-peroxynitritase T	no
Rv2043c	pncA	pyrazinamidase/nicotinamidas	yes
Rv2245	kasA	3-oxoacyl-[acyl-carrier protein] synthase 1	no
Rv2427Ac	oxyR'	hypothetical protein	no
Rv2428	ahpC	alkyl hydroperoxide reductase C protein	yes
Rv2764c	thyA	thymidylate synthase	yes
Rv2764c	ddl	D-alanine-D-alanine ligase ddlA	no
Rv3423c	alr	alanine racemase	no
Rv3793	embC	membrane indolylacetylinositol arabinosyltransferase	yes
Rv3794	embA	membrane indolylacetylinositol arabinosyltransferase	yes
Rv3795	embB	membrane indolylacetylinositol arabinosyltransferase	yes
Rv3854c	ethA	monooxygenase	yes
Rv3919c	gid	glucose-inhibited division protein B	yes
Rvnr01	rrs	ribosomal RNA 16S	no
Rvnr02	rrl	ribosomal RNA 23S	no

Table 3.5: List of sequenced genes and promoters available at the Broad Institute database, http://www.broadinstitute.org/annotation/genome/mtb_drug_resistance.

concatenated into one multiple alignment file. The preparation of the multiple alignment file as well as running PhyML was done with the use of eCAMBer.

Similarly, as in the *mtu173* dataset, we applied GWAMAR to compute MI, OR, H, WS, TGH and RBM association scores. As in the previously described case study, in order to compute TGH score efficiently, we averaged three TGH scores obtained over three random binary trees we obtained from the original tree by splitting its nodes with multifurcations. This step of computations, ran using 6 processors, took around 5s for MI, OR, H and WS; around 2h for TGH; and around 2s for all the considered variants of the RBM score. It took relatively long time to compute TGH score due to its high time complexity with respect to the numbers of strains considered 3.2.3.

Similarly as for the *mtu173* dataset, we sort the set of putative associations according to the TGH score, but restricting our analysis to only these associations which involve genes which are known to be associated with drug resistance for the corresponding drug — possessing at least one point mutation annotated as high-confidence in the TBDReaMDB database. Table 3.6 presents the list of the

drug name	gene id	gene name	mutation	all	h.c.	TGH
Fluoroquinolones	Rv0006	gyrA	$D94Y_6H_2A_{26}G_{78}N_{14}$	Y	Y	128.323
Rifampicin	Rv0667	rpoB	$S450L_{743}W_{22}$	Y	Y	72.284
Ethambutol	Rv3795	embB	$M306T_1L_{16}V_{290}I_{313}$	Y	Y	70.217
Fluoroquinolones	Rv0006	gyrA	$A90G_2V_{46}$	Y	Y	41.699
Streptomycin	Rv0682	rpsL	$K43R_{228}$	Y	Y	30.012
Isoniazid	Rv1908c	katG	$S315T_{895}G_2I_3R_3N_{27}$	Y	Y	27.966
Ethambutol	Rv3795	embB	$Q497H_5K_{18}P_{10}R_{43}$	Y	Y	17.081
Streptomycin	Rv0682	rpsL	$K88Q_1R_{28}T_{32}M_7$	Y	Y	16.327
Fluoroquinolones	Rv0005	gyrB	$N538K_1S_1T_9D_2$	Y	Y	12.605
Rifampicin	Rv0667	rpoB	$H445P_2Q_2L_{27}Y_{53}R_{42}D_{25}N_7$	Y	Y	12.252
Streptomycin	Rvnr01	rrs	$A1401G_{254}$	Y	N	9.509
Streptomycin	Rvnr01	rrs	$A514C_{90}$	Y	Y	8.940
Pyrazinamide	Rv2043c	pncA	$T135A_1P_{22}$	Y	N	8.814
Fluoroquinolones	Rv0006	gyrA	$S91P_9$	Y	Y	7.557
Rifampicin	Rv0667	rpoB	$D435H_1N_2A_2Y_{27}G_3V_{140}$	Y	Y	7.480
Ethambutol	Rv3795	embB	$G406C_3A_{68}D_{52}S_{43}$	Y	Y	7.057
Pyrazinamide	Rv2043c	pncA	$T\text{-}11G_3C_{24}$	Y	Y	6.766
Fluoroquinolones	Rv0006	gyrA	$D89G_2N_4$	Y	N	6.253
Pyrazinamide	Rv2043c	pncA	$L120P_{20}R_5$	Y	N	6.146
Streptomycin	Rvnr01	rrs	$C517T_{26}$	Y	Y	5.169
Pyrazinamide	Rv2043c	pncA	$Q10H_3R_{10}P_{12}$	Y	Y	5.053
Pyrazinamide	Rv2043c	pncA	$V139M_3G_2A_7L_1$	Y	Y	5.053
Ethambutol	Rv3795	embB	$D328G_5H_1Y_9$	Y	N	5.032
Streptomycin	Rvnr01	rrs	$A908C_7G_1$	Y	N	4.779
Pyrazinamide	Rv2043c	pncA	$D12E_1G_5N_1A_{12}$	Y	Y	4.725

Table 3.6: 25 top-scored associations between drug resistance profiles and point mutations in the case study for 1398 partially sequenced *M. tuberculosis* strains, when restricted to only these genes which are associated with drug resistance to the corresponding drugs. This dataset is provided by The Broad Institute. Each row corresponds to one association, whereas the consecutive columns describe: drug name, gene identifier, gene name, description of the mutation, association presence in the TBDReaMDB database, status indicating whether the association is categorized as high-confidence in TBDReaMDB, TGH score. Lower indexes in the mutation descriptions indicate the numbers of strains possessing the corresponding amino acid or nucleotide variant.

top 25 associations ordered according to TGH score. In the set of 25 top-scored associations, all are present in TBDReaMDB and 19 of them are categorized as high-confidence mutations. The presence in the TBDReaMDB database provides some additional support for the six associations which are categorized as high-confidence.

3.3.2.3 Assessment of accuracy

Here we use the two datasets described above to assess the accuracy of the various association scores, viz: mutual information, odds ratio, hypergeometric, weighted support, TGH and RBM.

We considered to use for comparison CCTSWEEP, proposed by Habib et al.

(2007) — a score conceptually similar to the TGH score. However, we failed to run its implementation, probably, due to rather poor documentation. Its authors had not responded to our queries in time. Thus, we omitted it from our experiments.

In order to assess the accuracy of different association scores we need a reliable dataset of known drug resistance associations. Here, we test two approaches to define our gold standard. In the first, we take all 607 associations from the Tuberculosis Drug Resistance Mutation Database (TBDReaMDB), developed by Sandgren et al. (2009). In the second, we use the subset of 81 drug resistance associations classified as *high-confidence* in the database. Table 3.6 presents the list of the mutations in TBDReaMDB with the distinguished subset of high-confidence associations. In all comparative experiments we assume a putative association to be a positive if it is present in the gold standard.

In both case studies, as the set of positives, we assume the subset of mutations present in our gold standard, also present in the available genotype data. This is the set of mutations which may be potentially detected (we say they are "detectable") using the available datasets. Thus, in the case when all TBDReaMDB associations constitute the gold standard, there are 94 and 212 of such "detectable" associations for the *mtu173* and *mtu_broad* datasets, respectively. Likewise, if only high-confidence associations are considered as gold standard, then 39 and 74 of such "detectable" associations for the *mtu173* and *mtu_broad* datasets, respectively

The set of negatives is constructed by random sampling from the whole set of identified putative associations except for the associations which are classified as positives. The number of sampled negatives equals the total number of mutations present in the genes which have at least one mutation in the gold standard. It should be noted that, among the mutations present within the genes which are associated with drug resistance, many can be real positives (associated with drug resistance), but lacking the annotation in TBDReaMDB. Thus, we use this approach of sampling from all mutations in order to significantly reduce the probability of classifying as negatives associations which are real but not present in the database.

Figure 3.7 presents statistics for the Area Under the Curve (AUC) for the precision and recall curves for different association scores. The results come from

drug name	gene id	gene name	positions	
			TBDReaMDB (high-conf.)	TBDReaMDB (all)
Ethambutol	Rv0340			173
	Rv0341	iniB		-89,47
	Rv0342	iniA		308,501
	Rv0343	iniC		248,351
	Rv1267c	embR		7,32,53... (24 in total)
	Rv3124	moaR1		-16
	Rv3125			54
	Rv3126			276
	Rv3264c	manB		152
	Rv3266c	rmlD		-71,257,284
	Rv3793	embC		5,244,247... (25 in total)
	Rv3794	embA		-16,-12,-11... (10 in total)
	Rv3795	embB	306,406,497	128,221,225... (85 in total)
Fluoroquinolones	Rv0005	gyrB	538	457,458,472... (9 in total)
	Rv0006	gyrA	90,91,94,102,126	74,80,88... (10 in total)
Isoniazid	Rv0129c	fbpC		-63,-23,158
	Rv0340			163
	Rv0342	iniA		3,537
	Rv0343	iniC		83
	Rv1483	fabG1	-15,-8	-92,-67,-24... (10 in total)
	Rv1484	inhA		8,16,21... (10 in total)
	Rv1592c			-29,42,430
	Rv1772			4
	Rv1854c	ndh		13,18,110,239,268
	Rv1908c	katG	279,315	1,2,11... (171 in total)
	Rv1909c	furA		5
	Rv2243	fabD		275
	Rv2245	kasA	269	66,77,121... (7 in total)
	Rv2247	accD6		229
	Rv2428	ahpC	-46,-39,21	-66,-49,-46... (21 in total)
	Rv2846c	efpA		73
	Rv3566c	nat		67,207
	Rv3795	embB		333
Pyrazinamide	Rv2043c	pncA	-11,7,10... (51 in total)	-12,-11,-7... (103 in total)
Rifampicin	Rv0667	rpoB	432,435,441,445,450,452	65,300,409... (38 in total)
Streptomycin	Rv0682	rpsL	43,88	9,40,41... (11 in total)
	Rv3919c	gidB		16,40,45... (19 in total)
	Rvnr01	rrs	492,513,514,517,907	190,427,462... (17 in total)

Figure 3.6: The list of drug resistance associations in the TBDReaMDB database. The first three columns correspond to the drug name, gene identifier and gene name of the gene corresponding to the point mutation. The next column lists the positions of the mutations corresponding to associations classified as *high-confidence* in the TBDReaMDB database. The last column lists positions of all the mutations present in the database. Each positive number indicates the position of the mutation in the amino acid sequence of the corresponding gene. Each negative number indicates the position of the mutation in the nucleotide sequence of the promoter of the gene, counting upstream its TIS. In some cases (if there are too many mutation to fit them within the table width) we do not list them all here — the complete list might be accessed at the project website, http://bioputer.mimuw.edu.pl/gwamar.

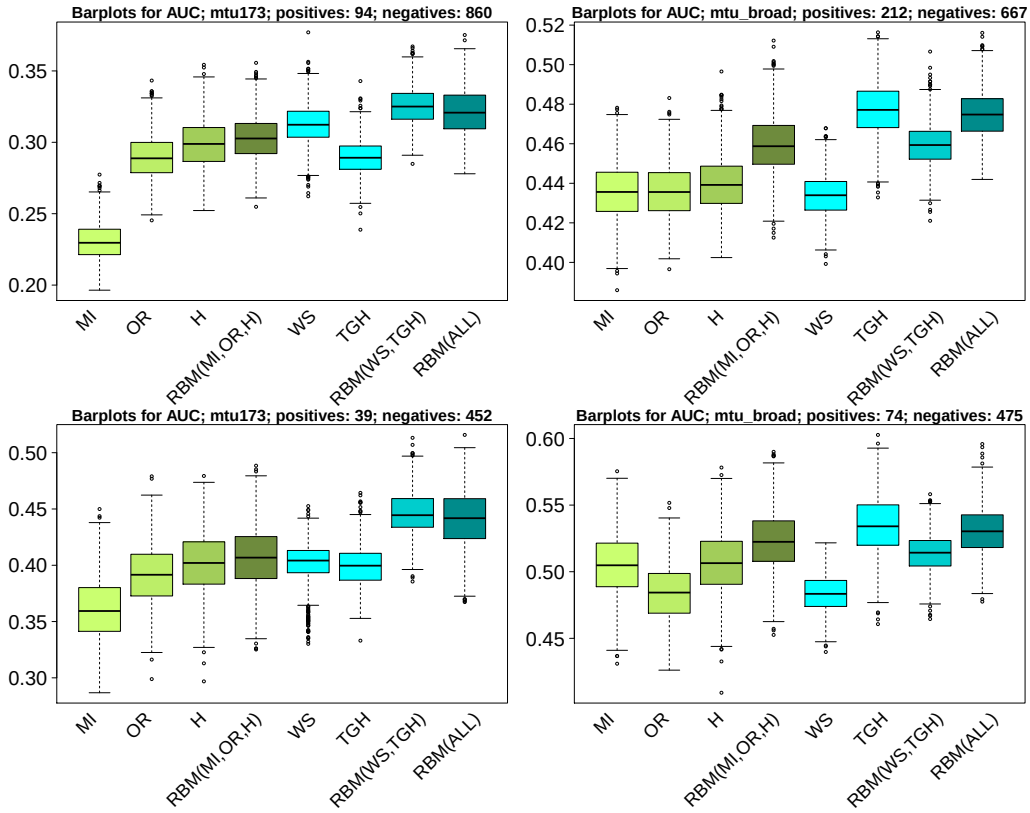

Figure 3.7: Comparison of different association scores implemented in GWAMAR based on the Area Under the Curve (AUC) statistic for the precision-recall curves. Left panels present the results for the *mtu173* dataset; right for the *mtu_broad* dataset. The first row of panels corresponds to the experiments in which all associations present in TBDReaMDB were used as the gold standard, whereas the second row corresponds to the experiments in which only high-confidence associations were used as the gold standard. The process of sampling the set of negatives was repeated 1000 times. The barplots for tree-ignorant and tree-aware scores are shown green and blue, respectively.

sampling the set of negatives and calculating the AUC, repeated 1000 times. The results show that on average, the tree-aware statistics (WS, TGH) performed slightly better than the tree ignorant scores on the *mtu173* dataset. They also show that, TGH performed best on the large *mtu_broad* dataset, but was slightly worse on the relatively small *mtu173* dataset. The presented results also show, that the Rank-based metascore performs consistently better than individual scores in most of the settings. For example, the RBM (MI,OR,H) outperformed all individual scores it combines in all the settings. Notably, the RBM(ALL) score, outperformed consistently all tree-ignorant scores in all the settings.

We conclude that, the tree-aware association scores outperform the tree-ignorant methods. In particular, the Rank-based metascore performed consistently better than the individual scores. However, the advantage is rather small and dependent on a setting. The performance may be influenced by the tree topology, the strains number or the small number of positives.

3.3.2.4 Compensatory mutations

The most common mechanism of rifampicin resistance in *M. tuberculosis* is acquired by point mutations within the rifampicin resistance determining region (RRDR) in the *rpoB* gene, which corresponds to the rifampicin binding spot (Patra et al., 2010).

Since the *rpoB* gene is essential for bacteria, mutations present in this gene, due to altering its structure, have often deleterious effect on the bacterial fitness in the drug-free environment (Brandis and Hughes, 2013). This effect may be potentially reversed by compensatory mutations. Thus, compensatory mutations tend to appear later, in the evolutionary history, than the mutations directly responsible for drug resistance. Hence, for a given compensatory mutation, we expect to observe it in a subset of strains which correspond to the mutation directly responsible for drug resistance.

Based on the above described assumption, we identify putative compensatory mutations using the following procedure applied to the *mtu_broad* and *mtu173* datasets. First, we identify the set of mutations within RRDR. Here, RRDR is defined as a region of 27 amino acids between positions 426 and 452 in the *rpoB* gene. Mutations from this region constitute the set of putative primary (directly responsible for drug resistance) mutations. For the *mtu173* dataset we obtained the following list of putative primary mutations:

- $\mathrm{L430P_1}$
- $\mathrm{D435Y_2F_5V_{11}G_3A_1}$
- $\mathrm{H445D_8Y_2R_1}$
- $\mathrm{S450L_{71}}$
- $\mathrm{L452P_2}$

Here, the description of each mutation comprises of the reference amino acid, the position of the mutation in the gene, and different amino-acid variants of the mutation among the strains. For each mutation, the lower indexes indicate the number of strains possessing the corresponding amino-acid variant of the mutation within the 173 strains in the *mtu173* dataset.

Applying the same method to the *mtu_broad* dataset we obtained the following list of putative primary mutations:

- $S428R_{2}$
- $Q429P_{1}H_{1}$
- $L430P_{3}R_{9}$
- $S431G_{1}$
- $Q432P_{5}E_{2}L_{1}K_{5}H_{5}$
- $M434I_{2}$
- $D435H_{1}N_{2}A_{2}Y_{27}G_{3}V_{140}$
- $N437H_{1}K_{1}$
- $N438H_{1}$
- $P439S_{1}$
- $S441L_{4}$
- $H445P_{2}Q_{2}L_{27}Y_{53}R_{42}D_{25}N_{7}$
- $R448Q_{7}$
- $S450L_{743}W_{22}$
- $L452V_{1}P_{24}$

Here, similarly, as for the previously described list of mutations, for each mutation, the lower indexes indicate the number of strains possessing the corresponding amino-acid variant of the mutation within the 173 strains in the *mtu_broad* dataset.

Interestingly, in both case studies the sets of strains possessing the primary mutations tend to be disjoint. For example, for the *mtu173* dataset, the sets of strains possessing mutations at positions 450 and 435 are disjoint (hypergeometric test p-value=$2.302 \cdot 10^{-8}$). The sets of strains possessing the mutations at positions 450 and 445 are also disjoint (hypergeometric test p-value=0.00026). Similarly, for the *mtu_broad* dataset, the sets of strains possessing mutations at positions 450 and 445 are disjoint (hypergeometric test p-value=$2.62 \cdot 10^{-63}$). The sets of strains possessing mutations at positions 450 and 435 overlap by only three elements (hypergeometric test p-value=$3.87 \cdot 10^{-64}$). We hypothesize that this phenomenon may be caused by the negative epistatic interactions between mutations from RRDR (Khan et al., 2011).

Finally, we identify a set of putative compensatory mutations, applying the following simple rule: a mutation is classified as a putative compensatory mutation if the set of strains possessing the mutation is contained within the set of strains corresponding to one of the mutations identified as primary mutations (from RRDR).

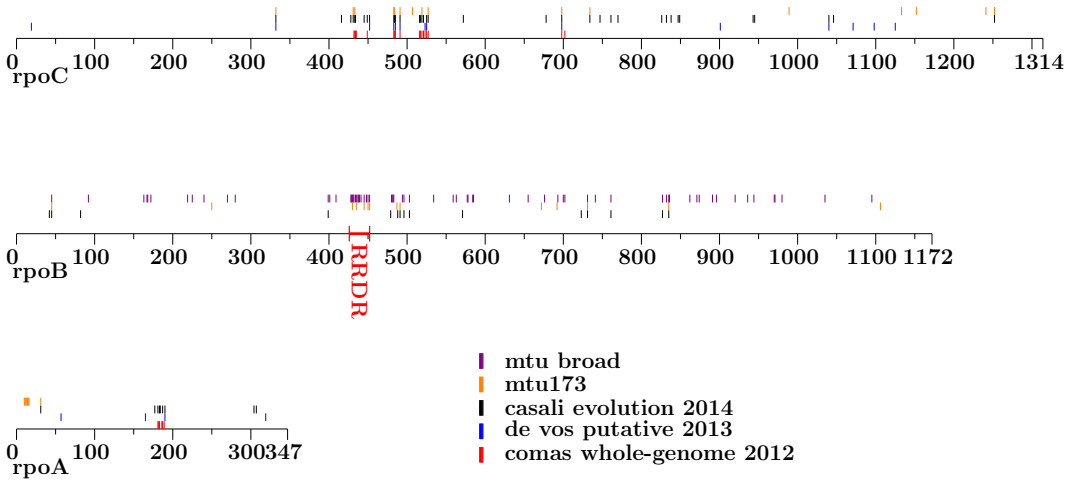

Figure 3.8: Comparison of the sets of putative compensatory mutations within the *rpoA*, *rpoB* and *rpoC* genes, reported in various sources and detected in our two datasets. Each mutation's position is indicated by a vertical line of the color corresponding to the source it was reported in. In particular orange and violet lines indicate positions of mutations identified by our approach applied to the *mtu173* and *mtu_broad* datasets, respectively. The other lines indicate mutations reported in the recent articles by Comas et al. (2012) (red), de Vos et al. (2013) (blue) and Casali et al. (2014) (black).

Here, we compare the sets of putative compensatory mutations for rifampicin

within the *rpoA*, *rpoB* and *rpoC* genes, identified by our approach with the mutations reported in other recent articles by Comas et al. (2012), de Vos et al. (2013), and Casali et al. (2014).

Figure 3.8 presents the distribution of the putative compensatory mutations identified in these recent studies, put together with the set of putative compensatory mutations identified based on our two case studies. Note that the identified putative compensatory mutations tend to cluster within the region of the *rpoC* gene from 430 to 530.

Table 3.7 presents another view on the list of putative compensatory mutations identified by our approach with comparison with those reported in the other recent articles. However, due to space limitation, in this table, we only list a subset of such mutations. A mutation is listed in the table, if it was identified in one of our two datasets and also reported in at least one of the three recent articles, or reported in at least two of the articles. The complete list of putative compensatory mutations is available on the website of our project. `http://bioputer.mimuw.edu.pl/gwamar`.

We conclude that using our approach we were able to re-identify most of the putative compensatory mutations identified previously. Moreover, in contrast to the other research groups, which used in house sequenced bacteria, we achieved our results by analysis on freely and publicly available data.

3.4 Summary

In this chapter, we presented GWAMAR, a tool we have developed for identifying of drug resistance-associated mutations based on comparative analysis of whole-genome sequences in bacterial strains.

The tool is designed as an automatic pipeline which employs eCAMBer for preprocessing of the genotype data. This preprocessing includes: (i) downloading of genome sequences and gene annotations, (ii) unification of gene annotations among the set of considered strains, (iii) identification of gene families, (iv) computation of multiple alignments and identification of point mutations which constitute the input genotype data.

GWAMAR implements various statistical methods — such as mutual information, odds ratio, hypergeometric score — to associate the drug resistance phe-

gene	position	ref aa	comas 2012	de vos 2013	casali 2014	mtu173	mtu broad
rpoA	31	G			A_1S_1	A_1S_1	
rpoA	181	T	I_1		A_1		
rpoA	183	V	A_1G_1		G_1		
rpoA	187	T	A_3P_1		A_1P_1		
rpoA	190	D		G_1	G_1		
rpoB	45	P			S_1	L_1	$A_3L_9S_7T_2$
rpoB	399	T			A_1		A_5I_4
rpoB	491	I			V_1	L_1	
rpoB	496	V			L_1M_1		G_1M_3
rpoB	503	F			S_1		S_3
rpoB	731	L			P_1	P_1	P_8
rpoB	761	E			D_1		D_1
rpoB	827	R			C_1		C_3
rpoB	835	H			P_1R_1	R_1	P_1R_3
rpoC	332	G		R_2	$C_1R_1S_1$	S_7	
rpoC	431	V			M_1	M_1	
rpoC	433	G	S_1		C_1S_1	S_1	
rpoC	434	P	A_1R_1		Q_1R_1		
rpoC	449	L	V_1		V_1		
rpoC	452	F		C_1	C_1		
rpoC	483	V	A_3G_3	A_1G_3	A_1G_1	A_1G_5	
rpoC	484	W	G_2		G_1	G_1	
rpoC	485	D	H_1N_1	Y_1	N_1Y_1		
rpoC	491	I	T_1V_2	T_2	T_1V_1	T_2	
rpoC	516	L	P_2		P_1		
rpoC	519	G	D_1		D_1	V_1	
rpoC	521	A	D_1		D_1		
rpoC	525	H	N_1	Q_1	Q_1		
rpoC	527	L	V_1		V_1	V_8	
rpoC	698	N	$H_1K_1S_1$	H_1S_1	$H_1K_1S_1$	K_1	
rpoC	734	A			V_1	V_2	
rpoC	1040	P		R_1	$R_1S_1T_1$		
rpoC	1252	V			L_1	M_4	

Table 3.7: The list of putative compensatory mutations identified by our approach applied to the *mtu_broad* and *mtu173* datasets, identified in one of our two datasets and also reported in at least one of the three recent articles, or reported in at least two of the articles. The first two columns correspond to the gene name, and the reference amino acid, respectively. The next three columns provide brief descriptions of the mutations identified in the three recent studies: by Comas et al. (2012), de Vos et al. (2013) and Casali et al. (2014), respectively. The last two columns list the mutations identified based on our two case studies. Each mutation's description comprises of the reference amino acid, the position of the mutation in the gene, and different amino-acid variants of the mutation among the strains. For each mutation, the lower indexes indicate the number of strains, in the corresponding dataset, possessing the corresponding amino-acid variant of the mutation.

notypes with point mutations. In this work, we also present weighted support (WS) and tree-generalized hypergeometric (TGH) score — two new statistical scores — which employ phylogenetic information. As a part of this work, we also present yet another score, called Rank-based metascore (RBM) to combine multiple scores, thus compromising for weak points of the individual scores being

combined.

In order to test our approach, we prepared one dataset for *S. aureus* and two datasets for *M. tuberculosis*. The presented results demonstrate usefulness of our approach to identify drug-resistance associated mutations based on publicly available sequencing data. In particular, we were able to re-identify most of the known drug-resistance associations. Our results also support the phenomena previously reported in the literature, such as: (i) drug resistance-associated mutations tend to have multiple variants observed; or (ii) drug resistance associated mutations tend to cluster together in close genomic proximity.

Moreover, since most of the recent studies on the subject of compensatory mutations and in general drug resistance-associated mutations used in-house sequenced bacteria, we achieved our promising results basing our analysis solely on freely available public data.

The presented results also suggest that tree-aware methods (WS and TGH) perform better than methods which do not incorporate phylogenetic information. The results also show that the RBM score outperforms the individual scores in most of the settings. In particular, the RBM (ALL) score performed better than any tree-ignorant score in all the experiments.

Finally, despite some promising results, the presented tool has some limitations. First, it does not take into account epistatic interactions between mutations. Second, it takes into account only genomic changes, ignoring levels of gene expression. Thirdly, it provides putative *in silico* associations which should be subjected to further investigation in wet lab experiments.

The tools, case-study input data and the obtained results are available at the website of this project, `http://bioputer.mimuw.edu.pl/gwamar`.

"Science knows no country because knowledge belongs to humanity, and is the torch which illuminates the world. Science is the highest personification of the nation because that nation will remain the first which carries the furthest the works of thought and intelligence."

Louis Pasteur, Toast at banquet of the International Congress of Sericulture, 1876

4

Conclusions

This thesis is devoted to our work on exploring the potential of using whole-genome comparative analysis to deepen our knowledge on drug resistance mechanisms at the molecular level. In particular, we approach the problem of identifying drug resistance-associated mutations by comparative analysis of multiple bacterial genomes.

The continuous progress in whole-genome sequencing technologies makes the bacterial genome sequencing more affordable. As a consequence, large amount of genomic information is being generated and made publicly available. This allows for studying the genomic basis for different aspects of bacterial phenotype, such as drug resistance, virulence, and interactions with the host.

However, one issue we faced using publicly available genomes of bacterial strains is that their annotations are often inconsistent and of poor quality. This phenomenon has also been reported previously as a potential cause of bias for various analysis of the genome. For example, in the context of drug resistance, a missing gene annotation, may lead to a wrong conclusion, that the corresponding drug resistance mechanism, encoded by the gene, is absent in the bacteria. It has been shown also that the inconsistent genome annotations result from using

different annotations strategies in different sequencing laboratories.

4.1 Comparative genome annotations

In chapter 2, we present our work addressing this issue. In particular, we describe CAMBer and its highly optimized version, eCAMBer. These are tools that we have developed to improve the consistency and overall quality of bacterial genome annotations by comparative genome annotation. In its key step, called the closure procedure, eCAMBer tries to transfer gene annotations among all considered bacterial strains. The underlying idea behind the efficient implementation of the procedure in eCAMBer is to avoid redundant BLAST queries. Moreover, eCAMBer supports multiple-threading for all its procedures. This allows eCAMBer to be much faster than CAMBer and its other competitors — Mugsy-Annotator and the GMV pipeline — making it applicable to datasets comprising hundreds of bacterial strains.

In order to assess the impact of using eCAMBer on the quality of annotations, we applied it on the dataset of 20 *E. coli* strains, comprising genome sequences and annotations from the PATRIC database. As a gold standard for this dataset we used genome annotations manually curated by biologists. The results showed that eCAMBer improved the quality of the input annotations.

Although the development of eCAMBer was motivated by our research on identifying genetic variations associated with drug resistance, it may be used in other contexts where high-quality annotations of bacterial strains are needed. We expect eCAMBer to be a valuable tool for the research community working on comparative analysis of multiple bacterial strains.

We expect that, with the increasing amount of genomic data being generated, the need for similar tools will continue to grow. For example, to date, there are already about three thousand sequenced *S. aureus* genomes. Comparative analysis of such large datasets may be difficult for all currently available tools. One promising approach to achieve higher efficiency may rely on the idea of compressive genomics, which can greatly speed up running time of BLAST queries when the target genome sequences are highly similar, as in the case of closely related bacterial strains. We leave this, however, as one of the potential directions for further research.

4.2 Drug resistance-associated mutations

In chapter 3, we present our work on identifying the mutations associated with drug resistance based on comparative analysis of multiple bacterial strains. Thus, we tackle a general problem of associating genotype with phenotype. We describe GWAMAR, the tool we have developed to support this type of analysis.

The tool is designed as a pipeline. The input genomic data comprises genome sequences and annotations for a set of bacterial strains. The input phenotype data consists of information on drug susceptibility collected for the set of strains in question.

Then, in its first step, it employs eCAMBer to pre-process the genomic data, and to generate the set of genetic variations (mutations) among the set of considered strains, which could potentially be associated with drug resistance. This includes, the point amino-acid mutations, point promoter mutations (-50bp upstream the corresponding TIS) and gene gain/losses. In the default setting, we exclude from the analysis synonymous mutations, since, according to our knowledge, there is no evidence they may be related to drug resistance in bacteria.

The essential step of GWAMAR is scoring of the set of mutations with regard to the phenotype data. The goal of this procedure is to achieve a sorted list of mutations, such that, the mutations with the highest score are most likely to be associated with drug resistance. GWAMAR implements several statistics, such as mutual information, hypergeometric score, and odds ratio. As a part of this work we investigate the potential of scoring which employs phylogenetic information, represented as a tree, to strengthen the signal in data, which comes from the presence of the underlying association. We propose two such statistics, weighted support (WS) and the tree-generalized hypergeometric (TGH) score. Since it is not clear that trees are the best data structures to represent the phylogenetic information, one potential direction of the research could be to test the usability of other data structures, such as networks, to represent the phylogenetic information.

Furthermore, we introduced Rank-based metascore (RBM) for combining multiple scores into one in order to compromise between different approaches used to define different individual scores being combined. Our results demonstrate that, the RBM score outperform the individual scores in most of the settings. Specifically, the RBM (ALL) score, which combines all the scores considered in

this work, outperformed all the tree-ignorant scores considered separately.

We demonstrate the usability of the tool based on three datasets we prepared, one for *S. aureus* and two for *M. tuberculosis*. Applying GWAMAR to these dataset we show its ability to re-identify the known, gold standard associations. Moreover, we identify some putative associations, which haven't been yet reported. These *in silico* predictions may attract the attention of the experimental research community to test them experimentally in the labs.

Even though, in this work we focus solely on the problem of identifying drug resistance-associated mutations in bacteria, very similar approaches can be developed to study drug resistance in viruses, parasites or cancer. Moreover, in our opinion, the tool could be successfully applied to other types of phenotype data, such as virulence.

Finally, we expect that, with the increasing amount of genomic data being generated, more studies on whole-genome comparative approaches will be published in the context of drug resistance.

References

A Palleja, A., Harrington, E. D., Bork, P., 2008. Large gene overlaps in prokaryotic genomes: result of functional constraints or mispredictions?, BMC Genomics, 9(1), p. 335.

Alam, M. T., Petit, R. A., Crispell, E. K., Thornton, T. A., Conneely, K. N., Jiang, Y., Satola, S. W., Read, T. D., 2014. Dissecting vancomycin-intermediate resistance in staphylococcus aureus using genome-wide association, Genome Biology and Evolution, 6(5), p. 1174.

Alanis, A. J., 2005. Resistance to antibiotics: are we in the post-antibiotic era?, Archives of Medical Research, 36(6), p. 697.

Ali, A., 2013. Microbial comparative genomics: An overview of tools and insights into the genus corynebacterium, Journal of Bacteriology & Parasitology, 04(02).

Amir, A., Rana, K., Arya, A., Kapoor, N., Kumar, H., Siddiqui, M. A., 2014. Mycobacterium tuberculosis h37rv: In silico drug targets identification by metabolic pathways analysis, International Journal of Evolutionary Biology, 2014, p. e284170.

Andersson, D. I., Hughes, D., 2010. Antibiotic resistance and its cost: is it possible to reverse resistance?, Nature Reviews. Microbiology, 8(4), p. 260.

Angiuoli, S. V., Hotopp, J. C. D., Salzberg, S. L., Tettelin, H., 2011. Improving pan-genome annotation using whole genome multiple alignment, BMC Bioinformatics, 12(1), p. 272.

Angiuoli, S. V., Salzberg, S. L., 2011. Mugsy: fast multiple alignment of closely related whole genomes, Bioinformatics, 27(3), p. 334.

Bakheet, T. M., Doig, A. J., 2010. Properties and identification of antibiotic drug targets, BMC Bioinformatics, 11(1), p. 1.

Bakuła, A, Z., Napiórkowska, Rkowska, A., Bielecki, J., Augustynowicz-Kopeć, Ewa, Zwolska, Z., Jagielski, T., 2013. Mutations in the embB gene and their association with ethambutol resistance in multidrug-resistant Mycobacterium tu-

berculosis clinical isolates from poland, BioMed Research International, 2013, p. e167954.

Benson, D. A., Cavanaugh, M., Clark, K., Karsch-Mizrachi, I., Lipman, D. J., Ostell, J., Sayers, E. W., 2013. GenBank, Nucleic acids research, 41(Database issue), p. D36.

Binnewies, T. T., Motro, Y., Hallin, P. F., Lund, O., Dunn, D., La, T., Hampson, D. J., Bellgard, M., Wassenaar, T. M., Ussery, D. W., 2006. Ten years of bacterial genome sequencing: comparative-genomics-based discoveries, Functional & Integrative Genomics, 6(3), p. 165.

Blair, J. M. A., Webber, M. A., Baylay, A. J., Ogbolu, D. O., Piddock, L. J. V., 2015. Molecular mechanisms of antibiotic resistance, Nature Reviews Microbiology, 13(1), p. 42.

Blattner, F. R., Plunkett, r., G, Bloch, C. A., Perna, N. T., Burland, V., Riley, M., Collado-Vides, J., Glasner, J. D., Rode, C. K., Mayhew, G. F., Gregor, J., Davis, N. W., Kirkpatrick, H. A., Goeden, M. A., Rose, D. J., Mau, B., Shao, Y., 1997. The complete genome sequence of escherichia coli k-12, Science (New York, N.Y.), 277(5331), p. 1453.

Boehme, C. C., Nabeta, P., Hillemann, D., Nicol, M. P., Shenai, S., Krapp, F., Allen, J., Tahirli, R., Blakemore, R., Rustomjee, R., Milovic, A., Jones, M., O'Brien, S. M., Persing, D. H., Ruesch-Gerdes, S., Gotuzzo, E., Rodrigues, C., Alland, D., Perkins, M. D., 2010. Rapid molecular detection of tuberculosis and rifampin resistance, New England Journal of Medicine, 363(11), p. 1005.

Boto, L., 2010. Horizontal gene transfer in evolution: facts and challenges, Proceedings of the Royal Society of London B: Biological Sciences, 277(1683), p. 819.

Brandis, G., Hughes, D., 2013. Genetic characterization of compensatory evolution in strains carrying rpoB ser531leu, the rifampicin resistance mutation most frequently found in clinical isolates, Journal of Antimicrobial Chemotherapy, 68(11), p. 2493.

Bravo, L. T. C., Tuohy, M. J., Ang, C., Destura, R. V., Mendoza, M., Procop, G. W., Gordon, S. M., Hall, G. S., Shrestha, N. K., 2009. Pyrosequencing for rapid detection of mycobacterium tuberculosis resistance to rifampin, isoniazid, and fluoroquinolones, Journal of Clinical Microbiology, 47(12), p. 3985.

Buchfink, B., Xie, C., Huson, D. H., 2015. Fast and sensitive protein alignment using DIAMOND, Nature Methods, 12(1), p. 59.

Cabrera, C. P., Navarro, P., Huffman, J. E., Wright, A. F., Hayward, C., Campbell, H., Wilson, J. F., Rudan, I., Hastie, N. D., Vitart, V., Haley, C. S., 2012. Uncovering networks from genome-wide association studies via circular genomic permutation, G3 (Bethesda, Md.), 2(9), p. 1067.

Camacho, C., Coulouris, G., Avagyan, V., Ma, N., Papadopoulos, J., Bealer, K., Madden, T. L., 2009. BLAST+: architecture and applications, BMC Bioinformatics, 10(1), p. 421.

Campbell, E. A., Korzheva, N., Mustaev, A., Murakami, K., Nair, S., Goldfarb, A., Darst, S. A., 2001. Structural mechanism for rifampicin inhibition of bacterial rna polymerase, Cell, 104(6), p. 901.

Casali, N., Nikolayevskyy, V., Balabanova, Y., Harris, S. R., Ignatyeva, O., Kontsevaya, I., Corander, J., Bryant, J., Parkhill, J., Nejentsev, S., Horstmann, R. D., Brown, T., Drobniewski, F., 2014. Evolution and transmission of drug-resistant tuberculosis in a russian population, Nature Genetics, 46(3), p. 279.

CDC, 2013. Antibiotic resistance threats in the united states, 2013.

Cegielski, P., Nunn, P., Kurbatova, E. V., Weyer, K., Dalton, T. L., Wares, D. F., Iademarco, M. F., Castro, K. G., Raviglione, M., 2012. Challenges and controversies in defining totally drug-resistant tuberculosis, Emerging Infectious Diseases, 18(11), p. e2.

Chai, J., Kora, G., Ahn, T.-H., Hyatt, D., Pan, C., 2014. Functional phylogenomics analysis of bacteria and archaea using consistent genome annotation with UniFam, BMC Evolutionary Biology, 14(1), p. 207.

Chong, C. R., Sullivan, D. J., 2007. New uses for old drugs, Nature, 448(7154), p. 645.

Chopra, I., Roberts, M., 2001. Tetracycline antibiotics: mode of action, applications, molecular biology, and epidemiology of bacterial resistance, Microbiology and Molecular Biology Reviews: MMBR, 65(2), p. 232.

Chung, B. K.-S., Dick, T., Lee, D.-Y., 2013. In silico analyses for the discovery of tuberculosis drug targets, Journal of Antimicrobial Chemotherapy, p. dkt273.

Clarke, G. M., Anderson, C. A., Pettersson, F. H., Cardon, L. R., Morris, A. P., Zondervan, K. T., 2011. Basic statistical analysis in genetic case-control studies, Nature protocols, 6(2), p. 121.

Cock, P. J. A., Whitworth, D. E., 2010. Evolution of relative reading frame bias in unidirectional prokaryotic gene overlaps, Molecular Biology and Evolution, 27(4), p. 753.

Cole, S. T., Brosch, R., Parkhill, J., Garnier, T., Churcher, C., Harris, D., Gordon, S. V., Eiglmeier, K., Gas, S., Barry, r., C E, Tekaia, F., Badcock, K., Basham, D., Brown, D., Chillingworth, T., Connor, R., Davies, R., Devlin, K., Feltwell, T., Gentles, S., Hamlin, N., Holroyd, S., Hornsby, T., Jagels, K., Krogh, A., McLean, J., Moule, S., Murphy, L., Oliver, K., Osborne, J., Quail, M. A., Rajandream, M. A., Rogers, J., Rutter, S., Seeger, K., Skelton, J., Squares, R., Squares, S., Sulston, J. E., Taylor, K., Whitehead, S., Barrell, B. G., 1998. Deciphering the biology of mycobacterium tuberculosis from the complete genome sequence, Nature, 393(6685), p. 537.

Comas, I., Borrell, S., Roetzer, A., Rose, G., Malla, B., Kato-Maeda, M., Galagan, J., Niemann, S., Gagneux, S., 2012. Whole-genome sequencing of rifampicin-resistant mycobacterium tuberculosis strains identifies compensatory mutations in RNA polymerase genes, Nature Genetics, 44(1), p. 106.

Connell, S. R., Tracz, D. M., Nierhaus, K. H., Taylor, D. E., 2003. Ribosomal protection proteins and their mechanism of tetracycline resistance, Antimicrobial Agents and Chemotherapy, 47(12), p. 3675.

Corona, F., Martinez, J. L., 2013. Phenotypic resistance to antibiotics, Antibiotics, 2(2), p. 237.

Cox, G., Wright, G. D., 2013. Intrinsic antibiotic resistance: Mechanisms, origins, challenges and solutions, International Journal of Medical Microbiology, 303(6–7), p. 287.

Daniels, N. M., Gallant, A., Peng, J., Cowen, L. J., Baym, M., Berger, B., 2013. Compressive genomics for protein databases, Bioinformatics, 29(13), p. i283.

Daubin, V., Moran, N. A., Ochman, H., 2003. Phylogenetics and the cohesion of bacterial genomes, Science, 301(5634), p. 829.

Davies, G., Tenesa, A., Payton, A., Yang, J., Harris, S. E., Liewald, D., Ke, X., Le Hellard, S., Christoforou, A., Luciano, M., McGhee, K., Lopez, L., Gow, A. J., Corley, J., Redmond, P., Fox, H. C., Haggarty, P., Whalley, L. J., McNeill, G., Goddard, M. E., Espeseth, T., Lundervold, A. J., Reinvang, I., Pickles, A., Steen, V. M., Ollier, W., Porteous, D. J., Horan, M., Starr, J. M., Pendleton, N., Visscher, P. M., Deary, I. J., 2011. Genome-wide association studies establish that human intelligence is highly heritable and polygenic, Molecular Psychiatry, 16(10), p. 996.

Davies, J., Davies, D., 2010. Origins and evolution of antibiotic resistance, Microbiology and Molecular Biology Reviews : MMBR, 74(3), p. 417.

de Vos, M., Muller, B., Borrell, S., Black, P. A., van Helden, P. D., Warren, R. M., Gagneux, S., Victor, T. C., 2013. Putative compensatory mutations in

the rpoC gene of rifampin-resistant mycobacterium tuberculosis are associated with ongoing transmission, Antimicrobial Agents and Chemotherapy, 57(2), p. 827.

Drawz, S. M., Bonomo, R. A., 2010. Three decades of beta-lactamase inhibitors, Clinical Microbiology Reviews, 23(1), p. 160.

Dunbar, J., Cohn, J. D., Wall, M. E., 2011. Consistency of gene starts among burkholderia genomes, BMC Genomics, 12(1), p. 125.

Dutheil, J. Y., 2012. Detecting coevolving positions in a molecule: why and how to account for phylogeny, Briefings in Bioinformatics, 13(2), p. 228.

Džidić, S., Šušković, J., Kos, B., 2008. Antibiotic resistance mechanisms in bacteria: biochemical and genetic aspects., Food Technology and Biotechnology, 46(1), p. 11.

ECDC, 2012. Antimicrobial resistance surveillance in Europe, ECDC.

Ederveen, T. H. A., Overmars, L., van Hijum, S. A. F. T., 2013. Reduce manual curation by combining gene predictions from multiple annotation engines, a case study of start codon prediction, PLoS ONE, 8(5), p. e63523.

Edgar, R. C., 2004. MUSCLE: multiple sequence alignment with high accuracy and high throughput, Nucleic Acids Research, 32(5), p. 1792.

Farhat, M. R., Shapiro, B. J., Kieser, K. J., Sultana, R., Jacobson, K. R., Victor, T. C., Warren, R. M., Streicher, E. M., Calver, A., Sloutsky, A., Kaur, D., Posey, J. E., Plikaytis, B., Oggioni, M. R., Gardy, J. L., Johnston, J. C., Rodrigues, M., Tang, P. K. C., Kato-Maeda, M., Borowsky, M. L., Muddukrishna, B., Kreiswirth, B. N., Kurepina, N., Galagan, J., Gagneux, S., Birren, B., Rubin, E. J., Lander, E. S., Sabeti, P. C., Murray, M., 2013. Genomic analysis identifies targets of convergent positive selection in drug-resistant mycobacterium tuberculosis, Nature Genetics, 45(10), p. 1183.

Felciano, R. M., Bavari, S., Richards, D. R., Billaud, J.-N., Warren, T., Panchal, R., Krämer, A., 2013. Predictive systems biology approach to broad-spectrum, host-directed drug target discovery in infectious diseases, Pacific Symposium on Biocomputing. Pacific Symposium on Biocomputing, p. 17.

Felsenstein, J., 2005. PHYLIP (Phylogeny Inference Package) version 3.6, Department of Genome Sciences, University of Washington, Seattle.

Ferrero, L., Cameron, B., Crouzet, J., 1995. Analysis of gyrA and grlA mutations in stepwise-selected ciprofloxacin-resistant mutants ofStaphylococcus aureus., Antimicrobial Agents and Chemotherapy, 39(7), p. 1554.

Gillespie, J. J., Wattam, A. R., Cammer, S. A., Gabbard, J. L., Shukla, M. P., Dalay, O., Driscoll, T., Hix, D., Mane, S. P., Mao, C., Nordberg, E. K., Scott, M., Schulman, J. R., Snyder, E. E., Sullivan, D. E., Wang, C., Warren, A., Williams, K. P., Xue, T., Yoo, H. S., Zhang, C., Zhang, Y., Will, R., Kenyon, R. W., Sobral, B. W., 2011. PATRIC: the comprehensive bacterial bioinformatics resource with a focus on human pathogenic species, Infection and Immunity, 79(11), p. 4286.

Guindon, S., Dufayard, J.-F., Lefort, V., Anisimova, M., Hordijk, W., Gascuel, O., 2010. New algorithms and methods to estimate maximum-likelihood phylogenies: assessing the performance of PhyML 3.0, Systematic Biology, 59(3), p. 307.

Habib, F., Johnson, A. D., Bundschuh, R., Janies, D., 2007. Large scale genotype-phenotype correlation analysis based on phylogenetic trees, Bioinformatics (Oxford, England), 23(7), p. 785.

Hasman, H., Saputra, D., Sicheritz-Ponten, T., Lund, O., Svendsen, C. A., Frimodt-Møller, N., Aarestrup, F. M., 2014. Rapid whole-genome sequencing for detection and characterization of microorganisms directly from clinical samples, Journal of Clinical Microbiology, 52(1), p. 139.

Hazbón, M. H., Motiwala, A. S., Cavatore, M., Brimacombe, M., Whittam, T. S., Alland, D., 2008. Convergent evolutionary analysis identifies significant mutations in drug resistance targets of mycobacterium tuberculosis, Antimicrobial Agents and Chemotherapy, 52(9), p. 3369.

Hiller, N. L., Janto, B., Hogg, J. S., Boissy, R., Yu, S., Powell, E., Keefe, R., Ehrlich, N. E., Shen, K., Hayes, J., Barbadora, K., Klimke, W., Dernovoy, D., Tatusova, T., Parkhill, J., Bentley, S. D., Post, J. C., Ehrlich, G. D., Hu, F. Z., 2007. Comparative genomic analyses of seventeen streptococcus pneumoniae strains: Insights into the pneumococcal supragenome, Journal of Bacteriology, 189(22), p. 8186.

Hu, M., Nandi, S., Davies, C., Nicholas, R. A., 2005. High-level chromosomally mediated tetracycline resistance in Neisseria gonorrhoeae results from a point mutation in the rpsJ gene encoding ribosomal protein S10 in combination with the mtrR and penB resistance determinants, Antimicrobial Agents and Chemotherapy, 49(10), p. 4327.

Hyatt, D., Chen, G.-L., LoCascio, P. F., Land, M. L., Larimer, F. W., Hauser, L. J., 2010. Prodigal: prokaryotic gene recognition and translation initiation site identification, BMC Bioinformatics, 11, p. 119.

Ioerger, T. R., Koo, S., No, E.-G., Chen, X., Larsen, M. H., Jacobs, J., William R., Pillay, M., Sturm, A. W., Sacchettini, J. C., 2009. Genome analysis of multi-

and extensively-drug-resistant tuberculosis from KwaZulu-Natal, south africa, PLoS ONE, 4(11), p. e7778.

Jia, J., Zhu, F., Ma, X., Cao, Z. W., Li, Y. X., Chen, Y. Z., 2009. Mechanisms of drug combinations: interaction and network perspectives, Nature Reviews Drug Discovery, 8(2), p. 111.

Juhas, M., Eberl, L., Church, G. M., 2012. Essential genes as antimicrobial targets and cornerstones of synthetic biology, Trends in Biotechnology, 30(11), p. 601.

Karp, P. D., Keseler, I. M., Shearer, A., Latendresse, M., Krummenacker, M., Paley, S. M., Paulsen, I., Collado-Vides, J., Gama-Castro, S., Peralta-Gil, M., Santos-Zavaleta, A., Peñaloza-Spínola, M. I., Bonavides-Martinez, C., Ingraham, J., 2007. Multidimensional annotation of the escherichia coli K-12 genome, Nucleic Acids Research, 35(22), p. 7577.

Kasif, S., Steffen, M., 2010. Biochemical networks: The evolution of gene annotation, Nature Chemical Biology, 6(1), p. 4.

Khan, A. I., Dinh, D. M., Schneider, D., Lenski, R. E., Cooper, T. F., 2011. Negative epistasis between beneficial mutations in an evolving bacterial population, Science, 332(6034), p. 1193.

Khor, C. C., Hibberd, M. L., 2012. Host-pathogen interactions revealed by human genome-wide surveys, Trends in genetics: TIG, 28(5), p. 233.

Kinnings, S. L., Liu, N., Buchmeier, N., Tonge, P. J., Xie, L., Bourne, P. E., 2009. Drug discovery using chemical systems biology: Repositioning the safe medicine comtan to treat multi-drug and extensively drug resistant tuberculosis, PLoS Computational Biology, 5(7).

Klassen, J. L., Currie, C. R., 2013. ORFcor: identifying and accommodating ORF prediction inconsistencies for phylogenetic analysis, PLoS ONE, 8(3), p. e58387.

Klopper, M., Warren, R. M., Hayes, C., Gey van Pittius, N. C., Streicher, E. M., Müller, B., Sirgel, F. A., Chabula-Nxiweni, M., Hoosain, E., Coetzee, G., David van Helden, P., Victor, T. C., Trollip, A. P., 2013. Emergence and spread of extensively and totally drug-resistant tuberculosis, south africa, Emerging Infectious Diseases, 19(3), p. 449.

Knox, C., Law, V., Jewison, T., Liu, P., Ly, S., Frolkis, A., Pon, A., Banco, K., Mak, C., Neveu, V., Djoumbou, Y., Eisner, R., Guo, A. C., Wishart, D. S., 2011. DrugBank 3.0: a comprehensive resource for 'omics' research on drugs, Nucleic Acids Research, 39(Suppl 1), p. D1035.

Koch, A., Mizrahi, V., Warner, D. F., 2014. The impact of drug resistance on mycobacterium tuberculosis physiology: what can we learn from rifampicin?, Emerging Microbes & Infections, 3(3), p. e17.

Kuroda, M., Ohta, T., Uchiyama, I., Baba, T., Yuzawa, H., Kobayashi, I., Cui, L., Oguchi, A., Aoki, K., Nagai, Y., Lian, J., Ito, T., Kanamori, M., Matsumaru, H., Maruyama, A., Murakami, H., Hosoyama, A., Mizutani-Ui, Y., Takahashi, N. K., Sawano, T., Inoue, R., Kaito, C., Sekimizu, K., Hirakawa, H., Kuhara, S., Goto, S., Yabuzaki, J., Kanehisa, M., Yamashita, A., Oshima, K., Furuya, K., Yoshino, C., Shiba, T., Hattori, M., Ogasawara, N., Hayashi, H., Hiramatsu, K., 2001. Whole genome sequencing of meticillin-resistant staphylococcus aureus, Lancet, 357(9264), p. 1225.

Köser, C. U., Ellington, M. J., Peacock, S. J., 2014. Whole-genome sequencing to control antimicrobial resistance, Trends in Genetics, 30(9), p. 401.

Laing, C. R., Zhang, Y., Thomas, J. E., Gannon, V. P., 2011. Everything at once: Comparative analysis of the genomes of bacterial pathogens, Veterinary Microbiology, 153(1-2), p. 13.

Lambert, G., Estévez-Salmeron, L., Oh, S., Liao, D., Emerson, B. M., Tlsty, T. D., Austin, R. H., 2011. An analogy between the evolution of drug resistance in bacterial communities and malignant tissues, Nature Reviews Cancer, 11(5), p. 375.

Lange, C., Abubakar, I., Alffenaar, J.-W. C., Bothamley, G., Caminero, J. A., Carvalho, A. C. C., Chang, K.-C., Codecasa, L., Correia, A., Crudu, V., Davies, P., Dedicoat, M., Drobniewski, F., Duarte, R., Ehlers, C., Erkens, C., Goletti, D., Günther, G., Ibraim, E., Kampmann, B., Kuksa, L., de Lange, W., van Leth, F., van Lunzen, J., Matteelli, A., Menzies, D., Monedero, I., Richter, E., Rüsch-Gerdes, S., Sandgren, A., Scardigli, A., Skrahina, A., Tortoli, E., Volchenkov, G., Wagner, D., van der Werf, M. J., Williams, B., Yew, W.-W., Zellweger, J.-P., Cirillo, D. M., 2014. Management of patients with multidrug-resistant/extensively drug-resistant tuberculosis in europe: a TBNET consensus statement, The European Respiratory Journal, 44(1), p. 23.

Laxminarayan, R., Duse, A., Wattal, C., Zaidi, A. K. M., Wertheim, H. F. L., Sumpradit, N., Vlieghe, E., Hara, G. L., Gould, I. M., Goossens, H., Greko, C., So, A. D., Bigdeli, M., Tomson, G., Woodhouse, W., Ombaka, E., Peralta, A. Q., Qamar, F. N., Mir, F., Kariuki, S., Bhutta, Z. A., Coates, A., Bergstrom, R., Wright, G. D., Brown, E. D., Cars, O., 2013. Antibiotic resistance-the need for global solutions, The Lancet. Infectious Diseases, 13(12), p. 1057.

Levy, S. B., 2002. Factors impacting on the problem of antibiotic resistance, Journal of Antimicrobial Chemotherapy, 49(1), p. 25.

Levy, S. B., Marshall, B., 2004. Antibacterial resistance worldwide: causes, challenges and responses, Nature Medicine, 10(12 Suppl), p. S122.

Lew, J. M., Kapopoulou, A., Jones, L. M., Cole, S. T., 2011. TubercuList–10 years after, Tuberculosis (Edinburgh, Scotland), 91(1), p. 1.

Lewis, K., 2013. Platforms for antibiotic discovery, Nature Reviews Drug Discovery, 12(5), p. 371.

Ling, L. L., Schneider, T., Peoples, A. J., Spoering, A. L., Engels, I., Conlon, B. P., Mueller, A., Schäberle, T. F., Hughes, D. E., Epstein, S., Jones, M., Lazarides, L., Steadman, V. A., Cohen, D. R., Felix, C. R., Fetterman, K. A., Millett, W. P., Nitti, A. G., Zullo, A. M., Chen, C., Lewis, K., 2015. A new antibiotic kills pathogens without detectable resistance, Nature, advance online publication.

Liu, B., Pop, M., 2009. ARDB–antibiotic resistance genes database, Nucleic Acids Research, 37(Database issue), p. D443.

Loh, P.-R., Baym, M., Berger, B., 2012. Compressive genomics, Nature Biotechnology, 30(7), p. 627.

Loman, N. J., Constantinidou, C., Chan, J. Z. M., Halachev, M., Sergeant, M., Penn, C. W., Robinson, E. R., Pallen, M. J., 2012. High-throughput bacterial genome sequencing: an embarrassment of choice, a world of opportunity, Nature Reviews. Microbiology, 10(9), p. 599.

Lukjancenko, O., Wassenaar, T. M., Ussery, D. W., 2010. Comparison of 61 sequenced escherichia coli genomes, Microbial ecology, 60(4), p. 708.

Lázár, V., Nagy, I., Spohn, R., Csörgő, B., Györkei, ., Nyerges, ., Horváth, B., Vörös, A., Busa-Fekete, R., Hrtyan, M., Bogos, B., Méhi, O., Fekete, G., Szappanos, B., Kégl, B., Papp, B., Pál, C., 2014. Genome-wide analysis captures the determinants of the antibiotic cross-resistance interaction network, Nature Communications, 5.

Magiorakos, A.-P., Srinivasan, A., Carey, R. B., Carmeli, Y., Falagas, M. E., Giske, C. G., Harbarth, S., Hindler, J. F., Kahlmeter, G., Olsson-Liljequist, B., Paterson, D. L., Rice, L. B., Stelling, J., Struelens, M. J., Vatopoulos, A., Weber, J. T., Monnet, D. L., 2012. Multidrug-resistant, extensively drug-resistant and pandrug-resistant bacteria: an international expert proposal for interim standard definitions for acquired resistance, Clinical Microbiology and Infection: The Official Publication of the European Society of Clinical Microbiology and Infectious Diseases, 18(3), p. 268.

Maisnier-Patin, S., Andersson, D. I., 2004. Adaptation to the deleterious effects of antimicrobial drug resistance mutations by compensatory evolution, Research in Microbiology, 155(5), p. 360.

Malik, S., Willby, M., Sikes, D., Tsodikov, O. V., Posey, J. E., 2012. New insights into fluoroquinolone resistance in mycobacterium tuberculosis: Functional genetic analysis of gyrA and gyrB mutations, PLoS ONE, 7(6), p. e39754.

Manolio, T. A., 2010. Genomewide association studies and assessment of the risk of disease, New England Journal of Medicine, 363(2), p. 166.

Matteelli, A., Roggi, A., Carvalho, A. C., 2014. Extensively drug-resistant tuberculosis: epidemiology and management, Clinical Epidemiology, 6, p. 111.

McAleese, F. M., Foster, T. J., 2003. Analysis of mutations in the Staphylococcus aureus clfB promoter leading to increased expression, Microbiology, 149(1), p. 99.

Missiakas, D., Georgopoulos, C., Raina, S., 1993. The escherichia coli heat shock gene htpY: mutational analysis, cloning, sequencing, and transcriptional regulation, Journal of bacteriology, 175(9), p. 2613.

Nielsen, C. B., Cantor, M., Dubchak, I., Gordon, D., Wang, T., 2010. Visualizing genomes: techniques and challenges, Nature Methods, 7, p. S5.

Nikaido, H., 2009. Multidrug resistance in bacteria, Annual review of biochemistry, 78, p. 119.

O'Neill, A. J., Huovinen, T., Fishwick, C. W. G., Chopra, I., 2006. Molecular genetic and structural modeling studies of Staphylococcus aureus RNA polymerase and the fitness of rifampin resistance genotypes in relation to clinical prevalence, Antimicrobial Agents and Chemotherapy, 50(1), p. 298.

Overbeek, R., Bartels, D., Vonstein, V., Meyer, F., 2007. Annotation of bacterial and archaeal genomes: improving accuracy and consistency, Chemical Reviews, 107(8), p. 3431.

Pati, A., Ivanova, N. N., Mikhailova, N., Ovchinnikova, G., Hooper, S. D., Lykidis, A., Kyrpides, N. C., 2010. GenePRIMP: a gene prediction improvement pipeline for prokaryotic genomes, Nature Methods, 7(6), p. 455.

Patra, S. K., Jain, A., Sherwal, B. L., Khanna, A., 2010. Rapid detection of mutation in RRDR of rpo b gene for rifampicin resistance in MDR-pulmonary tuberculosis by DNA sequencing, Indian journal of clinical biochemistry: IJCB, 25(3), p. 315.

Pavlović, V., Garg, A., Kasif, S., 2002. A bayesian framework for combining gene predictions, Bioinformatics (Oxford, England), 18(1), p. 19.

Pearson, H., 2006. Genetics: What is a gene?, Nature, 441(7092), p. 398.

Philippe, H., Douady, C. J., 2003. Horizontal gene transfer and phylogenetics, Current Opinion in Microbiology, 6(5), p. 498.

Poptsova, M. S., Gogarten, J. P., 2010. Using comparative genome analysis to identify problems in annotated microbial genomes, Microbiology, 156(Pt 7), p. 1909.

Projan, S. J., 2003. Why is big pharma getting out of antibacterial drug discovery?, Current Opinion in Microbiology, 6(5), p. 427.

Raman, K., Chandra, N., 2008. Mycobacterium tuberculosis interactome analysis unravels potential pathways to drug resistance, BMC Microbiology, 8(1), p. 234.

Read, T. D., Massey, R. C., 2014. Characterizing the genetic basis of bacterial phenotypes using genome-wide association studies: a new direction for bacteriology, Genome Medicine, 6(11), p. 109.

Richardson, E. J., Watson, M., 2013. The automatic annotation of bacterial genomes, Briefings in Bioinformatics, 14(1), p. 1.

Rost, B., 1999. Twilight zone of protein sequence alignments, Protein engineering, 12(2), p. 85.

Sabath, L. D., 1982. Mechanisms of resistance to beta-lactam antibiotics in strains of Staphylococcus aureus, Annals of Internal Medicine, 97(3), p. 339.

Sacerdot, C., Fayat, G., Dessen, P., Springer, M., Plumbridge, J. A., Grunberg-Manago, M., Blanquet, S., 1982. Sequence of a 1.26-kb DNA fragment containing the structural gene for e.coli initiation factor IF3: presence of an AUU initiator codon, The EMBO journal, 1(3), p. 311.

Salzberg, S. L., 2007. Genome re-annotation: a wiki solution?, Genome Biology, 8(1), p. 102.

Sandgren, A., Strong, M., Muthukrishnan, P., Weiner, B. K., Church, G. M., Murray, M. B., 2009. Tuberculosis drug resistance mutation database, PLoS Med, 6(2), p. e1000002.

Sauer, U., Heinemann, M., Zamboni, N., 2007. Getting closer to the whole picture, Science, 316(5824), p. 550.

Schmitz, F., Sadurski, R., Kray, A., Boos, M., Geisel, R., Kohrer, K., Verhoef, J., Fluit, A. C., 2000. Prevalence of macrolide-resistance genes in Staphylococcus aureus and Enterococcus faecium isolates from 24 European University Hospitals, Journal of Antimicrobial Chemotherapy, 45(6), p. 891.

Shah, S. P., McVicker, G. P., Mackworth, A. K., Rogic, S., Ouellette, B. F. F., 2003. GeneComber: combining outputs of gene prediction programs for improved results, Bioinformatics (Oxford, England), 19(10), p. 1296.

Shakil, S., Khan, R., Zarrilli, R., Khan, A. U., 2008. Aminoglycosides versus bacteria–a description of the action, resistance mechanism, and nosocomial battleground, Journal of Biomedical Science, 15(1), p. 5.

Shlaes, D. M., Projan, S. J., Edwards, J. E., 2004. Antibiotic discovery: state of the state., ASM News, 70(6), p. 275.

Smith, K. L. J., Saini, D., Bardarov, S., Larsen, M., Frothingham, R., Gandhi, N. R., Jacobs Jr., W. R., Sturm, A. W., Lee, S., 2014. Reduced virulence of an extensively drug-resistant outbreak strain of mycobacterium tuberculosis in a murine model, PLoS ONE, 9(4), p. e94953.

Stadler, Z. K., Thom, P., Robson, M. E., Weitzel, J. N., Kauff, N. D., Hurley, K. E., Devlin, V., Gold, B., Klein, R. J., Offit, K., 2010. Genome-wide association studies of cancer, Journal of Clinical Oncology, p. JCO.2009.25.7816.

Sykes, R., 2010. The 2009 garrod lecture: The evolution of antimicrobial resistance: a darwinian perspective, Journal of Antimicrobial Chemotherapy, p. dkq217.

Tatusova, T., Ciufo, S., Fedorov, B., O'Neill, K., Tolstoy, I., 2014. RefSeq microbial genomes database: new representation and annotation strategy, Nucleic Acids Research, 42(Database issue), p. D553.

Touchon, M., Hoede, C., Tenaillon, O., Barbe, V., Baeriswyl, S., Bidet, P., Bingen, E., Bonacorsi, S., Bouchier, C., Bouvet, O., Calteau, A., Chiapello, H., Clermont, O., Cruveiller, S., Danchin, A., Diard, M., Dossat, C., Karoui, M. E., Frapy, E., Garry, L., Ghigo, J. M., Gilles, A. M., Johnson, J., Le Bouguénec, C., Lescat, M., Mangenot, S., Martinez-Jéhanne, V., Matic, I., Nassif, X., Oztas, S., Petit, M. A., Pichon, C., Rouy, Z., Ruf, C. S., Schneider, D., Tourret, J., Vacherie, B., Vallenet, D., Médigue, C., Rocha, E. P. C., Denamur, E., 2009. Organised genome dynamics in the escherichia coli species results in highly diverse adaptive paths, PLoS Genet, 5(1), p. e1000344.

Trauner, A., Borrell, S., Reither, K., Gagneux, S., 2014. Evolution of drug resistance in tuberculosis: Recent progress and implications for diagnosis and therapy, Drugs, 74(10), p. 1063.

Turnidge, J., Kahlmeter, G., Kronvall, G., 2006. Statistical characterisation of bacterial wild-type MIC value distributions and the determination of epidemiological cut-off values, Clinical Microbiology and Infection, 12(5), p. 418.

Udwadia, Z. F., Amale, R. A., Ajbani, K. K., Rodrigues, C., 2012. Totally drug-resistant tuberculosis in india, Clinical Infectious Diseases, 54(4), p. 579.

Vallenet, D., Labarre, L., Rouy, Z., Barbe, V., Bocs, S., Cruveiller, S., Lajus, A., Pascal, G., Scarpelli, C., Médigue, C., 2006. MaGe: a microbial genome annotation system supported by synteny results, Nucleic Acids Research, 34(1), p. 53.

Velayati, A. A., Farnia, P., Masjedi, M. R., 2013. The totally drug resistant tuberculosis (TDR-TB), International Journal of Clinical and Experimental Medicine, 6(4), p. 307.

Velayati, A. A., Masjedi, M. R., Farnia, P., Tabarsi, P., Ghanavi, J., Ziazarifi, A. H., Hoffner, S. E., 2009. Emergence of new forms of totally drug-resistant tuberculosis bacilli: super extensively drug-resistant tuberculosis or totally drug-resistant strains in Iran, Chest, 136(2), p. 420.

Wall, M. E., Raghavan, S., Cohn, J. D., Dunbar, J., 2011. Genome majority vote improves gene predictions, PLoS Comput Biol, 7(11), p. e1002284.

Warnes, S. L., Highmore, C. J., Keevil, C. W., 2012. Horizontal transfer of antibiotic resistance genes on abiotic touch surfaces: Implications for public health, mBio, 3(6), p. e00489.

Wattam, A. R., Abraham, D., Dalay, O., Disz, T. L., Driscoll, T., Gabbard, J. L., Gillespie, J. J., Gough, R., Hix, D., Kenyon, R., Machi, D., Mao, C., Nordberg, E. K., Olson, R., Overbeek, R., Pusch, G. D., Shukla, M., Schulman, J., Stevens, R. L., Sullivan, D. E., Vonstein, V., Warren, A., Will, R., Wilson, M. J. C., Yoo, H. S., Zhang, C., Zhang, Y., Sobral, B. W., 2014. PATRIC, the bacterial bioinformatics database and analysis resource, Nucleic Acids Research, 42(D1), p. D581.

Weinstock, G. M., Peacock, S. J., 2014. Next-generation pathogen genomics, Genome Biology, 15(11), p. 528.

Werckenthin, C., Schwarz, S., Roberts, M. C., 1996. Integration of pT181-like tetracycline resistance plasmids into large staphylococcal plasmids involves IS257., Antimicrobial Agents and Chemotherapy, 40(11), p. 2542.

WHO, 2010. Guidelines for ATC classification and DDD assignment.

WHO, 2013. Global tuberculosis report, WHO.

WHO, 2014. Antimicrobial resistance: global report on surveillance 2014.

Wiesch, P. S. z., Engelstädter, J., Bonhoeffer, S., 2010. Compensation of fitness costs and reversibility of antibiotic resistance mutations, Antimicrobial Agents and Chemotherapy, 54(5), p. 2085.

Wolf, Y. I., Rogozin, I. B., Kondrashov, A. S., Koonin, E. V., 2001. Genome alignment, evolution of prokaryotic genome organization, and prediction of gene function using genomic context, Genome research, 11(3), p. 356.

Wong, L., Liu, G., 2010. Protein interactome analysis for countering pathogen drug resistance, Journal of Computer Science and Technology, 25(1), p. 124.

Wood, D. E., Lin, H., Levy-Moonshine, A., Swaminathan, R., Chang, Y.-C., Anton, B. P., Osmani, L., Steffen, M., Kasif, S., Salzberg, S. L., 2012. Thousands of missed genes found in bacterial genomes and their analysis with COMBREX, Biology Direct, 7(1), p. 37.

Woźniak, M., Tiuryn, J., Wong, L., 2012. An approach to identifying drug resistance associated mutations in bacterial strains, BMC Genomics, 13(Suppl 7), p. S23.

Woźniak, M., Tiuryn, J., Wong, L., 2014a. GWAMAR: Genome-wide assessment of mutations associated with drug resistance in bacteria, BMC Genomics, 15(Suppl 10), p. S10.

Woźniak, M., Wong, L., Tiuryn, J., 2011a. CAMBer: an approach to support comparative analysis of multiple bacterial strains, BMC Genomics, 12(Suppl 2), p. S6.

Woźniak, M., Wong, L., Tiuryn, J., 2011b. CAMBerVis: visualization software to support comparative analysis of multiple bacterial strains, Bioinformatics (Oxford, England), 27(23), p. 3313.

Woźniak, M., Wong, L., Tiuryn, J., 2014b. eCAMBer: efficient support for large-scale comparative analysis of multiple bacterial strains, BMC Bioinformatics, 15(1), p. 65.

Wright, G. D., 2011. Molecular mechanisms of antibiotic resistance, Chemical Communications (Cambridge, England), 47(14), p. 4055.

Wu, C., Li, S., Cui, Y., 2012. Genetic association studies: An information content perspective, Current Genomics, 13(7), p. 566.

Yada, T., Takagi, T., Totoki, Y., Sakaki, Y., Takaeda, Y., 2003. DIGIT: a novel gene finding program by combining gene-finders, Pacific Symposium on Biocomputing. Pacific Symposium on Biocomputing, p. 375.

Yu, J.-F., Xiao, K., Jiang, D.-K., Guo, J., Wang, J.-H., Sun, X., 2011. An integrative method for identifying the over-annotated protein-coding genes in microbial genomes, DNA research: an international journal for rapid publication of reports on genes and genomes, 18(6), p. 435.

Zhang, Q., Lambert, G., Liao, D., Kim, H., Robin, K., Tung, C.-k., Pourmand, N., Austin, R. H., 2011. Acceleration of emergence of bacterial antibiotic resistance in connected microenvironments, Science, 333(6050), p. 1764.

Zhou, H., Jin, J., Zhang, H., Yi, B., Woźniak, M., Wong, L., 2012. IntPath–an integrated pathway gene relationship database for model organisms and important pathogens, BMC Systems Biology, 6(Suppl 2), p. S2.

Zhou, H., Wong, L., 2011. Comparative analysis and assessment of m. tuberculosis h37rv protein-protein interaction datasets, BMC Genomics, 12(Suppl 3), p. S20.

Zhou, J., Rudd, K. E., 2013. EcoGene 3.0, Nucleic Acids Research, 41(D1), p. D613.

Zoraghi, R., Reiner, N. E., 2013. Protein interaction networks as starting points to identify novel antimicrobial drug targets, Current Opinion in Microbiology, 16(5), p. 566.

Zumla, A., Nahid, P., Cole, S. T., 2013. Advances in the development of new tuberculosis drugs and treatment regimens, Nature Reviews Drug Discovery, 12(5), p. 388.