FACULTY OF BIOSCIENCE ENGINEERING

A COMPUTATIONAL STUDY OF BACTERIA-PHAGE INTERACTIONS TO REVEAL DETERMINANTS OF PHAGE-HOST SPECIFICITY

Word count: 23,327

Dimitri Boeckaerts
Student number: 01202040

Promotors: Prof. dr. ir. Yves Briers, Prof. dr. Bernard De Baets
Tutors: Dr. ir. Michiel Stock, Ir. Bjorn Criel, Ir. Hans Gerstmans

Master's dissertation submitted to Ghent University in partial fulfilment of the requirements for the degree of Master of Science in Bioscience Engineering: Cell and Gene Biotechnology.
Academic year: 2017 - 2018

UNIVERSITEIT GENT

De auteur en promotor geven de toelating deze scriptie voor consultatie beschikbaar te stellen en delen ervan te kopiëren voor persoonlijk gebruik. Elk ander gebruik valt onder de beperkingen van het auteursrecht, in het bijzonder met betrekking tot de verplichting uitdrukkelijk de bron te vermelden bij het aanhalen van resultaten uit deze scriptie.

The author and promoter give the permission to use this thesis for consultation and to copy parts of it for personal use. Every other use is subject to the copyright laws, more specifically the source must be extensively specified when using results from this thesis.

Gent, June 7, 2018

The promotors,	The author,
Prof. dr. ir Yves Briers	Dimitri Boeckaerts
Prof. dr. Bernard De Baets	

Acknowledgments

Everything takes longer than you think.

- every thesis student ever.

The quote above is an important lesson I learned throughout this work. Still, even though I frequently saw evidence of that statement, I refused to believe it was a general rule. Of course, I've learned many more things during this work. I've come to known the marvelous world of bacteriophages. Arguably the smallest biological entities on planet Earth (and possibly beyond), yet capable of amazing things.

Although I am the author of this work, it cannot and should not be seen as a one-man's-effort. I would like to thank Bernard and Yves for giving me the opportunity to study an amazingly interesting subject. The fields of research touched upon in this thesis sparked my interest early on, and have become even greater interests throughout this year. Additionally, their useful insights along the way contributed to a successful end of this thesis. I would also like to express enormous gratitude to my tutors Michiel, Bjorn and Hans. Their continued enthusiasm, support and time investment had a big impact on the success of this dissertation. Every other week, I had the opportunity to bother them with questions, and their ideas often became important parts of this work. A special thanks as well to my dear sister Laura for taking the time to give feedback on the final chapter, while she had a million other things to do. Lastly, I would like to thank my computer and my brain for being able to keep up with all the work.

Samenvatting

Pathogene bacteriën worden in toenemende mate resistent tegen antibiotica. Door een gebrek aan ontwikkeling van nieuwe klassen antibiotica, alsook door een groter wordende resistentie, zijn er steeds minder middelen beschikbaar in de strijd tegen bacteriën. Hierdoor stijgt al jaren de nood naar alternatieve en meer specifieke therapeutische middelen. Een veelbelovende alternatieve oplossing voor dit probleem zijn de natuurlijke vijanden van bacteriën, de bacteriofagen. Deze bacteriële virussen infecteren vaak slechts één of enkele bacteriële species, of zelfs specifieke stammen. Fagen gebruiken hiervoor specifieke eiwitten die een geschikte bacteriële host kunnen herkennen. In dit werk werd faag-host specificiteit bestudeerd met twee computationele technieken: optimal transport en machine learning. Hierdoor draagt dit werk bij tot een beter begrip van faag-host specificiteit. Dit werk toont ook aan dat computationele tools gebruikt kunnen worden om faag-host specificiteit op een nieuwe manier te bestuderen. Deze tools kunnen potentieel voor verschillende applicaties gebruikt worden.

Een van de centrale vragen in dit werk is of specifieke determinanten van faag-host specificiteit geïdentificeerd kunnen worden in faag proteomen en proteïnen. Hiervoor werd optimal transport gebruikt, een techniek om similariteit te meten tussen faag proteomen en proteïnen. Als 'proof-of-concept' werd deze techniek gebruikt om afstanden te berekenen tussen faag proteomen om zo een classificatie boom te construeren. Daarna werd optimal transport toegepast om afstanden te berekenen tussen de proteïnen van drie fagen van de T7 virus groep. Hier werd geprobeerd om de proteïnen te identificeren die uniek waren tussen de drie faag proteomen. Dit werk toont aan dat afstanden berekend door optimal transport mogelijks een goede similariteitsmaat kunnen zijn tussen proteïnen en proteomen.

De proteïnen die fagen gebruiken om hun host te herkennen zijn vaak gelegen op zgn. tail fibers en/of tail spikes. In een tweede deel van dit werk werd tail fiber and tail spike proteïne data gebruikt om bacteriële hosts te voorspellen m.b.v. machine learning methoden. Verder werd getracht deze proteïnen te karakteriseren o.b.v. de 'features' die belangrijk zijn in de predicties. De ontwikkelde methode is in staat om drie verschillende bacteriële hosts te onderscheiden met accuraatheden tot 93%. Echter, om deze methoden verder uit te breiden is een betere annotatie van eiwit

data noodzakelijk. Daarnaast toont dit werk aan dat faag-host specificiteit complex is en het verschil in host specificiteit niet kan verklaard worden o.b.v. een beperkt aantal eenvoudige eigenschappen.

Trefwoorden: faag-host specificiteit, tail fiber eiwitten, optimal transport, machine learning.

Summary

Pathogenic bacteria become increasingly resistant to antibiotics. The lack of discovery of new classes of antibiotics along with the emerging resistance leads to a decreasing number of therapeutic options. As a result, the need for alternative, more specific approaches keeps growing year by year. A promising alternative approach to this problem are the natural enemies of bacteria, bacteriophages. These bacterial virusses are known to be highly specific to only one or a few bacterial species, or even particular strains. Phages employ specific proteins to recognize a suitable bacterial host. In this work, phage-host specificity is studied using two computational approaches: optimal transport and machine learning. In doing so, this work contributes to a broader understanding of phage-host specificity. It also shows that computational tools can be adopted to study host specificity in a new way. Potentially, these tools can be used for various applications.

One of the central questions of this work was whether determinants of phage-host specificity could be identified in phage proteomes and proteins. Therefore, optimal transport was applied as a technique to measure similarities between phage proteomes and proteins. As a proof-of-concept, the technique was first adopted to compute distances between phage proteomes in order to construct a classification tree. Secondly, optimal transport was used to compute distances between the proteins of three phages of the T7 virus group. Here, it was attempted to identify the proteins that are unique among the three phage proteomes. This work shows that distances computed by optimal transport can be a good measure for similarity between proteomes and proteins.

Often, the proteins used by phages to recognize a suitable bacterial host are located on tail fibers and/or tail spikes. In the second part of this work, tail fiber and tail spike protein data were used to predict bacterial hosts using machine learning methods. Furthermore, it was attempted to characterize the tail fiber and tail spike proteins based on features important in prediction. The methods developed in this work are able to discriminate between three classes of bacterial hosts with performances of up to 93%. However, to extend these methods even further, a better annotation of data is needed. Additionally, this work shows that phage-host specificity is complex and

differences in host specificity can not be explained only based on simple characteristics.

Keywords: phage-host specificity, tail fiber proteins, optimal transport, machine learning.

List of abbreviations

AA	Amino acid
Acc	Accuracy
BLAST	Basic local alignment search tool (N: nucleotide, P: protein, X: tr. nucleotide)
CDS	Coding sequence
CRISPR	Clustered regularly interspaced short palindromic repeats
DNA	Deoxyribonucleic acid
EPS	Exopolysaccharide
FISH	Fluorescence *in situ* hybridization
GB	Gradient boosting
GO	Gene ontology
Gp	Gene product
ICTV	International committee on taxonomy of viruses
ID	Identifier
LC-MS/MS	Liquid chromatography linked tandem mass spectrometry
LDA	Linear discriminant analysis
LDS	Lipopolysaccharide
OD	Optical density
ORF	Open reading frame
P	Precision
PCA	Principal component analysis
PCR	Polymerase chain reaction
R	Recall
RBP	Receptor binding protein
RF	Random forest
RNA	Ribonucleic acid
TAP	Tandem affinity purification

Contents

Introduction and outline

Introduction

Pathogenic bacteria continue to evolve and become resistant to new antibiotics. What if by 2050, as much as ten million lives will be lost annually due to multidrug resistant bacteria (ONeill *et al.*, 2016)? Antibiotics are typically small molecules that inhibit bacterial growth in some way. Because of their misuse and selective pressure on bacterial communities, over the years bacteria have developed numerous resistance mechanisms against these molecules. Examples are the metabolization or excretion of these molecules, or changing the target to render the antibiotic ineffective (Davies and Davies, 2010). As a result, classical discovery and development of new antibiotics is becoming increasingly difficult. Particularly, the lack of discovery of new classes of antibiotics along with the emergence and spread of resistance leads to a decreasing number of therapeutic options, and even no options in case of pan-drug resistant strains (Rossolini *et al.*, 2014). Another problem is the disruptive effect of broad-spectrum antibiotics on microbiota that have positive health effects (Ando *et al.*, 2015). Moreover, these beneficial bacteria can transfer resistance genes to pathogenic bacteria via horizontal gene transfer, leading to an even faster pace of resistance formation. As a result, the need for alternative, more specific approaches keeps growing year by year.

A promising alternative approach to this problem are the natural enemies of bacteria, bacteriophages. Every day, as much as half of all bacteria on earth are killed by bacteriophages (Rohwer *et al.*, 2009). These are bacterial viruses that can effectively infect and lyse bacterial cells by coding specific enzymes that degrade the bacterial cell wall. Phages themselves have been used to combat bacterial infections for many decades, a treatment which is called phage therapy. However, regulatory issues together with the success of small molecules as antibiotics hampered the popularity of this type of treatment (Wittebole *et al.*, 2014). Today, there is a renewed interest in phages, not only for their use in phage therapy, but also for specific proteins they encode (Doss *et al.*, 2017). Tail fiber and/or tail spike proteins are necessary to specifically recognize a suitable bacterial host. Additionally, proteins are needed to degrade the bacterial cell wall to escape the bacterial host. One class of enzymes that

degrades the bacterial cell wall are the endolysins (Schmelcher *et al.*, 2012). These proteins are all necessary to complete the lytic life cycle of these viruses. As bacteria have continued to evolve, bacteriophages have coevolved with them.

By studying bacteria-phage interactions and the proteins involved, important proteins can be identified and synthesized in the lab using recombinant DNA technology. With the help of protein engineering, these proteins can be used as novel antibiotics (so-called enzybiotics) or used for genomic engineering to create synthetic viruses with modified tail fibers and/or tail spikes. Another use of tail fibers is the development of customized pyocins or tailocins (Ghequire and De Mot, 2015). Tailocins are bacteriocins that kill bacteria by dissipating the membrane potential after binding to cell surface receptors. These protein complexes highly resemble bacteriophage tail structures, in which host specificity is determined by the specific tail fiber protein (Yao *et al.*, 2017). However, in contrast to phages, these protein complexes cannot self-replicate. Therefore, proteins such as enzybiotics, tail fibers and tailocins are more appealing to the current regulatory framework, compared to phage therapy.

The aim of this project is to get a better understanding of the factors that are important for host specificity of bacteriophages and to define general rules regarding host specificity of bacteriophages. To do so, a computational approach will be adopted. More specifically, phage-host interactions will be studied at two distinct levels: first at the proteome level and subsequently at the level of specific proteins that are known to be important for host specificity. The hypothesis is that by leveraging computational techniques, patterns can be uncovered that would otherwise not be identified by studying these interactions one by one. These can provide valuable insights in how phage-host specificity works at the protein level. At a later stage, the goal is to translate the obtained knowledge to the field of synthetic biology for the design of synthetic virusses with modified host specificity.

Outline of this dissertation

The first three chapters of this work comprise a series of bioinformatics analyses that are carried out to better understand phage proteomes and how specific proteins are important for specificity. Chapter one will start with a general overview of bacteria and phage characteristics, as well as their interactions and why it is important to study these interactions. In Chapter two, phages and their proteomes are discussed in depth, and a mathematical framework called optimal transport is adopted to study these proteomes. Chapter three discusses specific proteins that are known to be important for specificity in bacteria-phage interactions and applies optimal transport

again to identify these proteins in different phage proteomes. Chapter four of this work focuses on one type of phage protein that is known to be essential for host specificity: tail fiber proteins. Machine learning methods are developed to predict the bacterial hosts related to these proteins and try to discover patterns in these proteins that are related to a difference in host specificity. At the end of each chapter, the obtained results are discussed and useful insights are highlighted. Finally, a general conclusion is presented and some future perspectives are elaborated upon.

CHAPTER 1

How and why do bacteria and phages interact?

1.1 Bacteriophage life cycles

A bacteriophage or phage is a virus that infects a bacterial cell. Every phage consists at least of a viral genome composed of nucleic acid and a protein shell, the capsid, surrounding the genome (Weinbauer, 2004). This complete entity is referred to as the virion. Virions themselves have no metabolic or reproductional capabilities. In order for phages to reproduce, the virions have to come in contact with a suitable host. After finding a suitable bacterial host, the phage genome is injected into the prokaryotic cell and the reproduction cycle can continue. Generally, there are two methods for phages to reproduce: via the lytic lifecycle and/or the lysogenic lifecycle. The type of viral lifecycle determines the type of interaction between the phage and its host (Clokie *et al.*, 2011). Figure 1.1 gives an overview of these two cycles.

Both cycles start with a fully assembled virion particle. The virion contains proteins that specifically recognize the host which the phage is able to infect (Weinbauer, 2004). These are the proteins that determine host recognition and they are discussed in more detail in chapter three. After the virion binds to the correct receptor, the phage genome is translocated into the cytosol of the bacterial host cell. In a typical lytic cycle, the host's metabolism is quickly redirected to start synthesizing phage proteins and replicating phage genomes, while keeping the bacterial cell intact until new virion particles are ready to exit the host. Often, transcription of genes starts as soon as the first part of the genome has entered the host cell. The products of these genes assist in further entry of the genome (e.g. nuclease inhibitors) as well as in modifying the host's metabolism to better suit the needs of the phage (Cenens *et al.*, 2013). Phage protein production is highly regulated to be as efficient as possible (Youle, 2017). The bacterial metabolism will usually produce phage proteins and genomes in specific quantities as to maximize the number of assembled virions and minimize the duration in which the cycle is completed. Phage proteins usually

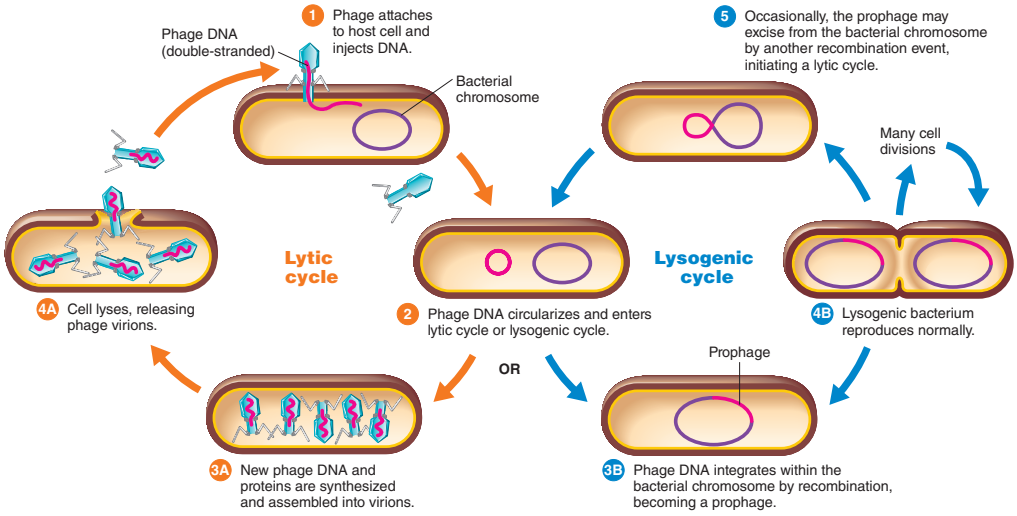


Figure 1.1: Overview of the lytic and lysogenic cycles of bacteriophage λ in *E. coli*.

In the lytic cycle, phages infect their host, inserting their genome in the host. The phage genome is transcribed and copied while virion production is started. Once enough virions are completed and packaged with a new copy of the phage genome, endolysins encoded by the phage genome will result in lysis of the bacterial cell, releasing the progeny virions. In the lysogenic cycle, the phage genome is inserted after which it is stably integrated with the bacterial chromosome, forming a prophage. This lysogenic bacterium can reproduce normally while replicating the phage genome along with it. Triggerred by several conditions, the prophage can excise out of the genome, to start a lytic cycle afterwards (Tortora *et al.*, 2013).

self-assemble, after which the phage genome is packaged into the newly formed capsid (Diaz-Munoz and Moskella, 2014). During protein production, transcription of the phage genome typically also results in the production of endolysins. At the end of the lytic cycle, these enzymes will degrade peptidoglycan in the cell wall of the host, causing the cell to lyse (Weinbauer, 2004). New virions exit the cell and the lytic cycle can repeat itself. Phages that reproduce only through means of the lytic cycle are referred to as virulent or lytic phages. The number of occurring lytic cycles is estimated at approximately 10^{25} every second, and they have been occurring for billions of years (Youle, 2017).

In the lysogenic cycle, the phage genome is injected in the host cell but is not (completely) transcribed. Instead, the genome either forms a (self-replicating) plasmid or integrates in the host genome. Integration is mediated by phage-encoded integrases. These enzymes bind at specific locations in the host chromosome and subsequently integrate the phage chromosome into the host genome by site-specific recombination. The integrated phage chromosome is now called a prophage and will replicate along with the host genome while the bacterial host replicates. This process is called propagation (Clokie *et al.*, 2011).

Because the pace of replication is more determined by the growth of the host, prophages replicate markedly slower than virulent phages. This process continues for an indefinite period, sometimes for several thousands of generations. The prophage genes needed for the lytic cycle are silenced, while other genes are expressed. The expression of these genes can sometimes benefit the bacterium in this interaction, for example by blocking infection by related phages (Zinder, 1958). At a later stage, however, if survival of the host cell is threatened, the prophage excises from the bacterial chromosome and resumes the lytic cycle. This switch is called prophage induction and can be triggered by DNA damage to the host or external conditions (Clokie *et al.*, 2011; Cenens *et al.*, 2013). Phages that are able to switch between a lytic and lysogenic cycle are named temperate phages.

Only a few phages never undergo a lytic cycle (Youle, 2017). A third, less common lifecycle, is one observed in some archaeal phages. It is a lifecycle in which the bacterial cell's metabolism is also redirected towards assembly of new virions, but with the difference that no lysis occurs at the end of the cycle. Instead, the host continues to grow (more slowly than normal), while new virions continuously extrude through the cell membrane. This cycle is sometimes referred to as the chronic lifecycle (Weinbauer, 2004). Here as well, proteins are needed for specific recognition of a suitable host. This type of lifecycle, however, limits the size of produced virions and thus chromosome length. Phages with fewer genes generally possess less capabilities for host manipulation and defense against their host or external conditions (Youle, 2017). Other authors, such as Cenens *et al.*, mention two more phage lifecycles, pseudolysogeny and carrier-state lifecycles. In both these lifecycles, the phage neither starts a lytic cycle nor integrates into the host chromosome. This can provide several benefits to the phage, such as protection of their genomic material from deteriorating conditions outside the host, preventing a lytic cycle when host recources are scarce or avoiding complete dependence on the host's DNA damage response for prophage induction (Cenens *et al.*, 2013). In pseudolysogeny, the phage usually later decides to continue with a lytic or lysogenic cycle. It is therefore not clear whether pseudolysogeny represents an actual cycle or only a decision point in the life cycle of temperate phages (Diaz-Munoz and Moskella, 2014). This might indicate that this classification of viral lifecycles is an oversimplification of the actual diversity of viral lifecycles (Weinbauer, 2004).

1.2 The (dis)advantages of bacteria-phage interactions

As all viruses explicitly need a host cell to replicate, the main benefit of bacteria-phage interactions for the phage itself is the possibility of the propagation of its lineage. Through a lytic cycle, virulent phages can produce large numbers of progeny in a short period of time by benefiting from the host cell's metabolism and infrastructure. Temperate phages replicate slower in a lysogenic cycle and also need more genes than virulent phages. Besides performing a lytic cycle, temperate phages also need to decide what life cycle to follow, they have to be able to insert and excise from the host chromosome, silence specific genes and monitor the host cell to know when the lytic cycle can or has to be resumed. However, in environments with nutrient limitation, growth of the bacterial host will be limited or non-existent. As nutrients are also needed for virion production, delaying the lytic cycle in this scenario allows the temperate phage to maintain replication, while waiting for better conditions. Choosing lysogeny also protects the phage chromosome from UV radiation and ensures survival when host abundance is low (Jiang and Paul, 1996).

Bacterial hosts can also benefit from bacteria-phage interactions in several ways. Prophages protect the bacterium from infection by related phages by expressing genes that block superinfection (i.e. additional infection by other phages), for example by modifying cell surface receptors of the bacterial host. A lysogenic cycle is also blocked for these related phages as the integration site in the bacterial chromosome is not available anymore (Susskind *et al.*, 1974; Youle, 2017). A second benefit is the bacterial diversity that arises from the infection with prophages. Prophages frequently express metabolic genes, to help the host grow and replicate in adverse conditions, which can be a competitive advantage (Cenens *et al.*, 2013). These and other genes can also be acquired by the bacterium itself when the genes needed for excision of the phage chromosome lose their function due to mutations from replication errors. In that case, the prophage is no longer able to resume a lytic cycle, and the defective prophage remains in the bacterial chromosome, while most of the phage genes remain functional. These genes can benefit the bacterium, acting as specialization genes that are not present in other strains of the species (Youle, 2017). Thirdly, genes from the prophage chromosome can encode toxins and virulence factors that are essential to the pathogenicity of several bacterial strains. Nearly all pathogenic bacteria carry at least one prophage, e.g. *Vibrio cholerae* and *E. coli* O157 (Ross *et al.*, 2016). The encoded toxins can also serve as self-defense of the host against the inside of food vacuoles of protists. Finally, phages contribute to bacterial

evolution by enabling horizontal gene transfer through a process called transduction (Diaz-Munoz and Moskella, 2014). This is the transfer of cellular genes from one bacterial host to another by phages. For successful transduction, DNA from the first host has to be packaged inside a virion, the packaged DNA has to be delivered to a new host cell and the DNA should finally be integrated in the new host's chromosome. This process can be done in two ways, which are visualized in Figure 1.2. The first way results after an imperfect excision of the prophage chromosome from the host chromosome. Here, the prophage is cut in a way that includes some adjacent host DNA. If the resulting virion infects a new cell and is inserted into the new bacterial chromosome, the bacterial DNA from the first host is inserted as well. This type of transduction only transfers bacterial genes that are adjacent to insertion and excision sites of the phage chromosome. Therefore, it is called specialized transduction. A second type of transduction can transfer any region of the bacterial chromosome. At the end of a lytic cycle, phage packaging proteins can mistake bacterial DNA for the phage chromosome and accidentally package this DNA inside a virion. Because there is no restriction on what genes can be packaged this way, this type of transduction is called generalized transduction (Trevors, 1999).

1.3 Coevolution of bacteria and their phages

Not every delivery of a phage chromosome results in a successful infection of the host. As bacteria continue to evolve over time, they have developed numerous defense mechanisms to stop phages from infecting them (Chaturongakul and Ounjai, 2014). Bacteria can reduce adsorption by modifying receptors, avoid takeover of the cell's metabolism post-infection (e.g. via CRISPR-Cas adaptive immunity) or commit suicide in which both the cell and phage die (Fineran *et al.*, 2009; Diaz-Munoz and Moskella, 2014). Reducing phage adsorption is the most studied process, as this can be observed under laboratory conditions. Phages rely on bacterial receptors for recognition of their hosts, and these receptors can be modified to avoid phage infection. In coculture experiments of bacteria and their phages, bacteria with altered phage receptors can rapidly take over the population. For example, the segment which is recognized by the phage can be concealed without compromising the function of the receptor. The recognized segment can also be modified by mutation. Mutation of just one amino acid in a critical location can be sufficient to block phage recognition. However, this alteration can reduce a bacterium's fitness in the environment as these receptors are often responsible for nutrient uptake (Weitz *et al.*, 2005). Even a minor reduction in fitness can significantly reduce the bacterium's share in the population over the long term. However, this mutation can be maintained as long as the benefits

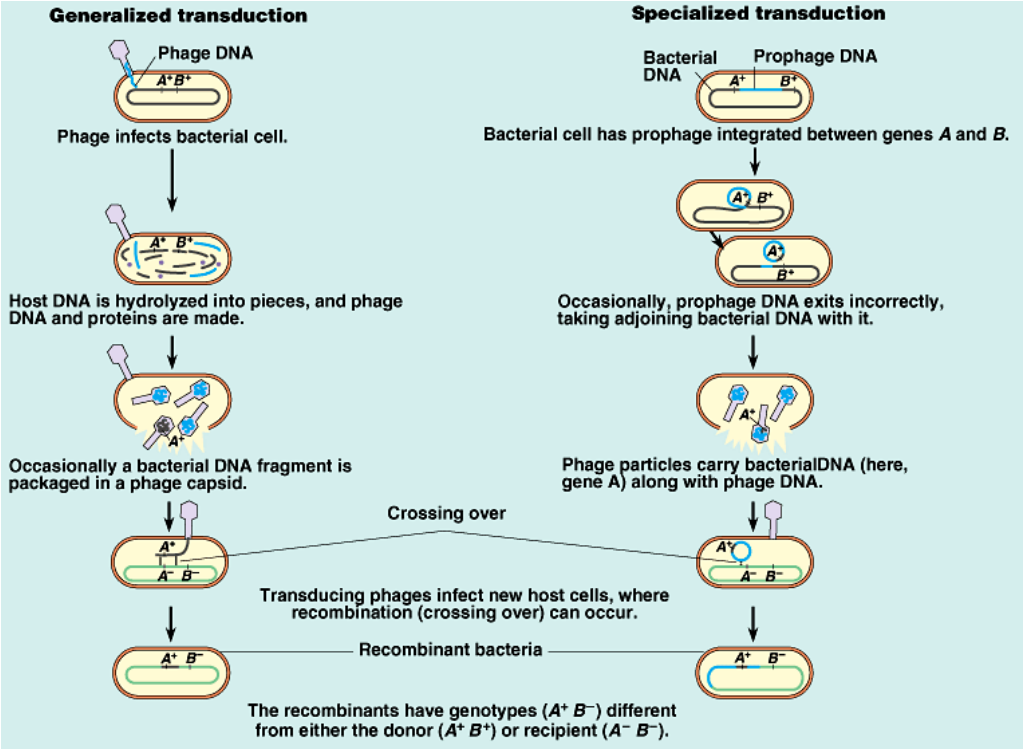


Figure 1.2: Transduction mechanisms in phage bacteria interactions that result in horizontal gene transfer.

Generalized transduction (left) is the process in which, after phage infection, bacterial DNA is packaged in a virion. If this virion infects a new host, crossing over of this bacterial DNA can occur to the genome of a new (related) bacterial host. In specialized transduction (right), bacterial DNA is packaged in a new virion as a consequence of imprecise excision of a prophage. This virion can infect a new bacterial host as well, possibly transferring genes located next to the prophage integration sites to a new (related) bacterial cell (Simon *et al.*, 2010).

from reduced phage predation outweigh the disadvantages of reduced nutrient uptake (Youle, 2017). Additionally, phages can evolve to come up with new strategies to infect their host. The genes encoding the proteins needed for recognition are known to be the fastest evolving genes in the phage genome. Lenski does note that evolution of this highly specific adsorption process might come with structural constraints that are more serious than nutrient uptake restrictions of bacteria. Because of this, there exists an asymmetry in the evolutionary potential of bacteria and their phages (Lenski, 1984). Nevertheless, coevolution as a whole has been ongoing for billions of years and this is one of the reasons that studying interactions between bacteria and their phage is interesting.

While coevolution of virulent phages and their host is antagonistic, lysogeny can result in mutualistic coevolution (Lenski, 1984). Indeed, phages and bacteria also adopt mechanisms from each other. After lysogeny, the inserted phage chromosome is subject to the same rate of mutation as the bacterial chromosome. When such a mutation results in the phage chromosome being unable to excise, the prophage remains

present in the bacterial chromosome, while most of the genes remain intact. Bacteria can use some of these genes, and gradually eliminate other genes that are not useful. For example, major capsid proteins of defective prophages can evolve towards proteins used by bacteria to build compartments that house bacterial enzymes (Youle, 2017). Compartmentalization in a metabolic pathway involving multiple steps can increase the efficiency of production in this pathway. Some bacteria also use these capsid proteins to produce virions containing bacterial DNA as a means of horizontal gene transfer. As these modified virions still retain their original host specificity, they are able to transfer genes between closely related host cells. However, because these modified virions can only be released through cell lysis, this process is only deployed under stress conditions. A third example is the adoption of phage tail structures by bacteria. Several hosts retain the phage tail gene cluster and modify these tail structures to serve as so called pyocins or tailocins (Chaturongakul and Ounjai, 2014). These proteins can either puncture the cell membrane of target cells or inject toxic substances into them. However, assembled tailocins are only released through cell lysis, therefore tailocins are only produced when the cell is faced with irreparable DNA damage (Youle, 2017). In this regard, tailocins provide the bacterial species as a whole an increased resilience, particularly under stressful conditions (Diaz-Munoz and Moskella, 2014).

Whether coevolution is a never-ending process can be debated. For example, phages can still exist in a population of resistant bacterial hosts if sensitive hosts are present and have a competitive advantage over resistant hosts (Lenski, 1984). On the other hand, when bacteria mutate or conceal the receptor needed for phage recognition without a reduction in fitness, these bacteria develop complete resistance, while avoiding competitive disadvantages. In such a scenario, phage evolution may not be able to catch up with resistant hosts and coevolution stops for this bacteria-phage combination (Weitz, 2005). Still, there is substantial evidence that coevolution of bacteria and their phages contribute to bacterial diversity as well as effect bacterial virulence and bacterial evolvability (Diaz-Munoz and Moskella, 2014). It is also clear that phages play a key role in the never-ending evolution of bacteria, and vice versa (Chaturongakul and Ounjai, 2014).

1.4 Experimental approaches to determine phage-host specificity

Over billions of years, coevolution has also determined phage host range (phage-host specificity). Many phages are highly specific to only one or a few bacterial species,

even particular strains. On the other hand, some phages do exhibit a broad host range, even spanning different bacterial genera. This can be explained by the fact that phages can also experience a reduction in fitness when evolving. Phages can evolve to infect a broader range of hosts, but if this expansion in host range implies a reduction in reproductional capabilities, these phages might not be able to compete with other phages for a finite number of hosts. Host range probably also depends on the environment in which bacteria and phages interact, which might explain why some phages do continue to exhibit broader host range. For example, when host abundance is low relative to the number of phages infecting these host, it is advantageous to exhibit a broader host range (Koskella and Meaden, 2013).

Several laboratory methods exist to determine phage host range. Two commonly used techniques are plaque assays and spot assays. In plaque assays, one or a few phages (e.g. derived from an environmental sample and diluted) are inoculated with a growing bacterial culture. After repeated rounds of infection and bacterial lysis, plaque formation (clearing) is observed for cells where phages are able to produce progeny. In spot assays, a small volume of phage is placed on growing bacteria, after which lysis of bacterial cells can be observed as a confluent clearing zone (Middelboe *et al.*, 2010). However, when a large number of phages absorb to the bacterial cell, lysis can occur without infection, thus producing a false positive result (Edwards *et al.*, 2016). To avoid this, dilution series have to be made. Only true positive results will still produce individual plaques when diluted. In addition, both methods can produce false negative results if temperate phages choose lysogeny over the lytic cycle. A third option is a liquid assay, in which bacterial growth in liquid culture is measured by optical density (OD). After addition of phages, bacterial growth will decrease relative to a control when phage infection is successful. However, cell lysis forms debris that can impact OD measurements. Phages can also be fluorescently labeled (viral tagging). After adsorption of labeled phages to bacterial cells, the tagged bacterial cells can be sorted using flow cytometry and identified using sequencing techniques (Mossier-Boss *et al.*, 2003). This method only measures phage adsorption, which does not necessarily indicates a successful infection. Several other experimental approaches exist, including PCR, fluorescence *in situ* hybridization (FISH) and sequencing, as discussed by Edwards *et al.* (Edwards *et al.*, 2016).

In general, different methods can give different results (Diaz-Munoz and Moskella, 2014). This is due to the fact that different methods depend on different steps of the phage infection process. The presence of prophages or plasmids and bacterial resistance mechanisms can all influence phage infection. Phage host range may well neither be a stable nor binary characteristic (Diaz-Munoz and Moskella, 2014; Ross *et al.*, 2016). In addition, results from laboratory techniques may hardly be extend-

able to phage host ranges in their native environment (Chaturongakul and Ounjai, 2014). Generally speaking, phage-host specificity is not a very well understood process. Therefore, this work will attempt to contribute to this area by studying phage proteomes in a bioinformatics and machine learning context. The following section briefly describes the work that has already been done in this area.

1.5 Phage-host specificity in a computational context

As the amounts of genomic and proteomic sequence data continue to increase dramatically, there is a need for computational tools to analyze this data. This presents an opportunity to study biological organisms in a novel way by leveraging computer tools. For example, sequencing an entire viral community (virome) may allow for the identification of viruses in the community without culturing, hence avoiding culturing-associated biases (Mokili *et al.*, 2012). On the other hand, by avoiding culturing, a direct link with its host is lost. As discussed in Section 1.4, determination of phage host range in a laboratory environment presents several challenges and even ambiguities. But because of the fact that coevolution between bacteria and their phages shapes their genomes and proteomes, *in silico* analyses of these data can serve as worthy alternatives to study and determine phage-host specificity (Edwards *et al.*, 2016).

DNA sequence information can provide information regarding bacteria-phage interactions in several ways. These include abundance profiles, genetic homology, CRISPR spacers, exact matches between phage and bacterial genomes and oligonucleotide profiles. Firstly, abundance profiles of phage and bacterial sequence across metagenomic datasets can provide information regarding phage host specificity. As phages explicitly depend on their host for reproduction, phages will only be present in environments where corresponding bacterial hosts are also present. These abundance profiles change over time. Changes in phage abundance coinciding with changes in bacterial abundance can indicate an interaction between both (Stern *et al.*, 2012). One disadvantage of this method is that population dynamics can blur the correlation between the abundance profiles of phages and their hosts. For example, antibiotic treatments will decrease host abundance, while phages are still present. Genetic homology provides an alternative approach to predict bacteria-phage interaction (Modi *et al.*, 2013). Secondly, both lytic and temperate phages can incorporate bacterial DNA into their chromosome due to errors in DNA packaging into the virion or excision of the prophage out of the bacterial chromosome. If this bacterial DNA provides

a competitive advantage to the phage, natural selection will retain this DNA in the phage chromosome. These homologous genes in phages and bacteria can be used to predict bacteria-phage interaction, through sequence similarity searches (Edwards *et al.*, 2016). Thirdly, CRISPR spacers can also be used for this purpose (Stern *et al.*, 2012). Several, but not all bacteria, retain a short 25-75 base pair long nucleotide sequence (called spacer) of the phage in their genome. The CRISPR-Cas system provides adaptive immunity to the bacterium (Horvath and Barrangou, 2010). Through sequence alignment, these spacers can be linked to the phages that infect the bacterial cell of interest. Edwards *et al.* (2016) note that this approach strongly depends on the number of mismatches allowed between spacer and phage genome. Finding an appropriate CRISPR match is rare, but of strong significance if identified. Another point of consideration is that CRISPR spacers are replaced over time, thus making these sequences more suitable for prediction of recent phage-host interactions (Horvath and Barrangou, 2010). Yet another method of predicting bacteria-phage interaction is by looking at exact matches between phage and host genome. This is especially useful for temperate phages that integrate in the host genome. Prophage sequences as a whole can be searched for directly in the bacterial genome. Prophage integration sites also contain exact sequence matches. These integration sites consist of flanking DNA (P and P' in the phage genome, B and B' in the bacterial genome), with in between both ends a common core that is identical between phage and host (Hoess and Landy, 1978). This common core does vary in length, and shorter sequence matches can hardly be distinguished from random matches. Prophinder, a computational tool to predict prophages in bacterial genomes, takes this one step further. The algorithm identifies phage-like coding sequences (CDSs) in the bacterial genome by gapped BLASTP search. Subsequently, prophages are predicted based on regions in the bacterial genome that are enriched in phage-like genes (Lima-Mendez *et al.*, 2008). One final approach to predict bacteria-phage interaction is through the use of oligonucleotide profiles. Phages will adapt their nucleotide composition to cope with intracellular nucleotide pools and tRNA availability as well as to avoid recognition by host restriction enzymes (Pride *et al.*, 2006). Comparing oligonucleotide usage profiles of bacteria and phages by calculating the Euclidean distance between these profiles can be used to predict phage-host relationships (Roux *et al.*, 2015). Edwards *et al.* (2016) assessed the predictive power of the above-mentioned methods by analyzing 820 phages with annotated hosts. In their study, homology-based approaches resulted in the highest number of correct predictions to identify phage hosts. Using exact matches tended to be the most informative, predicting the correct host species in approximately 40% of the cases. Genetic homology using nucleotide BLAST (BLASTN) performed only slightly worse in predicting the correct host species, which was correct in 38.5% of the cases. On the other hand, homology-independent

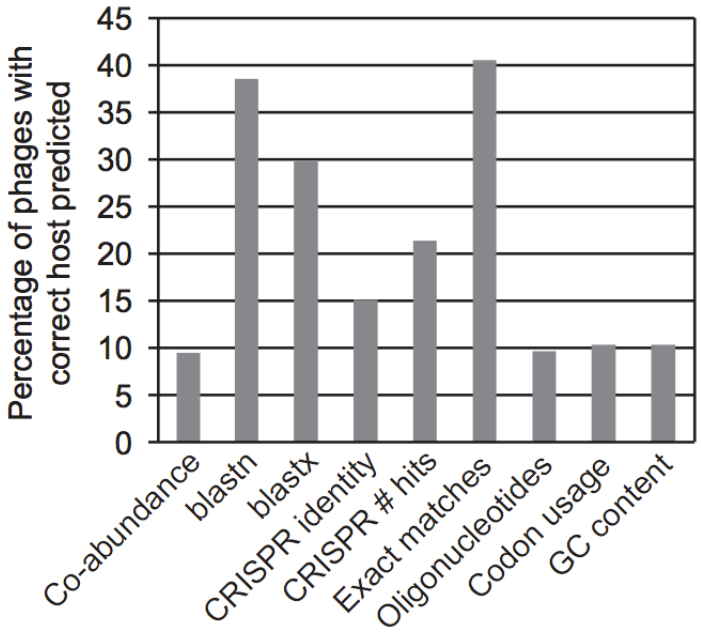


Figure 1.3: Results of the study by Edwards *et al.* which compared different computational methods to predict bacteriophage-host relationships.

The figure shows the percentage of phages for which the host was correctly predicted (top scoring hosts) across different used computational approaches.

approaches also seem to be appropriate for prediction of phage hosts. Additionally, performances of about 40% still have a lot of room for improvement. An overview of these results is given in Figure 1.3 (Edwards *et al.*, 2016).

In another approach, Ahmed *et al.* (2009) used oligostickiness as a measure to predict hosts of 25 phage species. Oligostickiness is a measure based on binding stability of an oligonucleotide to a genome sequence. Indirectly, this is a measure of relaxed sequence similarity. Oligostickiness calculates the free energies of all possible hybridization structures between an oligonucleotide at positions along a genome sequence. In this regard, diverged (relaxed) sequences can still be analyzed because binding energy is more robust against mutations. Oligostickiness was used to calculate a similarity score between genomes. This similarity score appeared to be significantly higher between phage genomes and genomes of their known hosts as opposed to unrelated bacterial genomes. Similarity scores even allowed to discriminate between virulent phages and temperate phages (Ahmed *et al.*, 2009).

As described in Section 1.1, phages use specific receptor binding proteins that bind with bacterial receptors located on the cell surface. These receptor binding proteins provide a link between phage and host. However, this is not a one-to-one relationship. One bacterium may be able to be infected by multiple phages using multiple bacterial receptors for phage adhesion (Chaturongakul and Ounjai, 2014). The question

remains whether these receptor binding proteins possess unique characteristics that enable to distinguish between different hosts. This is the main hypothesis of Chapters three and four, which will elaborate on this question using different approaches. However, Chapter two will first introduce a mathematical framework called optimal transport that can be used to study phage proteomes and proteins, after which it will be used to study phage-host specificity at the proteome level. Chapter three will then discuss receptor binding proteins in more detail and apply optimal transport to the proteomes of T7-like phages to identify unique elements in these highly similar proteomes. Chapter four will specifically focus on certain receptor binding proteins and apply machine learning algorithms to identify unique characteristics of these proteins, which will be used to predict bacteria-phage interaction.

CHAPTER 2

Phages and their proteomes

2.1 Characteristics of phage proteomes

Bacteriophages generally exhibit a highly specific host range. Therefore, they are a viable option to control bacterial pathogens. However, in order to use them effectively, knowledge of phage biology and bacteria-phage interaction is needed. As discussed in the previous section, studying phage genomes can considerably aid in understanding bacteria-phage interactions. Studying phage proteomes can serve this purpose as well.

Phages, and viruses in general, are spectacularly diverse in the nature and organization of their genetic material, gene sequences and encoded proteins (Simmonds *et al.*, 2017). In general, proteins are needed for DNA replication, integration (for lysogenic phages), packaging, structural aspects (head and tail) and performing lysis. This wide spectrum of proteins and protein functionalities results in an interplay with proteins in host cells (ShengTao *et al.*, 2011). Interestingly, the genes needed for these different functionalities may be present in distinct modules in the genome, as is the case for phages of *Staphylococcus aureus* (Kwan *et al.*, 2005). It is the variation in these proteins that allows bacteriophages to diversely interact with bacterial hosts.

Traditionally, the proteome composition of a specific virus is determined by cultivating the virus, extracting proteins, separating those proteins by electrophoresis and finally identifying the proteins through the use of immunoblotting or Edman degradation (ShengTao *et al.*, 2011). In more recent years, new methods have been developed and subsequently used in viral studies. For example, two mass spectrometry approaches have been widely used to study viral proteins: matrix-assisted laser desorption ionization (MALDI) time of flight (TOF) mass spectrometry and liquid chromatography-linked tandem mass spectrometry (LC-MS/MS). Both techniques are schematically represented in Figure 2.1. Other approaches are also used, and most methods are complementary to one another (Maxwell and Frappier, 2007).

Besides mass spectrometry, several other proteomic techniques can be used to study virus-host interactions, in order to identify the proteins that allow viruses to infect

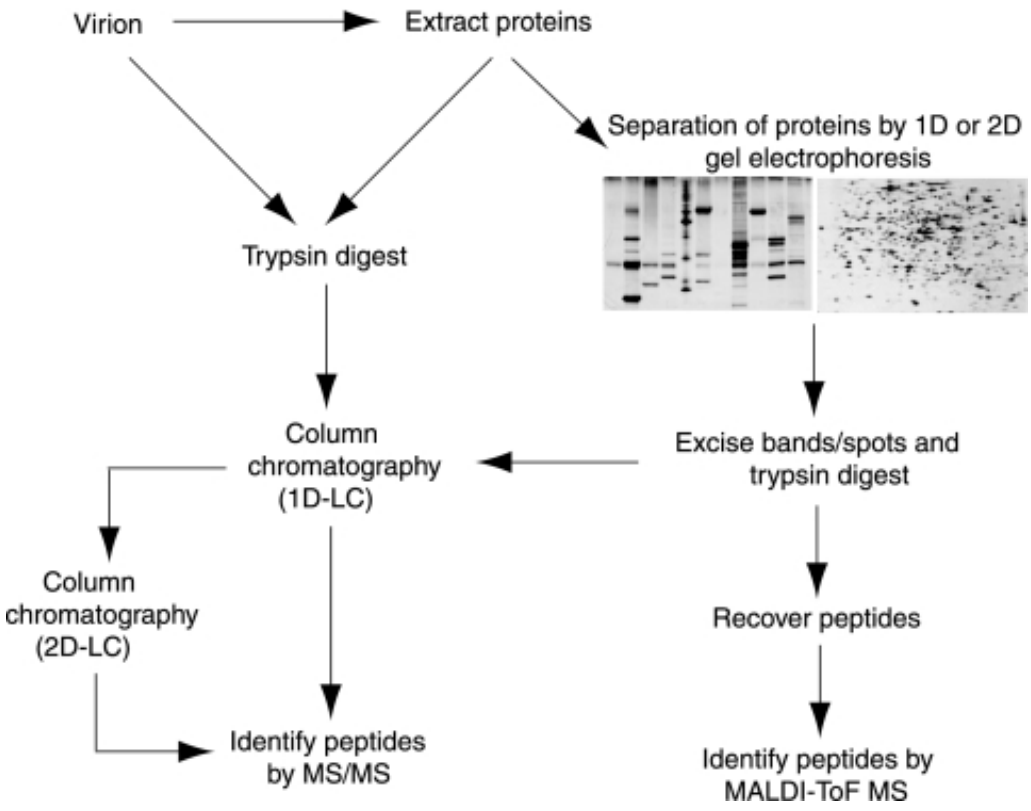


Figure 2.1: Schematic representation of workflows for popular mass spectrometry techniques in viral protein studies.

The left part of the figure displays the use of liquid chromatography (LC) followed by two-step mass spectrometry (MS/MS) as a way to characterize viral proteins. The right part of the figure displays an alternative approach to study viral proteins, that uses gel electrophoresis for protein separation, followed by MALDI-ToF MS to identify peptides of interest (Maxwell and Frappier, 2007).

a host and replicate within it. Most often, yeast two-hybrid screenings are used to screen for protein interactions, because of their simplicity and easy applicability to relatively large cDNA libraries. However, results depend on the quality of the library, as well as the expression levels of individual proteins and their ability to be transported to yeast nuclei. Another technique that is gaining popularity is tandem affinity purification (TAP) tagging. However, studying protein interactions at a genome-wide scale is difficult and time consuming. In general, just a fraction of viral proteins of a small number of viruses have been studied to date. Thus, a wealth of information on viruses and their interactions with hosts is not yet uncovered. This information would likely lead to a broader biological understanding as well as possibly contribute to technological innovation (Maxwell and Frappier, 2007). Specific computer analyses can contribute to this broader understanding, by using publicly available online data. In that way, computer analyses could also serve as starting point for functional studies.

One of the goals of this work is to study a new approach of comparing phage proteomes and discovering similarities and differences in these proteomes. As a first proof-of-concept of this new approach, it will be used for phage classification by constructing a bacteriophage species tree. For this reason, Section 2.2 will briefly discuss the history of phage taxonomy. Afterwards, Sections 2.3 and 2.4 will explain the method that will be used in this work.

2.2 Phage classification

Virus taxonomy started being addressed in 1965, when an international committee was formed that later grew into what is now called the International Committee on Taxonomy of Viruses (ICTV). Because no single homologous gene is shared by all bacteriophages, taxonomy based on a single gene (as is done for prokaryote classification) is impossible. Instead, classical phage classification uses a classification scheme written by David Bradley in 1967. Here, different virus families are defined based on the nature of phage nucleic acid (dsDNA, ssDNA, ssRNA, dsRNA) as well as overall virion morphology (tailed, polyhedral, filamentous, pleomorphic). The use of electron microscopy, together with the discovery of different forms of nucleic acids are central to this classification. Other properties used in classification include replication characteristics in cell culture, serology, host range and more. As of today, the ICTV still uses this classification scheme as the basis of virus taxonomy. Thus, classifying a new virus still requires investigating these specific characteristics (Adriaenssens *et al.*, 2015; Simmonds *et al.*, 2017).

In recent years, the focus in bacteriophage research has gradually shifted towards genomics and proteomics (Lavigne *et al.*, 2008). Today, there are many more viruses known from sequence data alone than viruses that have been characterized experimentally. Viruses that are of less importance to the economy or to society are not likely to ever be fully experimentally characterized (Simmonds *et al.*, 2017). As a result, many of the completely sequenced phages in GenBank have not been added to the official ICTV classification (Rohwer and Edwards, 2002). To close the gap between unclassified and classified phages, several new classification methods have been proposed.

In 2002, Rohwer and Edwards studied 105 completely sequenced phage genomes. After concluding that no single homologous gene is shared by every phage genome, they developed a new taxonomic system based on the predicted phage proteome. All predicted phage protein sequences were compared in a pairwise manner using

BLASTP. Subsequently all proteins (per comparison) with an E-value[1] (expectation value) < 0.1 were aligned using CLUSTALW. The cutoff was arbitrarily set to 0.1. For each alignment protein, distance scores were calculated using the PROTDIST tool (part of PHYLIP software package). Finally, a proteomic distance score was calculated from the sums of protein distance scores correcting for penalties in alignment, average length of proteins and the number of missing proteins. The resulting distance matrix can then be used to construct a phage tree. Rohwer and Edwards dubbed this approach the Phage Proteomic Tree. The authors argue that the more characteristics two organisms share, the more closely related they are. In this regard, protein sequences are an obvious choice to use in classification, when it is expected that related phages will have similar proteins. This hypothesis will also be used in this work and will be elaborated upon further in this chapter (Rohwer and Edwards, 2002).

In 2008, Lavigne *et al.* presented their CoreExtractor.vbs and CoreGenes software tools. Both tools are complementary to each other and provide a way of comparing (total) genome similarity (at the protein level) by using BLASTX and iterative BLASTP. The CoreExtractor tool analyses BLASTX output files of each viral gene of all phages to be compared. All BLAST output files were searched for each phage name in the analysis, and a matrix is returned that contains correlations based on the number of gene products that are similar between phages. The CoreGenes tool is based on the GeneOrder algorithm, which uses progressive iterative BLASTP to detect a common set of proteins for up to five genomes. Finally, the outputs of CoreExtractor and CoreGenes were converted to their relative correlation and reciprocally compared to form an appropriate threshold value for minimum similarity. Their approach has been evaluated on 55 phage genomes from the *Podoviridae* family and, with the exception of five proposed new genera, aligns with the classification schemes of the ICTV (Lavigne *et al.*, 2008).

In 2015, Adriaenssens *et al.* combined both genomic and proteomic comparisons to classify previously unclassified members of the *Siphoviridae* family. In their approach, trees based on the entire genome were constructed using ClustalW 2.0. In parallel, proteomic trees were constructed using the Phage Proteomic Tree approach. Both trees were used in a qualitative manner to identify clusters of phages. In each cluster, a type phage was chosen. Afterwards, phages in the same cluster showing more than 95% DNA identity to the type phage were grouped in the same species. Finally, a CoreGenes analysis was used to group phages into the same genus when over 40% of proteins were shared between the phages. Several other research groups have proposed other classification schemes. The two main reasons why research groups keep developing new classification systems is the fact that the official classification

[1]The E-value is the number of alignments with a certain score S that are expected to occur in a database search by coincidence.

system is outdated and that there is no consensus on which newly developed system is most appropriate (Addriaenssens *et al.*, 2015).

As recent as 2017, a consensus statement endorsed by the ICTV identified the potential of metagenomic data in characterizing the global virome. The purpose of the consensus statement was to address the need for an easily applicable classification system for newly sequenced viruses. A classification scheme solely based on metagenomics would be a substantial departure from the current basis of virus taxonomy (as discussed earlier in this section). However, within a robust framework and with appropriate quality control, viruses that are only known from metagenomic data can now be incorporated in the official ICTV virus taxonomy. Indeed, as characteristics of viruses are encoded in the genome, properly analyzed sequence data can provide the necessary information for viral classification using ICTV's criteria (Simmonds *et al.*, 2017).

2.3 Alignment-free sequence analysis

Most commonly, either pairwise or multiple sequence alignment is used to quantify similarity between sequences. For large-scale comparisons, these methods become unfeasible due to large computational time and high memory consumption (Das *et al.*, 2017). Use of heuristics can circumvent these problems, for which BLAST and FASTA are the two most known approaches (Vinga and Almeida, 2003). An alternative is the use of alignment-free methods to quantify sequence similarity. Several of these methods use the frequencies of words (also defined as a k-tuple or k-mer) occurring in a sequence to capture sequence similarity by statistical testing. In a sequence X of length n, a k-mer is defined as a subpart of the sequence with length $k < n$. The counting of different k-mers in a sequence is usually performed by using a sliding window of length k that runs over the sequence from position 1 to n-k+1 (Vinga and Almeida, 2003). This results in a collection of m counts C_k^X for the sequence X, where $m = 1, \ldots, n-k+1$. This is given by Eq. (2.1).

$$C_k^X = (C_{k,1}^X, \ldots, C_{k,m}^X). \tag{2.1}$$

From this list of counts, k-mer frequencies can be calculated as the relative abundance of every particular k-mer by dividing the count for that particular k-mer by the total number of counts. Equation (2.2) is used for this calculation (Vinga and Almeida, 2003). Together, these frequencies can be seen as a discrete probability distribution, as the frequencies add up to one.

$$F_k^X = \frac{C_k^X}{\sum_{j=1}^{m} C_{k,j}^X}. \tag{2.2}$$

In this work, *k*-mers will also be used to quantify sequence similarity by representing proteins and proteomes as probability distributions of *k*-mers. These probability distributions are then compared using a mathematical framework called optimal transport. It is hypothesized that optimal transport will show to be a convenient alternative approach in alignment-free methods, in particular to study phage proteomes. This will be elaborated upon further.

The distances between these probability distributions can subsequently be used to construct a phage tree. One disadvantage of the use of an alignment-free method based on proteomes is that extra information such as genome organization (i.e. the ordering of genes in the phage chromosome) is not incorporated when comparing phages.

2.4 Optimal transport as a way to compare phages

2.4.1 Introduction to optimal transport

Optimal transport is a mathematical framework that can be used to measure distances between mathematical functions, probability distributions or more general objects. It can also be used for interpolating probability density functions.

The optimal transport problem was initially formalized and studied by French mathematician Gaspard Monge (Monge, 1781). Monge wanted to minimize the total amount of work needed to transform a terrain with a particular landscape into another landscape. Mathematically, this problem translates into finding a function (if any) that transforms the current landscape into the target landscape, while minimizing the product of the amount of transported earth with the distance over which this earth is transported (Lévy and Schwindt, 2017). The generality of this theory makes broad applications of it possible, ranging from optimally dividing resources between factories to computer vision and, as studied in this work, comparing phages among each other.

Optimal transport can be formally defined as follows. Given two vectors **r** and **c** of dimensions *n* and *m* respectively, let $U(\mathbf{r},\mathbf{c})$ be a polyhedral set of $n \times m$ matrices P where the rows sum to **r** and the columns sum to **c**. This is expressed by Eq. (2.3):

$$U(\mathbf{r},\mathbf{c}) = \{P \in \mathbb{R}_{>0}^{n\times m} \mid P\mathbf{1}_m = \mathbf{r}, P^T\mathbf{1}_n = \mathbf{c}\}\,. \tag{2.3}$$

Intuitively, the set $U(\mathbf{r},\mathbf{c})$ represents all the possible and valid ways of transporting earth from $\mathbf{r}$ to $\mathbf{c}$. An element P of $U(\mathbf{r},\mathbf{c})$ is therefore often referred to as a transportation matrix. The optimal transport problem can then be defined by Eq. (2.4), where the sum of the product of P and M represents the cost of mapping $\mathbf{r}$ to $\mathbf{c}$ using a $n \times m$ cost matrix M:

$$d_M(\mathbf{r},\mathbf{c}) = \min_{P\in U(\mathbf{r},\mathbf{c})} \sum_{i,j} P_{ij}M_{ij}\,. \tag{2.4}$$

The goal is to minimize this cost, which then results in a metric $d_M(\mathbf{r},\mathbf{c})$ called the Wasserstein distance. It can be interpreted as a distance, whenever M also represents a distance matrix (Cuturi, 2013).

2.4.2 Optimal transport with entropic regularization

The minimization problem above can be solved relatively easy using linear programming. However, the time complexity scales at least in $O(d^3 log(d))$ when computing the distance between a pair of histograms of dimension d. The time complexity can be improved substantially by adding a regularization term to the optimal transport problem. This way, the linear problem is transformed into a strictly convex problem, which can be solved much faster using the Sinkhorn-Knopp matrix scaling algorithm (Cuturi, 2013). The modified version of the optimal transport is given by Eqs. (2.5) and (2.6).

$$d_M^\lambda(\mathbf{r},\mathbf{c}) = \min_{P\in U(\mathbf{r},\mathbf{c})} \sum_{i,j} P_{ij}M_{ij} - \frac{1}{\lambda}h(P)\,. \tag{2.5}$$

with

$$h(P) = -\sum_{i,j} P_{ij}\log(P_{ij})\,. \tag{2.6}$$

Here, $d_M^\lambda(\mathbf{r},\mathbf{c})$ is called the Sinkhorn distance and $h(P)$ is the entropic regularization term, also termed the information entropy of P. A lower value of λ will result in more regularization (more entropy), and vice versa. From a practical point of view, regularization makes sense. A higher entropy will result in a more even, smooth joint distribution. Finding such transportation plans is often more informative than finding extreme plans that are not as likely to appear in real-world situations. However,

when the objective is to calculate distances between probability distributions, a small value of λ is often unwanted. Smoothing the distribution causes some approximation errors (i.e. overestimating the actual distance) (Cuturi, 2013). Therefore, it might be advantageous to use a higher value of λ. Section 2.6 will elaborate on the choice of the value of λ.

2.4.3 Applying optimal transport to phage proteomes

In this work, optimal transport will be applied to phage proteomes at two distinct levels. First, the optimal transport framework will be used to calculate similarity between phage proteomes in a pairwise manner. Phage proteomes are split up into k-mers using a sliding window over the protein sequence. These k-mers thus represent overlapping parts of the full proteome sequence, each having a length of k. For every proteome, k-mers are counted and these counts are subsequently normalized to represent a probability distribution. These probability distributions can be compared with optimal transport. Using Eqs. (2.5) and (2.6), the Sinkhorn distance is calculated for every pair of phages. These distances are subsequently used to construct a phage tree. Finally, the constructed phage tree will be compared to another, more established method.

2.5 Data aqcuisition and data quality assessment

Proteome data was gathered from the UniProt Knowledge Database (UniProtKB) (The UniProt Consortium, 2017). UniProtKB is the open-source alternative to the SwissProt and trEMBL databases and is known for its high-quality protein data. In addition, Uniprot provides a way to search for complete proteomes. The database provides a set of 'reference proteomes', which have been selected among all proteomes in a manual and algorithmic way, to have a decent quality based on several criteria. Searching for the word 'phages' in the UniProt proteome database, and restricting to 'reference proteomes' in the super kingdom of viruses, resulted in a set of 985 reference proteomes, which were downloaded from UniProt (November, 2017).

2.6 Constructing a phage distance tree

To first motivate the use of optimal transport as a way of comparing phage proteomes, a relationship between Sinkhorn distances and pairwise local alignment scores was

constructed using combinations of both low and high values for λ and k. More specifically, the value of λ was either 0.1 (low) or 30 (high), while the value of k was either 3 (low) or 15 (high). This relationship was computed to show whether Sinkhorn distances, resulting from optimal transport, could be good measures for sequence similarity (using alignment scores as a proxy for sequence similarity). If Sinkhorn distances are a good measure for sequence similarity, it is expected for Sinkhorn distances to have an inverse relationship to alignment scores. Highly similar sequences will have a high alignment score, and should have a small Sinkhorn distance. Conversely, sequences that are not alike have a low alignment score and should have a high Sinkhorn distance. Because computing optimal transport for the entire dataset would be too computationally expensive, one hundred proteins from the collected dataset were sampled at random. These were compared in a pairwise manner using both optimal transport and local alignment. The 985 reference proteomes consisted of a total of 91193 proteins. To at least sample approximately 1% of the entire dataset, this analysis was repeated ten times, each time sampling a new set of hundred proteins from the dataset.

Furthermore, an optimal choice of the value of λ was based on the Pearson correlation between the Sinkhorn distances and the pairwise local alignment scores of these hundred proteins. For increasing values of λ and fixed value of k, hundred proteins were sampled again and compared using optimal transport and pairwise local alignment. Subsequently, the Pearson correlation was calculated between the resulting Sinkhorn distances and alignment scores for each value of λ. Again, to at least sample a representative number of proteins from the entire dataset, this analysis was repeated ten times.

Subsequently, an appropriate value for k had to be chosen. In comparing entire proteomes using k-mers, different values for k can lead to differences in tree topology (Wu *et al.*, 2009). Several research groups have reported different optimal values for k (both at genomic and proteomic level) (Das *et al.*, 2017; Mahmood *et al.*, 2011; Wu *et al.*, 2009; Yu *et al.*, 2010; Zhang, 2016). There is no consensus about a universal optimal value of k, as k can be used in different methods and different measures can be used to select the optimal value of k. This optimum might also be dependent on the data used. Furthermore, the combination of using k-mers for protein comparison together with optimal transport is new, thus optimal values for k used in other methods might be sub-optimal in this scenario. To cope with this, three values of k were chosen ranging from low to high values. More specifically, the value of k was either 3, 9 or 15. In the subsequent analyses, all three values were used to construct phage trees (explained in the next paragraph) and the resulting trees were compared to assess which value of k works best.

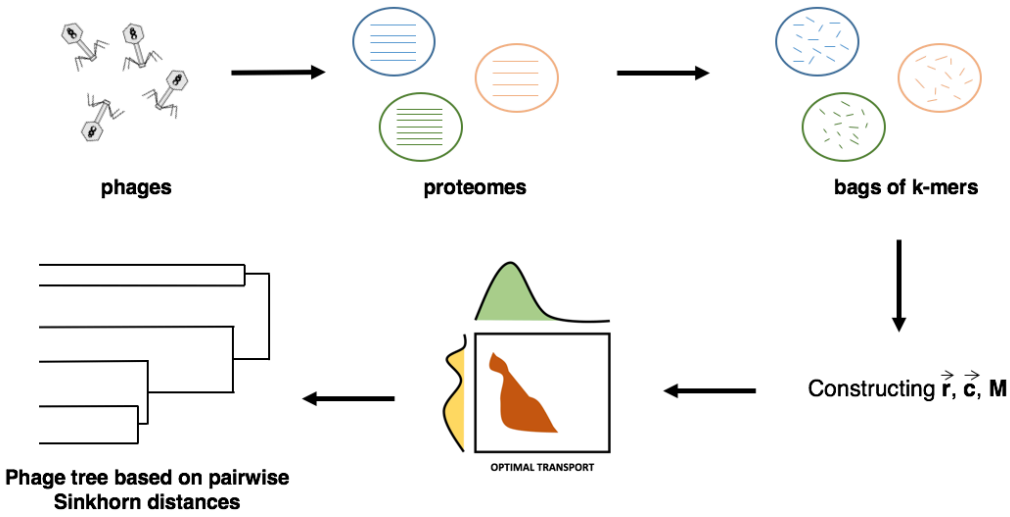


Figure 2.2: Schematic overview of the followed steps to construct a phage tree using Sinkhorn distances calculated with optimal transport.

First, proteomes of phages are collected. In a second step, each proteome is split up in k-mers and each unique k-mer is counted. Subsequently, these counts are normalized to represent probability distributions as vectors **r** and **c**. After constructing the cost matrix M based on Hamming distances between the k-mers, optimal transport can be applied using the Sinkhorn-Knopp algorithm. This results in a Sinkhorn distance for each pairwise comparison of two phages. Finally, these distances are used to construct a phage tree with the neighor-joining method.

Afterwards, part of the gathered data was used to construct a phage tree based on Sinkhorn distances. More specifically, seven phages were chosen for their small proteomes in order to minimize computational load. A general overview of the construction of the phage tree is visually represented in Figure 2.2. Every phage proteome was compared to every other phage proteome in a pairwise manner by calculating probability distributions from k-mer counts of each of the proteomes. These probability distributions represent the vectors **r** and **c**. The cost matrix M was constructed based on Hamming distances[2] between each of the k-mers in every two distributions that were compared. Finally, using vectors **r** and **c** together with the cost matrix M, optimal transport was applied using the Sinkhorn-Knopp algorithm. The resulting Sinkhorn distances were used to contruct a phage tree based on neighbor-joining clustering (Saitou and Nei, 1987). All data manipulation steps and optimal transport were executed in Python. The workflow represented in Figure 2.2 was also implemented in Python and is given in appendix B. All supporting scripts are also available in digital appendix A.

Finally, the obtained trees (for different values of k) were compared to a tree constructed for the same seven phages based on an altered version (i.e. by calculating normalized tBLASTx scores between viral genomes) of the Phage Proteomic Tree

[2]The Hamming distance is calculated by counting the number of corresponding positions that don't match between two strings

method by Rohwer and Edwards (Rohwer and Edwards, 2002). More specifically, an online tool called ViPTree was used to generate the proteomic tree (Nishimura *et al.*, 2017).

2.7 Results and discussion

2.7.1 Relationship between Sinkhorn distances and alignment scores

Figure 2.3 shows the relationships between Sinkhorn distances and alignment scores for on hundred proteins randomly sampled from the dataset using different combinations of values for λ and k. This analysis was repeated ten times. The other nine figures are given in Figures A.1 and A.2 of appendix A. There is a clear inverse relationship between Sinkhorn distances and alignment scores using a low value of k and high value of λ. This is not unexpected. A high alignment score corresponds to highly similar proteins, which should indeed correspond to a low Sinkhorn distance between the k-mer probability distributions of these proteins. Furthermore, alignment tries to match single amino acids. This process is approximated better by optimal transport using a low value for k, corresponding to small k-mers, rather than a high value of k. Additionally, using a high value for λ corresponds to a small amount of entropic regularization, minimizing the overestimation of actual distance (without regularization). No clear trend is observed in the two upper plots, using a low value of λ. This indicates that using a high regularization might not work well in comparing proteins and proteomes. The bottom-right plot, corresponding to a high value of k and a high value of λ, also shows an inverse trend. However, this trend is less clear in comparison to the bottom-left plot, indicating that lower values of k might not work well when constructing phage trees.

2.7.2 Tuning of λ

Figure 2.4 shows the Pearson correlation between the resulting Sinkhorn distances and alignment scores for each value of λ, after comparing proteins via both optimal transport and pairwise local alignment. As mentioned in Section 2.6, the analysis was repeated ten times. Every line corresponds to one sampling of hundred proteins from the dataset. From this plot, it is clear that higher values of λ correspond to a higher correlation between alignment scores and Sinkhorn distances, indicating that the use of higher values for λ is appropriate in optimal transport. The relationships converge

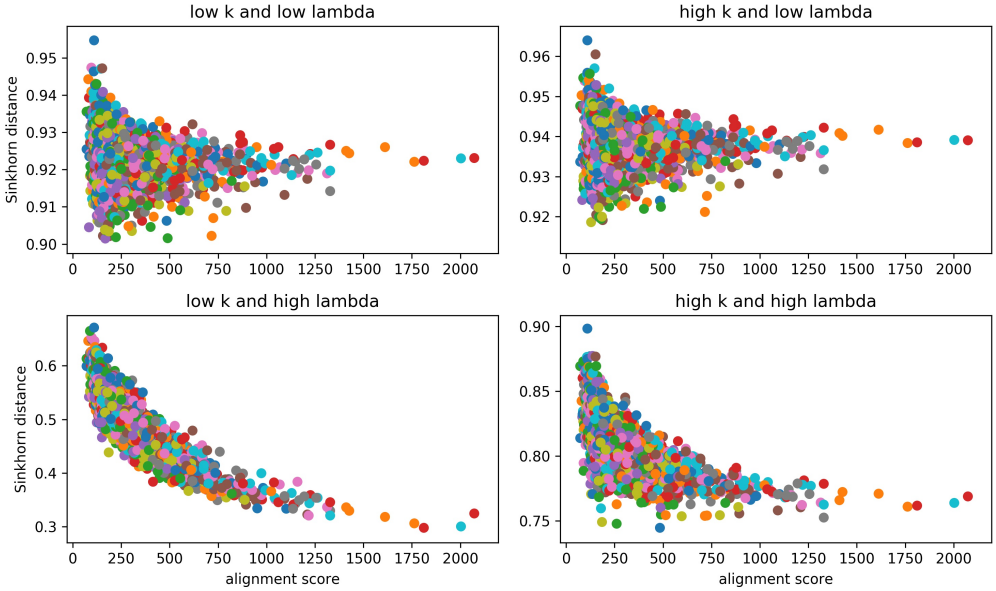


Figure 2.3: Relationship between Sinkhorn distances and alignment scores for combinations of low and high values of k and λ, using one hundred proteins sampled at random from the dataset.

Hundred proteins from the collected dataset were sampled at random, and these were compared in a pairwise manner using both optimal transport and local alignment. The plots above show the alignment score in function of the Sinkhorn distance for pairwise comparison of the proteins. In the upper-left plot, λ was equal to 0.1 and k equal to 3. In the upper-right plot, λ was equal to 0.1 and k equal to 15. In the bottom-left plot, λ was equal to 30 and k equal to 3. In the bottom-right plot, λ was equal to 30 and k equal to 15.

for values of λ larger than 30. Therefore, λ was chosen to be 30 in the subsequent construction of phage trees.

2.7.3 Tree construction with optimal transport and comparison with Phage Proteomic Tree

Using optimal transport with a value for λ of 30 and values for k of 3, 9 and 15, three phage trees were constructed. Additionally, using ViPTree, a proteomic tree was constructed (Nishimura *et al.*, 2017). Figure 2.5 shows the resulting trees. It is hard to know what the correct tree (i.e. the one representing the actual evolutionary relationship) looks like. However, according to the ICTV classification, Pseudomonas phage YuA, Mycobacterium phage Wonder and Streptomyces phage phiSASD1 should cluster together as they are all part of the *Siphoviridae* family. Pseudomonas phage F116 and Vibrio phage Vc1 are both part of the *Podoviridae* family. Furthermore, Stenotrophomonas phage Smp131 is part of the *Myoviridae* family. Lastly, Bdellovibrio phage phiMH2K is an ssDNA phage part of the *Microviridae* family. The phage tree (d) constructed based on the 'Phage Proteomic Tree' method matches the ICTV

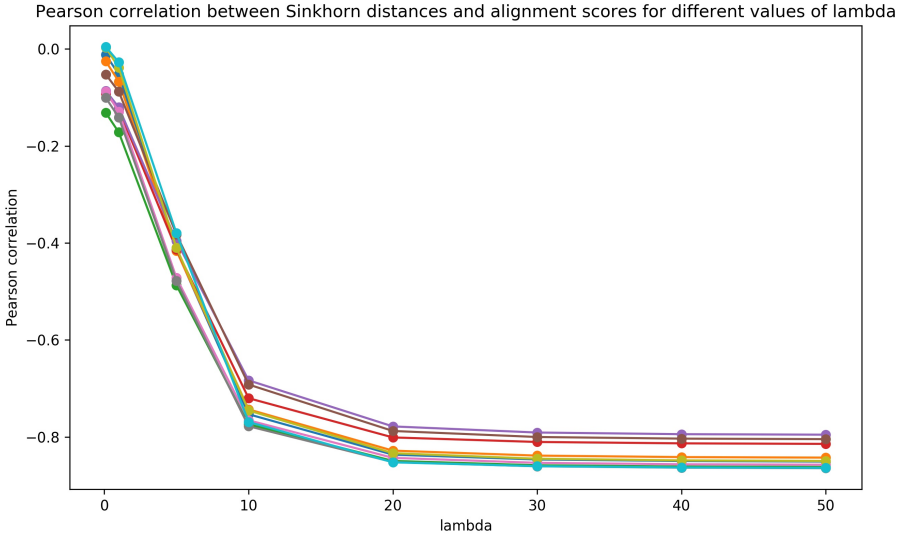


Figure 2.4: Pearson correlation between the resulting Sinkhorn distances and alignment scores for each value of λ, after comparing proteins via both optimal transport and pairwise local alignment.

Hundred proteins were sampled at random from the dataset, after which these were compared using optimal transport and pairwise local alignment. For optimal transport, different values of λ were used, ranging from 0.1 to 50. From the resulting Sinkhorn distances and alignment scores, the Pearson correlation was calculated and plotted for each value of λ. The analysis was repeated ten times, corresponding to the ten lines on the plot.

classification quite well. Surprisingly, Bdellovibrio phage phiMH2K clusters together with Vibrio phage Vc1 while these phage are not even classified in the same order according to the ICTV. Furthermore, this clustering is present in all four constructed phage trees. Apparently, these phages do exhibit significant proteome similarity, despite using a different type of genome (dsDNA vs. ssDNA). Considering the trees constructed with optimal transport, none of them match exactly with the classification of either the ICTV nor the 'Phage Proteomic Tree'. In tree (a), all three members of the *Siphoviridae* family are present in different clusters. Moreover, both in tree (a) and (b), Pseudomonas phage YuA and Mycobacterium phage Wonder appear to be more similar to Vibrio phage Vc1 and Bdellovibrio phage phiMH2K instead of being similar to Streptomyces phage phiSASD1.

Among the trees constructed with optimal transport, tree (c) corresponds best with both the ICTV classification and the 'Phage Proteomic Tree'. This is surprising because based on Figure 2.3 it would be expected for a low value of k to result in a tree that resembles the 'Phage Proteomic Tree' better. The 'Phage Proteomic Tree' also uses alignment (which is approximated best by using a small value of k in optimal transport). Perhaps, at the proteome level, using small k-mers potentially works less well because small identical k-mers can occur more often by chance. In addition, optimal

transport uses the cost matrix M to compute the Sinkhorn distance. Because of this, even similar (instead of exactly matching) k-mers influence the resulting distance. As a result, the probability distributions computed from the k-mers can be biased towards more similarity, resulting in a lower Sinkhorn distance. On the other hand, larger identical k-mers are less likely to occur by chance, mitigating this effect.

Another remark here is that using optimal transport with regularization, the actual distances (as given by optimal transport without regularization) are still overestimated by a tiny amount. A potential solution would be to substract the Sinkhorn distances between each proteome and itself from the distance between both proteomes.

2.8 Conclusion

Chapter two shows the use of optimal transport to study biological data in a way that was not attempted before. Proteomes of seven phages were compared in a pairwise manner. From the resulting Sinkhorn distances, classification trees were constructed and compared. The results indicate that optimal transport can be a decent measure for similarity among proteomes and proteins. However, the optimal value of k is still unclear. While lower values of k result in a higher correlation between Sinkhorn distances and alignment scores, higher values of k seem more appropriate in constructing classification trees.

In the next chapter, optimal transport will be applied again, now at the protein level to compare the proteomes of three T7-like phages. These phages all exhibit high genome and proteome similarity but infect different hosts. By applying optimal transport at the protein level, it is attempted to identify unique proteins across the T7-like phage proteomes. Additionally, these unique proteins should in part correspond to the proteins needed for different host specificity.

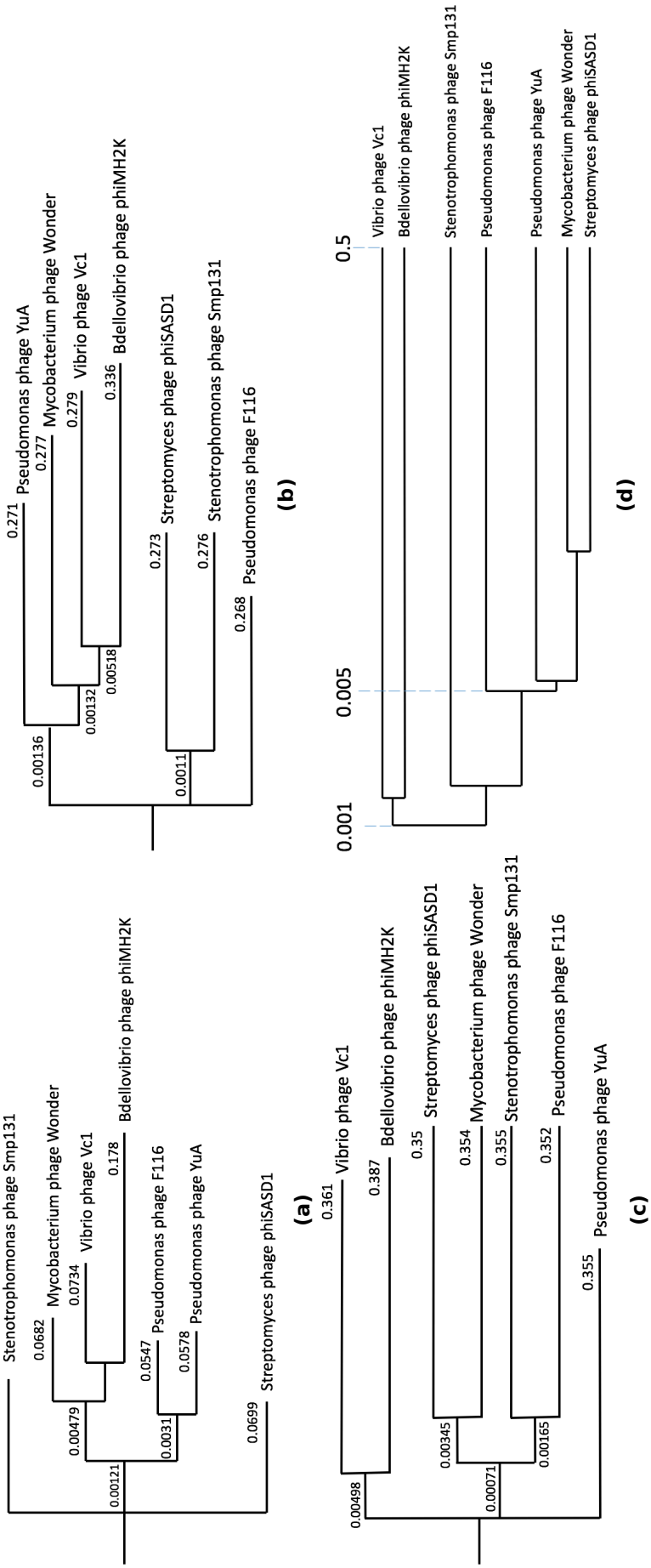


Figure 2.5: Phage trees of seven phages constructed using optimal transport (a-c) and Phage Proteomic Tree method (d).

Optimal transport was used to construct phage trees with λ equal to 30 and k equal to 3 (a), 9 (b) or 15 (c). Tree (a) was constructed with k equal to 3, tree (b) was constructed with k equal to 6 and tree (c) was constructed with k equal to 15. The proteomic tree was constructed using ViPTree (Nishimura *et al.*, 2017).

CHAPTER 3

The importance of specific proteins in bacteria-phage interactions

3.1 Understanding phage specificity

A phage's host range is defined by the breadth of bacteria the phage is able to infect (Hyman and Abedon, 2010). Host range and phage infectivity are known to depend on several factors. These include adsorption, structural changes of both the phage and the bacterial host, transport of DNA or RNA into the host cell and avoidance of degradation of the nucleic acid. Therefore, infectivity is depended on the phenotype of both phage and host. This infectivity, and thus the host range, can change over time as phages and hosts co-evolve (Koskella and Meaden, 2013; Leite *et al.*, 2017). This co-evolution of phage infectivity versus bacterial resistance to phages has two important consequences: shaping microbial communities and expanding genetic diversity among bacterial species. As phages are explicitly dependent on their hosts for reproduction, host abundance is an important determinant of environmental fitness of a phage. When abundance of suitable hosts is high, phages that are able to infect this host will thrive, effectively bringing this host abundance down. This process is known as the kill the winner hypothesis. As a consequence, microbial abundance continuously changes in time. Additionally, phages also influence bacterial genetic diversity. Indeed, phage-mediated horizontal gene transfer is an important factor in bacterial evolution, which was discussed in Section 1.2 (Chaturongakul and Ounjai, 2014).

In addition, most phages infect only a subset of bacterial species. Moreover, many phages seem to only infect a single species or even a few strains within a species. Phages have a tendency to specialize on a small number of hosts. This is because there is a trade-off between evolutionary fitness of the phage, and the broadness of its host range. A broader mechanism of infectivity, leading to a broader host range,

tends to be less efficient than specialized mechanisms that are optimal for infection of some specific host (Koskella and Meaden, 2013). On the other hand, phages can expand their host range, for example in environments where there exists strong competition for hosts. Oddly enough, some generalist phages fail to infect specific species or strains within the genera that these phages can infect. This observation indicates that specific rules regarding host range are complex. Finally, phage specificity also depends on environmental conditions such as local resources, temperature and dosage of the phage (Koskella and Meaden, 2013).

Nevertheless, phage specificity is in part explained by specific phage proteins. These proteins recognize and bind to certain receptors on the bacterial cell wall as the first step to phage propagation (Samson *et al.*, 2013). The next section explores bacterial receptors and their importance in the phage-host interaction. Thereafter, receptor binding proteins (RBPs) used by phages to interact with these receptors are discussed in more detail. Afterwards, optimal transport will be adopted to compare several phages that have similar genomes but nevertheless infect different hosts. The hypothesis is that optimal transport can help identify the proteins that are unique in these similar phages, among which proteins that are responsible for the difference in host specificity are expected.

3.2 Bacterial cell surface receptors

Numerous bacteria possess cellular appendages or structural components that extend beyond the plasma membrane or outer membrane. Besides flagella and pili, the outer membrane of Gram-negative bacteria consists of lipopolysaccharides, porins, transport proteins and other membrane-associated or embedded proteins (Lindberg, 1973). On the other hand, Gram-positive bacteria mostly expose (lipo)teichoic acids on the cell surface. Phages can utilize the components that are exposed on the cell surface for phage adsorption (Rakhuba *et al.*, 2010). For example, *Salmonella* phages can recognize glycolipids, membrane proteins (OmpF, BtuB, TolC) or flagellar proteins as receptors for phage adsorption (Chaturongakul and Ounjai, 2014). Figure 3.1 displays these various receptors of *Salmonella*.

The presence of specific components of these receptors can be essential to the phage adsorption process. For example, the presence of D-glucose in teichoic acids of Bacillus subtilis plays a key role in adsorption of phages specific to B. subtilis (Rakhuba *et al.*, 2010). Lastly, bacteria can also modulate the availability of their receptors (also discussed in Section 1.3). Conformational changes or alterations in spatial distributions of the receptors can both lead to a reduced ability of the phage to adsorb to the

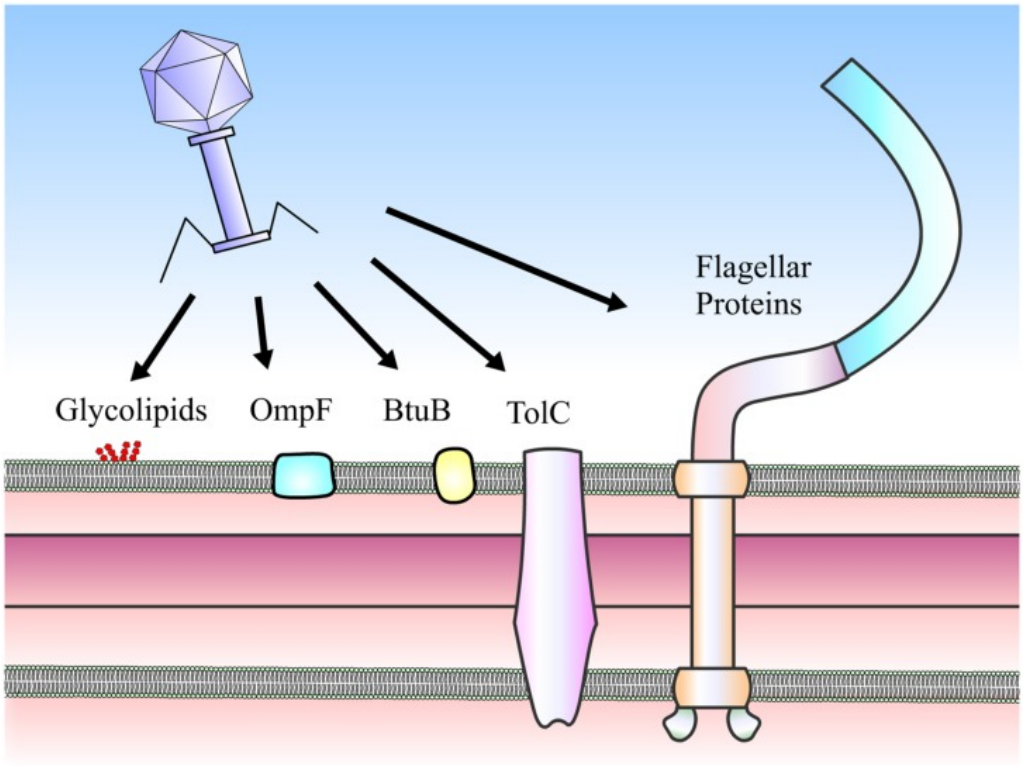


Figure 3.1: Various receptors of *Salmonella* bacteria used for phage adsorption.

Salmonella phages can use various receptors for phage adsorption, including glycolipids, membrane proteins such as OmpF, BtuB, TolC or flagella proteins (Chaturongakul and Ounjai, 2014).

bacterial cell surface (Moldovan *et al.*, 2007). Some bacteria express phage receptors in a stochastic manner. This can be a consequence of specific environmental conditions or as a response to specific stimuli. Phages can cope with this stochasticity by encoding multiple RBPs, which have affinity for different receptors on the cell surface, or by mutations in specific genes related to these RBPs. Figure 3.2 shows how phages can interact with different receptors using different RBPs. These remarks highlight the biological complexity that can arise in phage-bacteria interactions.

3.3 Receptor binding proteins

Most reported bacteriophages belong to the order of the *Caudovirales*. These phages contain, besides a chromosome and a capsid, also a tail. The order of the *Caudovirales* is divided into three families based on distinct morphological traits of these tails. The *Siphoviridae* family has long, flexible and non-contractile tails. The *Myoviridae* family is characterized by long, rigid and contractile tails. Finally, the *Podoviridae* have short and non-contractile tails (Ackermann, 2007).

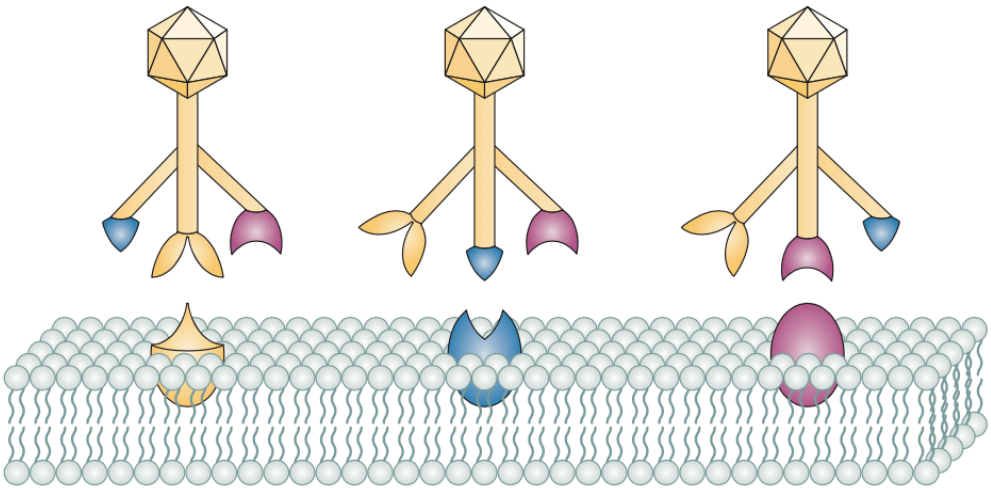


Figure 3.2: Phages can interact with different bacterial receptors by using different receptor binding proteins.

Some bacteria express their cell surface receptors in a stochastic manner, corresponding to differences in environmental conditions or specific stimuli. Phages are still able to interact with such bacteria by encoding several receptor binding proteins that have affinity for different receptors (Samson *et al.*, 2013).

Although the three families have distinct tails, all of them possess two common properties. All tails form tubular channels through which the dsDNA chromosome exits the capsid and all tails carry fibers or spikes. The RBPs of the *Caudovirales*, that recognize potential hosts when virions come in contact with them, are located on these fibers or spikes (Fokine and Rossmann, 2014). Because host recognition is an essential step in phage propagation, these tail structures are known to be the most rapidly evolving part of the phage genome (also discussed in Section 1.3). As a result, RBPs located on the tail are spectacularly diverse and capable of recognizing almost every host surface component (Charurongakul and Ounjai, 2014).

As phages cannot move independently, the adsorption process of phages to their bacterial host is the result of random phage-host collision in environments where both the phage and suitable host cells are present. Phage adsorption often implies two steps (Dupont *et al.*, 2004). First, contact with a primary receptor establishes a weak and reversible bond. This brief interaction is often enough to encourage the phage to start exploring the bacterial cell surface. In the second step, the phage continues to locate its secondary receptor, which results in an irreversible binding to the cell surface. Both steps include mechanisms that are specific for the phage-host interaction (Rakhuba *et al.*, 2010). In general, it is this irreversible binding that triggers the release of the phage genome into the host cell (Moldovan *et al.*, 2007; Rakhuba *et al.*, 2010). Additionally, phages that infect gram-negative bacteria may be equipped with enzymes close to their RBPs that degrade the polysaccharides that

constitute the outer membrane. It is only when the tail structure reaches the cell membrane that genome delivery is triggered (Youle, 2017).

Phages that possess similar genomes can nevertheless infect different bacterial hosts. An example of this was studied by Le *et al.*, where two *Pseudomonas aeruginosa* phages (PaP1 and JG004) were characterized at the genome level. These two phages exhibit high genome similarity, despite having different hosts (at the subspecies level). The reason for this is that both phages encode different RBPs. This can be observed at the genomic level, which is visualized in Figure 3.3. At most genetic loci, both phages appear remarkably similar. Some genetic loci, however, exhibit much less similarity. These genes encode putative tail fiber proteins (Le *et al.*, 2013).

In addition, Le *et al.* (2013) characterized spontaneous mutants of phage JG004 that were capable of infecting the same *P. aeruginosa* host as phage PA1. By designing primers to amplify the baseplate region and tail fiber region of one of these mutants, the authors detected a single point mutation in an open reading frame (ORF) that is predicted to encode a putative tail fiber protein. Finally, the authors constructed a recombinant phage where the ORF responsible for encoding the putative tail fiber protein of phage PaP1 was switched for the corresponding ORF of phage JG004. The genetic loci for this phage are also visualized in Figure 3.3. The resulting recombinant phage was tested on both its original host and the host of phage JG004 through a spot assay and adsorption assay. Both assays showed the recombinant phage's ability to infect phage JG004's host but not its original host. Together, these observations demonstrate that RBPs are an essential determinant of phage-host specificity (Le *et al.*, 2013).

Other examples are the phages in the T7 supergroup. Scholl *et al.* (2014) performed a genomic analysis of the closely related phages SP6 and K1-5. Phage SP6 infects *Salmonella typhimurium* LT2 and phage K1-5 infects *Escherichia coli* serotypes K1 and K5. In doing so, the authors found that both genomes differ the most in genes that are likely responsible for encoding tail appendages. Two unique ORFs of phage K1-5 encode a lyase and endosialidase that allow the phage to degrade surface components, which is essential for the interaction with *E. coli* strains K5 and K1, respectively. Neither of those proteins are present in phage SP6. However, this phage encodes a protein that strongly resembles tailspike proteins of *Salmonella* phages ST46T and P22. Phage SP6 also encodes another protein that slightly resembles the endosialidase of phage K1-5. The authors speculate that this protein is also part of the tail structure and enables specificity towards another unknown bacterial host (Scholl *et al.*, 2014). Also concerning phages of the T7 group, Ando *et al.* engineered synthetic phages by swapping single or multiple tail components between phages with both highly similar and less similar genomes. Their results indicate that between the two closely related

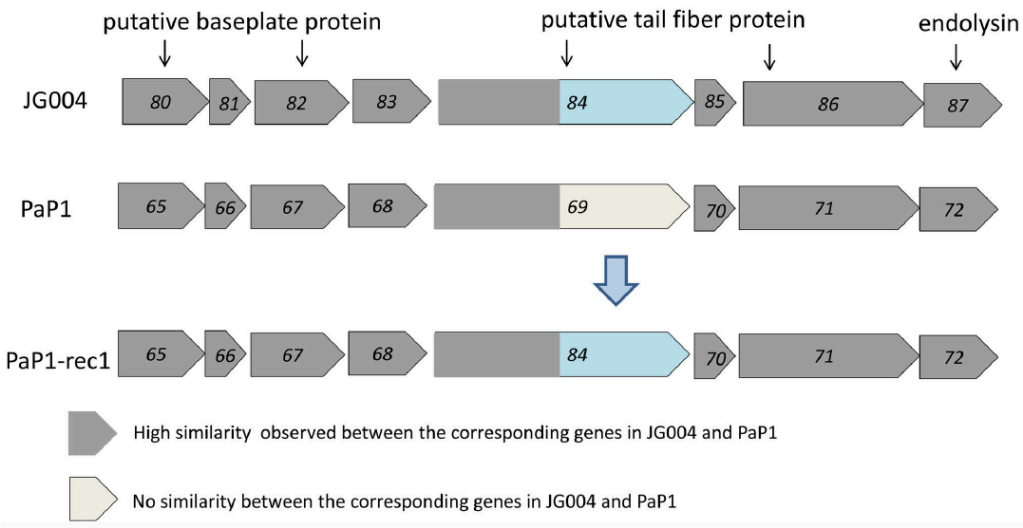


Figure 3.3: Visual representation of the genetic loci encoding baseplate proteins, putative tail fiber proteins and endolysins for phages JG004, PaP1 and recombinant phage PaP1-rec1.

The genetic loci of phages JG004 and PaP1 differ mainly in the ORFs 84 and 69, encoding putative tail fiber proteins. By switching ORF84 with ORF69 in phage PaP1, the new recombinant phage (PaP1-rec1) is now able to infect the host of JG004, but not the original host of PaP1 anymore (Le *et al.*, 2013).

phages T7 and T3, the main determinant of host specificity is the C-terminal domain of their tail fiber proteins. Switching only this C-terminal domain is sufficient to switch host specificity between both phages. The authors also switched complete between phage T7 and the less similar Klebsiella phage K11. Here, multiple tail components (including the tail fiber protein) had to be switched in order to switch host specificity between the phages (Ando *et al.*, 2015).

3.4 Applying optimal transport at the protein level

In this section, optimal transport will be applied to study phage proteomes (i.e. sets of proteins) at the level of the individual proteins. The goal of this chapter is to identify proteins that are unique in a particular phage proteome and investigate whether these identified proteins are related to host specificity. Two approaches can be adopted. First, by studying phages with the same host, factors responsible for host specificity can be identified if they are shared between different phages. However, two phages can infect the same host in different ways, corresponding to different surface receptors (Chaturongakul and Ounjai, 2014). The second approach is to study phages that are closely related, but have different hosts nonetheless. Here, proteins that are not shared between them are likely to be responsible for the difference in host specificity. This approach was adopted in the two examples in Section 3.3. This strategy is the

Table 3.1: Phage proteomes used for the comparison of proteins using optimal transport. The table lists the proteome ID, taxonomy ID and protein count of these phages.

Proteome ID	Organism	Taxonomy ID	Protein count
UP000000840	Enterobactera phage T7	10760	57
UP000008891	Erwinia phage vB_EamP-L1	1051673	51
UP000000335	Salmonella phage Vi06	866889	47

inverse of the first approach: now similar phages with different hosts are picked deliberately to find unique proteins. The second strategy will be adopted here for three phages of the T7 virus group (also referred to as the T7-like phages). As explained by Scholl *et al.*, phages from the T7 group exhibit high genome similarity (Scholl *et al.*, 2014). However, some of these phages infect different bacterial genera. Therefore, these phages form an interesting case to look for unique proteins among highly similar proteomes. Additionally, the three particular phages were chosen for their relatively good protein annotation in UniProt. The different phages are presented in Table 3.1.

These phage proteomes are compared in a pairwise manner. For each proteome pair (i, j), proteins in each of the proteomes are represented by p_{pi} and p_{qj} where $p = 1, \ldots, P$ and $q = 1, \ldots, Q$. Here, P and Q depict the number of proteins in proteomes i and j, respectively. Every protein p_{pi} of proteome i is compared to every proteins p_{qj} of proteome j. Optimal transport was used for this comparison; in the same way it was used before (described in Section 2.6). Every protein is split in overlapping k-mers, which are counted. By normalizing these counts, a probability distribution (of k-mers) for each protein is obtained. Just as in Section 2.6, these probability distributions represent the vectors r and c, which can be compared by using optimal transport. The cost matrix M is again implemented as the Hamming distances between every pair of k-mers from different proteins. Based on Figure 2.3, the value of k was set to 3 and the value of λ was set to 30.

The probability distribution of each protein p_{pi} in the first proteome is then compared to the probability distribution of every protein p_{qj} in the second proteome. The result of this comparison is a Sinkhorn distance $d_{pi,qj}$ between every two proteins p_{pi} and p_{qj} of proteomes i and j, respectively. This Sinkhorn distance represents a measure for similarity between the two proteins that were compared. For a pair of two proteomes (i, j), this results in a distance matrix, representing all pairwise distances between every protein in the first proteome and every protein in the second proteome.

The specific goal of this chapter is to find unique proteins among three selected proteomes of T7-like phages. A protein unique in one of the proteomes should have a unique probability distribution in comparison to the probability distributions of the

proteins it is compared to. When comparing this unique probability distributions to the other distributions, it will result in large Sinkhorn distances. On the other hand, if a protein matches to some protein in another proteome, the probability distributions of the matching proteins will be alike, resulting in a smaller Sinkhorn distance compared to distances between proteins that don't match. This smaller Sinkhorn distance indicates similarity between the two matching proteins. Figure 3.4 visually represents this method by depicting two proteins that are compared to another proteome in a fictional example. The first protein (protein p) has some corresponding similar protein in the proteome (consisting of ten proteins) which it is compared to. The second protein (protein q) is unique, i.e. it does not match with any protein in the proteome it is compared to. The Sinkhorn distance is smaller between protein p and its corresponding similar protein, relative to the other proteins it was compared to. The very light color on the heat map (distance between protein p and protein d of the other proteome) indicates this similarity. In the ideal case, the Sinkhorn distances from protein q to the proteins in the other proteome will all be high as there is no corresponding similar protein. All the colors in the second row of the heat map are then sufficiently dark, indicating no similarity between protein q and the proteins it was compared to. The method in this chapter will attempt to identify the proteins that have a match in another proteome. By subsequently discarding these proteins, the unique proteins (that don't match with any protein in another proteome) can be identified.

When using a small value of k, two long proteins could have similar probability distributions by chance. As explained in Section 2.7, this results in a lower Sinkhorn distance, potentially leading to a false result (i.e. falsely identifying a match). To avoid false results, a higher value of k could be chosen. However, this solution might not be optimal as the correlation between alignment scores and Sinkhorn distances decreases for higher values of k, as was previously shown in Figure 2.3. Another solution is to statistically test whether the probability distributions of two proteins are significantly more alike than expected by chance. Comparable to approaches used in sequence alignment, one of the proteins can be randomly shuffled and its probability distribution can be compared to the probability distribution of the other considered protein (Lipman *et al.*, 1984). Therefore, a thousand variants of the second protein were constructed in which the AAs of the protein were randomly shuffled. Subsequently, optimal transport was used to compare these thousand variants to the first protein using the same hyperparameters as before. The resulting thousand Sinkhorn distances were then used to statistically test whether two proteins under consideration were significantly more similar than expected by chance. More specifically, a one-sample t-test was performed with the thousand Sinkhorn distances and the observed Sinkhorn distance $d_{pi,qj}$ using the SciPy package in Python (Jones *et al.*, 2001). As the interest lies in discriminating protein pairs with a significantly smaller Sinkhorn

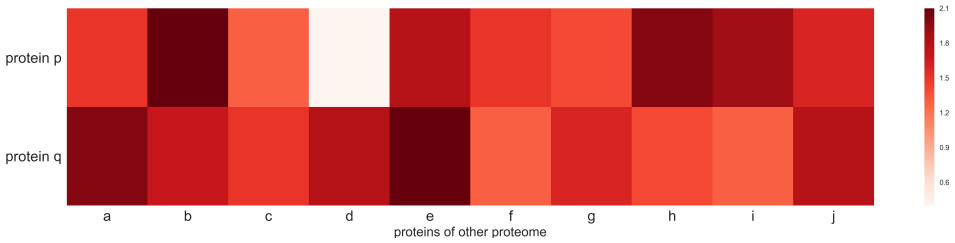


Figure 3.4: Visual representation of followed method for a fictional example of two proteins from one proteome being compared to another proteome.

The figure displays the Sinkhorn distances (as heat map) for two proteins from one proteome which were compared to ten proteins in some other proteome, by using optimal transport. Protein *p* has some corresponding similar protein in the other proteome. The comparison between protein *p* and its similar protein *d* will result in a lower Sinkhorn distance, relative to the other proteins that protein *p* was compared to. The very light color on the heat map indicates the similarity between both proteins. Protein *q* has no corresponding similar protein in the other proteome which means there will not be any Sinkhorn distance significantly smaller than the Sinkhorn distances resulting from comparisons with other proteins in the proteome. Indeed, all colors in the second row of the heat map are sufficiently dark, indicating no significant similarity between protein *q* and any of the proteins in the other proteome.

distance from other protein pairs, the *t*-test was also one-tailed (i.e. testing whether the observed Sinkhorn distance $d_{pi,qj}$ is significantly smaller than the Sinkhorn distance expected by chance). By using the same value of k for both computing the observed Sinkhorn distance and the thousand Sinkhorn distances expected by chance, the effect of using a low value of k is canceled out.

However, the effect of random shuffling on protein structure and functionality also has to be taken into account. Amino acids have a propensity to either reside in an α-helix, β-sheet or neither of both. Shuffling proteins at random will most likely obliterate these secondary and tertiary protein structures. Therefore, it is expected that functional proteins are more similar when compared to each other than when compared to random, likely non-functional proteins. Even if the difference in similarity between both cases is small, it could be statistically significant. This represents a shortcoming in the statistical testing. As a result, statistical testing can still falsely identify a match while it might not be biologically relevant. To further discriminate between these (small) significant differences and the larger differences between two proteins that show relevant similarity, effect sizes were computed for every protein-protein comparison. Effect sizes are computed by subtracting the mean of expected distances from the observed distance and subsequently dividing by the standard deviation of the expected distances. These effect sizes indicate the magnitude by which the observed Sinkhorn distance $d_{pi,qj}$ deviates from the mean of expected distances (obtained from the comparisons with shuffled variants). For proteins that show considerable similarity, the Sinkhorn distance will be much lower than the mean of expected distances, resulting in a high, negative effect size. As such, effect sizes allow for dis-

crimination between statistically significant matches that are or are not biologically relevant. All analyses were implemented in Python. The scripts and data used for these analyses are given in digital appendix B.

Taken together, proteins that have a match somewhere in another proteome were identified as having a significantly lower Sinkhorn distance for this match, measured by the resulting *p*-value after statistical testing, as well as a high, negative effect size for this match. These proteins were filtered out and the remaining proteins for each proteome were identified as the unique proteins for that particular proteome. These proteins were further investigated for their described biological function by manually searching UniProt and NCBI databases, as well as literature. If the identified protein did not have any known biological function, a BLASTP search was performed against UniProt KB to discover similar proteins with known functions. These unique proteins are further discussed in the section below.

3.5 Results and discussion

3.5.1 Identification of unique proteins in the comparison between Enterobacteria phage T7 and Erwinia phage vB_EamP-L1

Figure 3.5 shows the histogram of effect sizes (for every protein pair) in the comparison between Enterobacteria phage T7 and Erwinia phage vB_EamP-L1. The cut-off value for biological relevance was chosen to be -2, indicating that the observed Sinkhorn distance deviated more towards negative values at least two standard deviations from it's expected value. The colors on the histogram indicate the discimination between biologically relevant and non-relevant protein pairs. Every biologically relevant protein pair had a *p*-value of approximately zero. The matrices with *p*-values and effect sizes for this comparison are given in digital appendix B.

The left of Figure 3.6 displays the distance matrix resulting from the comparison of Enterobacteria phage T7 with Erwinia phage vB_EamP-L1. The right side of the figure visually represents the protein pairs that were labeled as significant after statistical testing and applying a cutoff for effect size. In the Enterobacteria phage T7 proteome, 47 of 57 proteins had a significant and biologically relevant match in the proteome of Erwinia phage vB_EamP-L1. Likewise, 44 of 51 proteins in the Erwinia phage vB_EamP-L1 proteome had a significant and biologically relevant match in the Enterobacteria phage T7 proteome. The proteins that are unique to one of both proteomes (i.e. not having a significant match) are listed in Table 3.2.

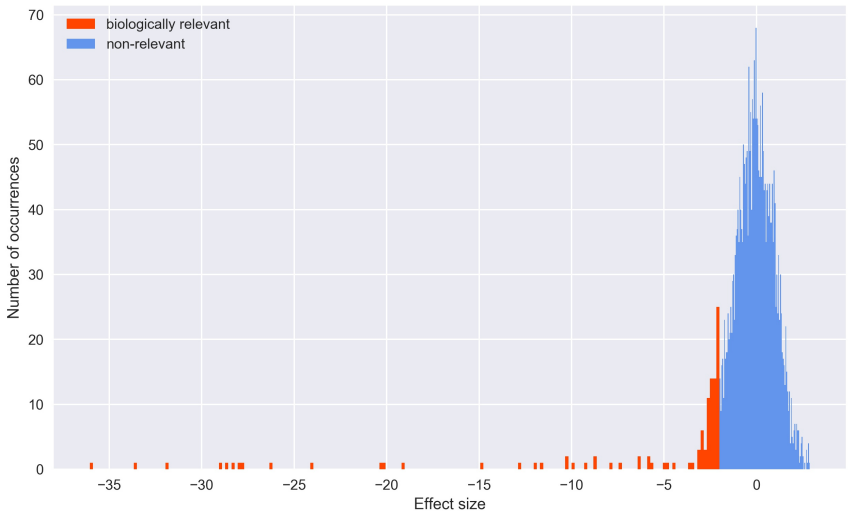


Figure 3.5: Histogram of the effect size of every protein pair resulting from the comparison of Enterobacteria phage T7 and Erwinia phage vB_EamP-L1.

The figure shows the effect sizes for every pairwise comparison of the proteins of phages Enterobacteria phage T7 and Erwinia phage vB_EamP-L1. Protein pairs were labeled as biologically relevant if their effect size was smaller than -2. Every biologically relevant protein pair had a *p*-value of approximately zero.

In the list of unique proteins of Enterobacteria phage T7, about half of them have a described biological function in either Uniprot or NCBI databases, which are given in Table 3.2. Gene product (Gp) 0.4 has a role in the inhibition of host cell division. While this protein is not essential for phage infectivity, it is able to increase competitiveness of the phage by inhibiting the function of filamenting temperature-sensitive mutant Z (FtsZ) division protein. By doing so, more resources are freed up for use by the phage, which can create an advantage in environments where rapid phage replication can lead to faster infection of more bacterial hosts (Kiro *et al.*, 2013). Gp0.3 is a protein that protects the phage host DNA by inhibiting nucleases employed by the bacterial host (Studier, 1975). Additionally, protein 4.1 provides helicase and primase functionalities necessary for DNA replication (Mendelman *et al.*, 1992). Finally, protein Gp17 is a tail fiber protein. As discussed in Section 3.1, tail fiber proteins interact with host membrane receptors, allowing the phage to adsorb to its potential host (Cuervo *et al.*, 2013). Taken together, all these proteins are related to successful phage host infectivity, either directly by providing ways to recognize host membrane receptors, helping in cell lysis or redirecting host resources to virion assembly, or indirectly by protecting the phage genome inside the host cell and aiding in DNA replication.

For proteins of Enterobacteria phage T7 that did not have a specified biological function or gene ontology, a BLASTP search against all proteins in UniProt KB was con-

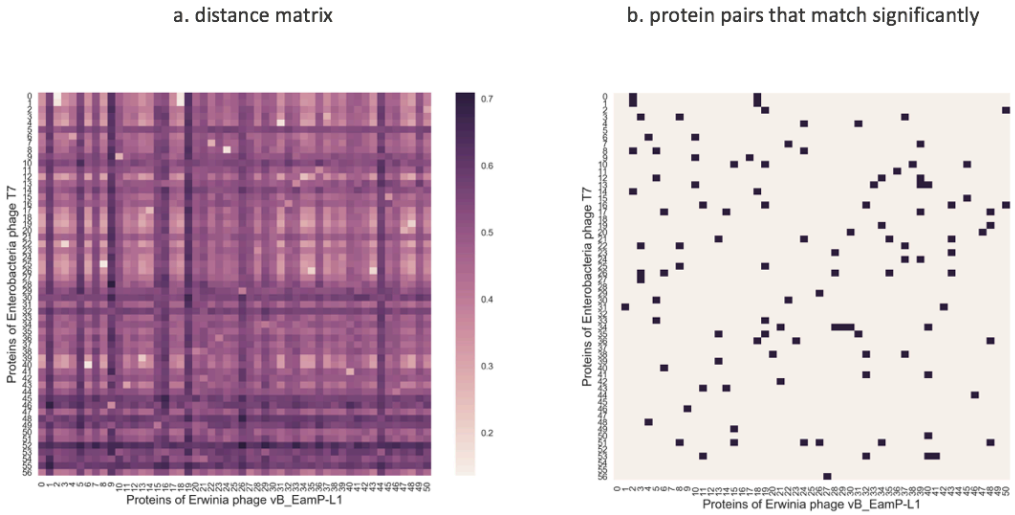


Figure 3.6: Distance matrix (left) and representation of protein pairs with significantly lower Sinkhorn distance after statistical testing (right) for the comparison of proteins in the phage proteomes of Enterobacteria phage T7 and Erwinia phage vB_EamP-L1.

The left panel of the figure shows the distance matrix resulting from the comparison of the proteome of Enterobacteria phage T7 and Erwinia phage vB_EamP-L1. The right panel of the figure shows a binary representation of the protein pairs with significantly lower Sinkhorn distance after statistical testing and applying cutoff for effect size. These significant protein pairs were given a value of one, while non-significant protein pairs were given a value of zero.

ducted. For protein 7, BLASTP results showed 99.2% identity with proteins of different *Yersinia* phages, with an E-value of 8e-96. These proteins were described either as uncharacterized or as 'host range protein'. Protein 1.8 is 100% identical to some proteins of several *Yersinia* phages with an E-value of 2.5e-33. However, the function of all of these proteins is uncharacterized. It is surprising to see some proteins of Enterobacteria phage T7 match with proteins of *Yersinia* phages, definitively as one of those is described as important for host range. As Enterobacteria phage T7 and phages of *Yersinia* clearly infect other bacterial hosts, it is unclear whether these proteins are important for phage host specificity. On the other hand, one protein will most often not be the single determinant of phage host range. Therefore, it is possible for this protein to be important in phage infectivity for both Enterobacteria phage T7 as well as phages of *Yersinia*, while the combination with other proteins unique in each of the phages determines the unique phage host range of both. It would be interesting to compare Enterobacteria phage T7 to *Yersinia* phages using optimal transport. However, a lot of the proteins of these phages are uncharacterized or poorly characterized. Therefore, this comparison was not further focused on. Protein 2.8 matched with an uncharacterized protein of Erwinia phage FE44 with an identity of 88.5% and an E-value of 3e-84. It also matched with an HNH endonuclease from Salmonella phage BP12A with an identity of 74.1% and an E-value of 9.8e-73. Potentially, protein 2.8 recycles bacterial DNA by cleaving it, thus making it available for reuse in

Table 3.2: Unique proteins found between the proteomes of Enterobacteria phage T7 and Erwinia phage vB_EamP-L1 after comparison of both proteomes using optimal transport.

(a) Enterobacteria phage T7

Protein ID	Protein name	Biological function
P03776	Gene product 0.4	Inhibition of host cell division
P03748	Tail fiber protein Gp17	Viral attachment to host cell
P03775	Classical restr. Gp0.3	Prevents degradation of T7 DNA by the host
P03782	Protein 4.1	DNA primase and helicase functionalities
P03750	Protein 7	Important for host range[a]
P03794	Protein 1.8	Unknown
P03795	Protein 2.8	Putative HNH endonuclease[a]
P03792	Protein 1.5	Unknown
P03789	Protein 19.2	Unknown
P03779	Uncharacterized protein 1.1	Unknown

(b) Erwinia phage vB_EamP-L1

Protein ID	Protein name	Biological function
G0YQ83	EPS depolymerase	Depolymerizing extracellular polysaccharide[a]
G0YQ42	Gp0.1	Unknown
G0YQ53	Gp1.65	Unknown
G0YQ68	Gp6.3	Unknown
G0YQ77	Gp13.5	Putative endonuclease[a]
G0YQ44	G0.6	Unknown
G0YQ49	Gp1.07	Unknown

a: inferred via BLASTP.

assemblage of the phage genome. Proteins 1.5, 19.2 and uncharacterized protein 1.1 did not match with any protein with known function and did not have any annotation linked to their function in the NCBI database.

In the unique proteins of Erwinia phage vB_EamP-L1, only one protein had a known function. The EPS depolymerase is an enzyme that depolymerizes extracellular polysaccharide. The enzyme matched with several proteins of other Erwinia phages, with identity scores ranging from 57.2% to 66.6% and all having an E-value of approximately zero. One of these proteins was described as a tail spike protein, while the other proteins were also described as EPS depolymerases. Protein Gp13.5 matched with endonucleases from several other phages with identity scores ranging from 40.2% to 48.3% and E-values of 1.6e-15 to 1.7e-18. Furthermore, protein Gp1.07 only matched with an inhibitor protein of Escherichia phage JSS1 with an identity score of 42.2% and an E-value of 4.6e-13. All other matches to protein Gp1.07 did not have any specified function. Finally, proteins Gp0.1, Gp1.65, Gp6.3 and Gp0.6 did not have a specified function and did not match with any protein with known function after performing BLASTP. Additionally, these proteins were not annotated with any function in the NCBI database.

3.5.2 Identification of unique proteins in the comparison between Enterobacteria phage T7 and Salmonella phage Vi06

Figure 3.7 again shows the histogram of effect sizes (for every protein pair), this time for the comparison between Enterobacteria phage T7 and Salmonella phage Vi06. Here, based on visual inspection, the cut-off value for biological relevance was chosen to be -4, which clearly separates extreme effect sizes from the rest. The colors on the histogram indicate the discimination between biologically relevant and non-relevant protein pairs. Every biologically relevant protein pair had a *p*-value of approximately zero. The matrices with *p*-values and effect sizes for this comparison are given in digital appendix B.

The left of Figure 3.8 displays the distance matrix resulting from the comparison of Enterobacteria phage T7 with Salmonella phage Vi06. The right side of the figure again represents the protein pairs that were labeled as significant after statistical testing and applying a cutoff for effect size. In the Enterobacteria phage T7 proteome, 39 of 57 proteins had a significant match in the proteome of Salmonella phage Vi06. In the Salmonella phage Vi06 proteome, 40 of 47 proteins had a significant match in the Enterobacteria phage T7 proteome. The proteins that are unique to one of both proteomes are listed in Table 3.3.

Nine out of 18 unique proteins found in the proteome of Enterobacteria phage T7 for this comparison were not identified in the earlier comparison with Erwinia phage vB_EamP-L1. The DNA-directed DNA polymerase is an enzyme that replicates viral genomic DNA (Tabor and Richardson, 1989). Protein kinase 0.7 modulates the hosts metabolism to favor the virus replication cycle. It works with protein Gp2 to shut off host transcription (Zillig *et al.*, 1975). Protein 4.7 provides helicase and primase functionalities necessary for DNA replication (Mendelman *et al.*, 1992). After running BLASTP, results show significant matches between protein 7.7 and head-to-tail joining proteins from Yersinia phage R and Yersinia phage Y. Both alignments had an identity score of 99.2% with an E-value of 3.2e-91. Terminase is an enzyme necessary for the translocation of viral DNA into empty capsids. The enzyme also possesses an endonuclease activity to cut the translocated DNA strands at specific positions in order to translocate exactly the same viral DNA in every empty capsid (Daudén *et al.*, 2013). Furthermore, protein 5.3 was annotated with GO:004518 which describes endonuclease activity. The GO was inferred from electronic annotation[1] through InterPro (Finn *et al.*, 2017). BLASTP results however did not reveal a match with any

[1]Electronic annotation is an automated method of annotation, without curatorial judgement, see http://geneontology.org/page/guide-go-evidence-codes

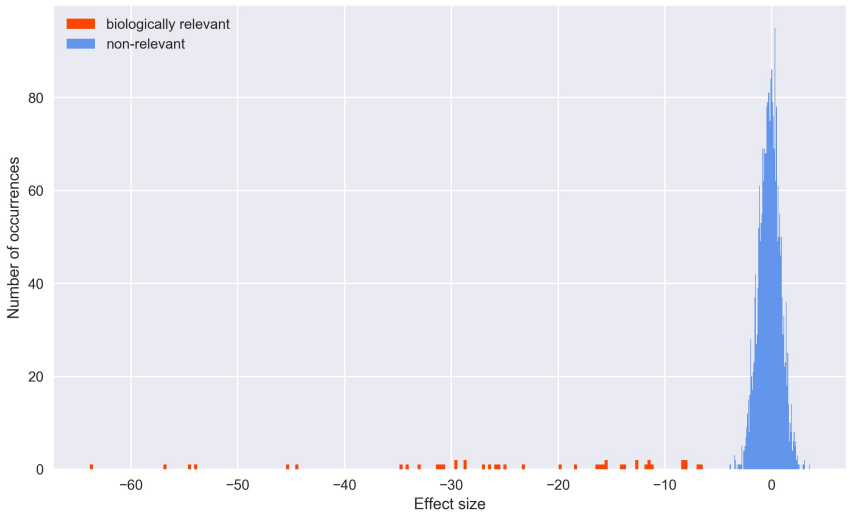


Figure 3.7: Histogram of the effect size of every protein pair resulting from the comparison of Enterobacteria phage T7 and Salmonella phage Vi06.

The figure shows the effect sizes for every pairwise comparison of the proteins of phages Enterobacteria phage T7 and Salmonella phage Vi06. Protein pairs were labeled as biologically relevant if their effect size was smaller than -4. Every biologically relevant protein pair had a *p*-value of approximately zero.

protein with known function. As such, the GO referring to nuclease activity should be interpreted with caution. No match was found for proteins 0.5, 4.2, 19.3 and 0.6B. Furthermore, these proteins did not have any function annotated to their record in the NCBI database. The eight other unique proteins found in Enterobacteria phage T7 for this comparison were already mentioned in the previous comparison.

For the Salmonella phage Vi06 proteome, 7 proteins were identified as being unique. This is a lot less than the number of proteins identified as unique in the Enterobacteria phage T7 proteome. However, Salmonella phage Vi06 possesses fewer proteins than Enterobacteria phage T7, so it is expected to find more unique proteins in Enterobacteria phage T7. BLASTP analysis matched protein E1XUA1 matched with a putative anti-sigma factor from Synechococcus phage S-CBS4 (identity score of 37.6%, E-value of 2.9e-10). It also matched with a putative HNH endonuclease from Pseudomonas phage ventosus (identity score of 34.0%, E-value of 4.9e-10). However, a lot of other matches were uncharacterized proteins. Therefore, based on BLASTP results alone, it is unclear what the function of protein E1XUA1 could be. Surprisingly, BLASTP results show a match between protein E1XU98 and protein 4.2 of Enterobacteria phage T7, which is also indicated as unique for Enterobacteria phage T7 in Table 3.3. The alignment has an identity score of 48.5% and an E-value of 6.5e-7. This false result could be a consequence of the chosen cut-off value for effect size. When comparing Fig-

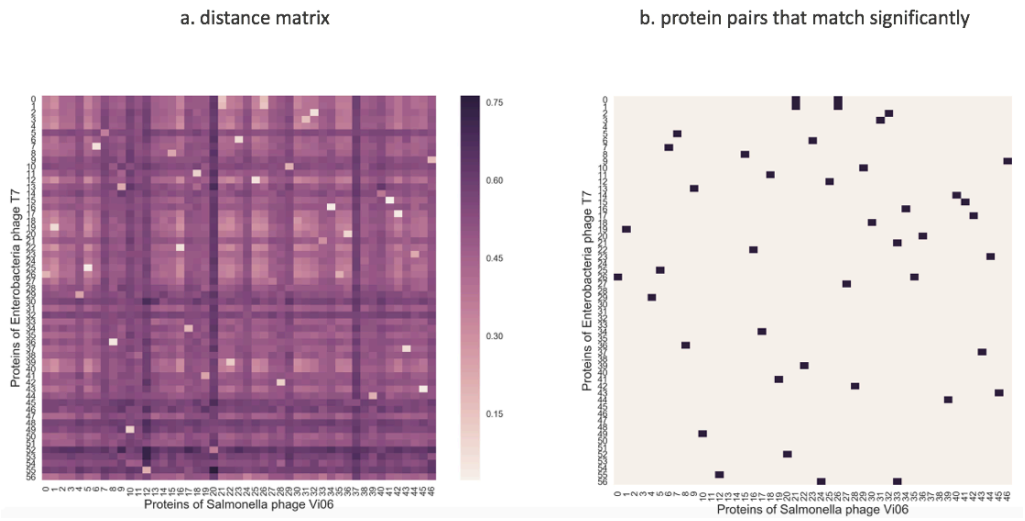


Figure 3.8: Distance matrix (left) and representation of protein pairs with significantly lower Sinkhorn distance after statistical testing (right) for the comparison of proteins in the phage proteomes of Enterobacteria phage T7 and Salmonella phage Vi06.

The left panel of the figure shows the distance matrix resulting from the comparison of the proteome of Enterobacteria phage T7 and Salmonella phage Vi06. The right panel of the figure shows a binary representation of the protein pairs with significantly lower Sinkhorn distance after statistical testing and applying cutoff for effect size. These significant protein pairs were given a value of 1, while non-significant protein pairs were given a value of 0.

ure 3.5 with Figure 3.7, the cut-off value for effect size might be too strict for this comparison. Choosing a less negative cut-off value for effect size would potentially identify the protein pair E1XU98 protein 4.2 as significant match. Both proteins did not have a specified biological function. Protein E1XUC6 is annotated with GO:004519, which describes endonuclease activity (inferred through electronic annotation). After performing BLASTP, the protein matches with several proteins with unknown functions, as well as an HNH homing endonuclease from Pectobacterium phage PP74 (identity score of 51.4%, E-value of 1.1e-45). Endonuclease activity could be responsible for cutting bacterial DNA, which frees up resources for phage replication. It could also be responsible for correct packaging of the phage DNA inside the virion or site-specific recombination (although this last activity is better described for integrases). Protein E1XU90 matches with protein 1.8 of Enterobacteria phage T7 (identity score of 47.1%, E-value of 3.8e-2). Additionally, protein E1XU82 matches with protein kinase 0.7 of Enterobacteria phage T7 (identity score of 47.6%, E-value of 1.6e-23). Again, these matches indicate a false result. The other proteins it matched with did not have any specified function. Proteins E1XU81 did not match with any other protein with known function. As most of the unique proteins for Salmonella phage Vi06 in this comparison do not have a specified function, it is difficult to link these proteins to phage host specificity. Additionally, Salmonella phage Vi06 is known to have a tail fiber protein, which could not be identified as being unique for its proteome. BLASTP

Table 3.3: Unique proteins found between the proteomes of Enterobacteria phage T7 and Salmonella phage Vi06 after comparison of both proteomes using optimal transport.

(a) Enterobacteria phage T7

Protein ID	Protein name	Biological function
P00581	DNA-directed DNA polymerase	Replicates viral genomic DNA
P00513	Protein kinase 0.7	Modulate host metabolism
P03775	Classical restr. gp0.3	Prevents degradation of T7 DNA by the host
P03777	Protein 0.5	Unknown
P03797	Protein 3.8	Putative HNH endonuclease[a]
P03782	Protein 4.1	DNA primase and helicase functionalities
P03796	Protein 7.7	Putative head-to-tail joining protein[a]
P03786	Protein 4.7	DNA primase and helicase functionalities
P03783	Protein 4.2	Unknown
P03694	Terminase, large subunit gp19	Viral DNA translocation
P03794	Protein 1.8	Unknown
P03790	Protein 19.3	Unknown
P03795	Protein 2.8	Putative HNH endonuclease[a]
P03799	Protein 6.3	Unknown
P03798	Protein 5.3	Nuclease activity[b]
P03778	Protein 0.6B	Unknown
P03791	Protein 1.4	Unknown
P03789	Protein 19.2	Unknown

(b) Salmonella phage Vi06

Protein ID	Protein name	Biological function
E1XUC5	Uncharacterized protein	Unknown
E1XUA1	Hypothetical phage protein (fragment)	Unknown
E1XU98	Hypothetical phage protein	Matches protein 4.2 of T7 phage[a]
E1XUC6	Predicted endonuclease	Endonuclease activity[b]
E1XU81	Uncharacterized protein	Unknown
E1XU90	Conserved hypothetical phage protein	Matches protein 1.8 of T7 phage[a]
E1XU82	Uncharacterized protein	Matches protein 0.7 of T7 phage[a]

a: inferred via BLASTP; b: inferred from electronic annotation through InterPro.

results reveal that the predicted tail fiber protein of Salmonella phage Vi06 indeed matched with the tail fiber protein Gp 17 of Enterobacteria phage T7 (identity score of 83.8%, E-value of 3e-69). However, the alignment only extended until the 148th residue of both proteins, while the tail fiber protein of Salmonella phage Vi06 is 657 AAs long and the tail fiber protein of Enterobacteria phage T7 is 553 AAs long. This could indicate that both tail fiber proteins share a common N-terminal domain, but still employ different functionalities related to the C-terminal domain(s). Indeed, a study by Steven *et al*. (1988) suggests conservation of the N-terminal domain among tail fiber proteins of different phages of the T7 group (Steven*et al.*, 1988). The C-terminal domain of the tail fiber protein from Enterobacteria phage T7 is known to bind to the cell-surface lipopolysaccharide receptor (Scholl *et al.*, 2014). On the other hand, the C-terminal domain of the tail fiber from Salmonella phage Vi06 is probably the key

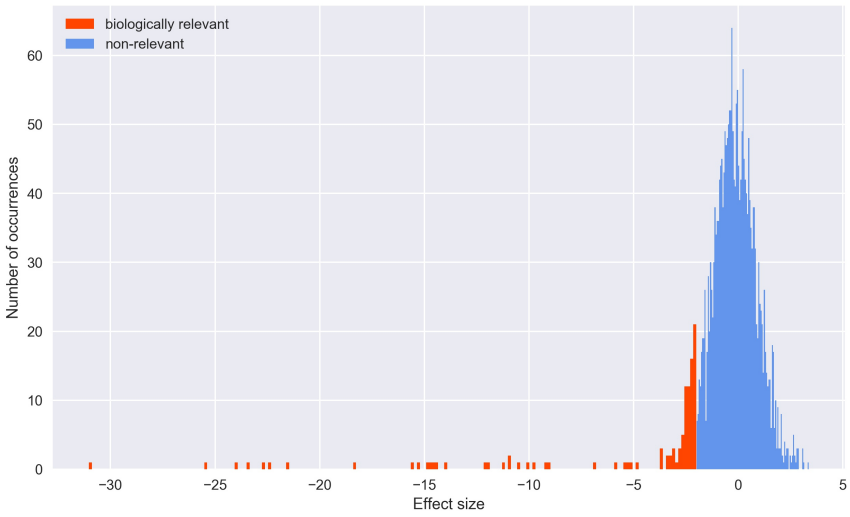


Figure 3.9: Histogram of the effect size for every protein pair resulting from the comparison of Erwinia phage vB_EamP-L1 and Salmonella phage Vi06.

The figure shows the effect sizes for every pairwise comparison of the proteins of phages Erwinia phage vB_EamP-L1 and Salmonella phage Vi06. Protein pairs were labeled as biologically relevant if their effect size was smaller than -2. Every biologically relevant protein pair had a *p*-value of approximately zero.

domain in recognition of the Vi capsular antigen (Park *et al.*, 2012). The difference in the C-terminal domain(s) could thus at least partially explain the difference in host specificity.

3.5.3 Identification of unique proteins in the comparison between Erwinia phage vB_EamP-L1 and Salmonella phage Vi06

Figure 3.9 again shows the histogram of effect sizes (for every protein pair), this time for the comparison between Erwinia phage vB_EamP-L1 and Salmonella phage Vi06. The cut-off value for biological relevance was chosen to be -2, as a better cut-off value could not be observed visually. The colors on the histogram again indicate the discimination between biologically relevant and non-relevant protein pairs. Every biologically relevant protein pair had a *p*-value of approximately zero. The matrices with *p*-values and effect sizes for this comparison are given in digital appendix B.

The left of Figure 3.10 displays the distance matrix resulting from the comparison of Erwinia phage vB_EamP-L1 with Salmonella phage Vi06. The right side of the figure again represents the protein pairs that were labeled as significant after statistical

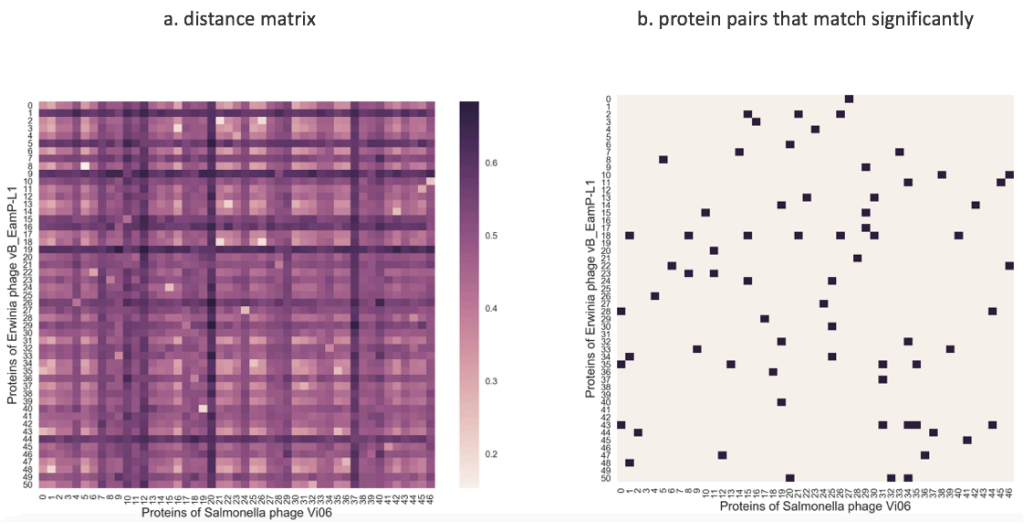


Figure 3.10: Distance matrix (left) and representation of protein pairs with significantly lower Sinkhorn distance after statistical testing (right) for the comparison of proteins in the phage proteomes of Erwinia phage vB_EamP-L1 and Salmonella phage Vi06.

The left panel of the figure shows the distance matrix resulting from the comparison of the proteome of Erwinia phage vB_EamP-L1 and Salmonella phage Vi06. The right panel of the figure shows a binary representation of the protein pairs with significantly lower Sinkhorn distance after statistical testing and applying cutoff for effect size. These significant protein pairs were given a value of 1, while non-significant protein pairs were given a value of 0.

testing and applying a cutoff for effect size. In the Erwinia phage vB_EamP-L1 proteome, 38 of 51 proteins have a significant match in the proteome of Erwinia phage vB_EamP-L1. Likewise, 44 of 47 proteins in the Salmonella phage Vi06 proteome have a significant match in the Erwinia phage vB_EamP-L1 proteome. The proteins that are unique to one of both proteomes are listed in Table 3.4.

In the proteome of Erwinia phage vB_EamP-L1, thirteen proteins were identified as unique in this comparison. Protein G0YQ46 was annotated with GO:0016301, which describes kinase activity. BLASTP analysis also revealed matches to various protein kinases of other phages (identity scores between 44.0% and 57.1%, E-values between 9.9e-40 and 6.6e-45). The identified HNH endonuclease was annotated with GO:0004519, describing endonuclease activity. Proteins 0.65, 19.2, 5.8 and 1.6 did not significantly match with any protein with known function. No proper annotation of these proteins was found in the NCBI database as well. The other identified proteins were already discussed in a previous comparison.

In the proteome of Salmonella phage Vi06, only three proteins were identified as unique in the comparison to the proteome of Erwinia phage vB_EamP-L1. The first protein was E1XUA1, which was already identified in the last comparison, and has an unknown function. Protein E1XU80 has no specified function but BLASTP results revealed a match to Gp0.4 from Enterobacteria phage T7 with an identity score of 71.8%

Table 3.4: Unique proteins found between the proteomes of Erwinia phage vB_EamP-L1 and Salmonella phage Vi06 after comparison of both proteomes using optimal transport.

(a) Erwinia phage vB_EamP-L1

Protein ID	Protein name	Biological function
G0YQ43	Gp0.2	Unknown
G0YQ45	Gp0.65	Unknown
G0YQ53	Gp1.65	Unknown
G0YQ68	Gp6.3	Unknown
G0YQ90	Gp19.2	Unknown
G0YQ77	Gp13.5	Putative endonuclease[a]
G0YQ63	DNA-directed DNA polymerase	Unknown
G0YQ48	Gp1.05	Unknown
G0YQ46	Protein kinase	Kinase activity (GO[b])
G0YQ66	Gp5.8	Unknown
G0YQ51	HNH endonuclease	Endonuclease activity (GO[b])
G0YQ52	G1.6	Unknown
G0YQ49	Gp1.07	Unknown

(b) Salmonella phage Vi06

Protein ID	Protein name	Biological function
E1XUA1	Hypothetical phage protein (fragment)	Unknown
E1XU80	Conserved hypothetical phage protein	Matches gp 0.4 of T7 phage[a]
E1XUA8	Host specificity protein A	Matches protein 7 of T7 phage[a]

a: inferred via BLASTP; b: inferred from electronic annotation through InterPro.

and E-value of 6.2e-13. As previously stated, Gp0.4 has a role in the inhibition of cell host division, which results in more resources being available for phage replication. Finally, protein E1XUA8 was only annotated as 'host specificity protein A' (mentioned as predicted protein on UniProt). After performing BLASTP, protein E1XUA8 was found to match with protein 7 of Enterobacteria phage T7 (identity score of 92.5%, E-value of 4.7e-90). Protein 7 is only known to be important for host specificity as well (previously inferred via BLASTP). Again, the predicted tail fiber protein of Salmonella phage Vi06 was not identified as unique. Likewise, the EPS depolymerase of Erwinia phage vB_EamP-L1 was not identified as a unique protein. BLASTP results also reveal a significant alignment spanning from residue 7 to residue 181 of the predicted tail fiber protein of Salmonella phage Vi06. The identity score for this alignment was 32.4%, the E-value was 3.6e-10. Although the identity score is less compared to the identity score between the tail fiber proteins of Salmonella phage Vi06 and Enterobacteria phage T7, there is again similarity in the N-terminal domains of both phages.

3.5.4 Discussion of the comparisons between Enterobacteria phage T7, Erwinia phage vB_EamP-L1 and Salmonella phage Vi06

It was previously hypothesized that choosing a low value of k would not have an impact on the results after statistical testing with the same low value of k. However, after observing the results, this statement could not be validated. Some proteins significantly match with several other proteins, even after applying a cut-off value for effect sizes. This is especially clear in the comparison between Erwinia phage vB_EamP-L1 and Salmonella phage Vi06 where protein 18 from Erwinia phage vB_EamP-L1 matches significantly with seven proteins from Salmonella phage Vi06. This can be seen in the right panel of Figure 3.10. This is not expected from a biological perspective. In these comparisons, it was either expected for a protein to match significantly with zero or only one protein (at least in comparison to matches with the other proteins it was compared to). One possible explanation is that gene duplication occurred in the genome of Salmonella phage Vi06. However, this number of gene duplications seems unlikely. It would be interesting to apply optimal transport to compare the proteins of Salmonella phage Vi06 with themselves to see whether gene duplication occurred. As a result, very few proteins of Salmonella phage Vi06 were identified as unique, while considerably more proteins were found as unique in the proteome of Erwinia phage vB_EamP-L1. In the other comparisons, the number of identified unique proteins was also not equal in the respective proteomes. Whether only the value of k or the combination of k and the value for the cut-off value for effect sizes had an impact on this result is unclear. Further comparisons with different values of k and cut-off value for effect sizes should be conducted to clarify this.

It is also clear that a lack of annotation of protein data hampers the complete interpretation of the results. A considerable number of unique proteins had no specified function. Therefore, it was not possible to interpret their possible role in host specificity. Some of the proteins that were identified as unique to a particular proteome did exhibit functions related to factors that are known to be important in defining a phage's host range. However, some proteins were identified as unique while BLASTP results revealed it matched with a protein in the other proteome that was also identified as unique. These false results indicate the importance of manual interpretation of the obtained results as well as the advantage of using complementary tools for interpretation such as BLAST. False results could be avoided by using a less strict cut-off for effect size. However, this could also lead to unique proteins not being identified as such because of low similarities to other proteins.

It was expected for tail fiber or related proteins to be identified as unique among the different proteomes, because of their essential role in phage adsorption. However, tail fiber proteins were only identified among the unique proteins in one comparison. On the other hand, BLASTP analysis also identified these proteins as matching. More specifically, these proteins share similarity in their N-terminal domains. These N-terminal domains functions as anchors that attach the tail fiber to the baseplate of the phage (Latka *et al.*, 2017). Alignments of both tail fiber proteins of Enterobacteria phage T7 and Salmonella phage Vi06 and the tail fiber protein of Salmonella phage Vi06 with the EPS depolymerase of Erwinia phage vB_EamP-L1 are given in Figure A.3 of appendix A. It is the C-terminal domain(s) that is unique in these proteins, and they are responsible for differences in the attachment process of the phages to their respective hosts. Because of the similarity in the N-terminal domain, the method used here was not able to identify these proteins as unique. It is speculated that a higher value of k, possibly together with a stricter cut-off for effect size, would result in the identification of these tail fiber proteins and EPS depolymerase as unique proteins. Furthermore, in the analyses above only three phages of the T7 virus group were investigated. When this comparison would be extended to multiple or all members of the T7 group that have a different bacterial host, it is likely that a lot less proteins would be identified as unique if proteins were only identified as such when having no similarity to any protein in phage proteomes related to different hosts. Possibly, the C-terminal domain(s) of tail fiber proteins could then be identified more consistently. However, these speculations were not investigated further.

Additionally, it would be interesting to investigate these tail fiber proteins further to characterize the C-terminal domains in which they differ. Optimal transport could be used to identify the k-mers that differ the most between these proteins. Repeating this comparison for multiple values of k could possibly identify useful protein features that are responsible for the difference in host specificity. Likewise, protein 7 from Enterobacteria phage T7 and protein E1XUA8 from Salmonella phage Vi06 could be investigated further to characterize the similarities and differences between these proteins.

3.6 Conclusion

In conclusion, Chapter three shows the use of optimal transport to find unique proteins among three related phages of the T7 virus group. This chapter shows that optimal transport can be used to search for similarities and differences among proteomes and proteins in a comparative way, by representing these proteomes or proteins as probability distributions of k-mers. Moreover, both the results of Chapters

two and three show that this method is appropriate to study biological systems to discover new knowledge at the level of proteomes and proteins. However, it is also clear that appropriate values for parameters as k and the cut-off for effect size are not straightforward to choose. Because optimal transport compares probability distributions using the cost matrix M, exactly matching as well as similar k-mers among proteins or proteomes influence the resulting Sinkhorn distance. Because of this, optimal transport computes similarity less strict compared to alignment. This could bias the perception of similarity, especially when k is small, as discussed in Section 2.7. In all cases, critical evaluation of the obtained results is necessary, and complementary tools such as BLAST can aid in this evaluation.

The next chapter specifically focuses on tail fiber and tail spike proteins and studies these proteins in more detail using machine learning techniques.

CHAPTER 4

Machine learning methods to predict phage-host specificity

4.1 Scope of this chapter

In Section 3.3, tail fiber and tail spike proteins were discussed as an important determinant of phage-host specificity. Therefore, these proteins will be used here to further study host specificity using a machine learning approach. There are two main objectives in this chapter. The first objective is to be able to correctly classify tail fiber and tail spike proteins to their bacterial host. If this proves to be successful, the second objective is to identify the protein characteristics that are most important for this classification. In this way, the methods below could prove useful in predicting bacteria-phage interactions, as well as in increasing the understanding of phage-host specificity at the level of the specific proteins. Below, machine learning is briefly introduced and work is described that has already been done in the area of predicting bacteria-phage interactions using machine learning.

Machine learning is the field of research that involves the development of algorithms that extract patterns from data (Bishop, 2006). An important problem studied in machine learning is classification, which involves predicting a discrete or qualitative response for each observation in a dataset (James *et al.*, 2013). The central goal here is to classify observations into one of K discrete categories C_k where $k = 1, \ldots, K$ (Bishop, 2006). In predicting bacteria-phage interactions, the goal then becomes to predict a phage's host or a bacterium-phage interaction based on information of the phage and/or the bacterium. This information is represented as so-called features, from which the algorithms learn patterns in a process called training.

In a recent study, Leite *et al.* (2017) used machine learning to predict bacteria-phage interactions based on protein data. The authors used protein-protein interaction scores (i.e. sums of protein-domain interaction scores from the DOMINE database) between proteins of both bacteria and their phages to predict whether or not a particular bacterium-phage pair could interact. This is an example of a binary classification

problem, where the outcome of prediction is either interaction or no interaction. Postive interactions (i.e. bacteria and phages that are known to interact) were collected from PhagesDB and Genbank databases. Negative interactions were created by randomly selecting bacteria and phages in the positive dataset to form pairs that were not present in the positive dataset and of which the bacteria belonged to another species as the phage's known host. In total, a dataset containing 1065 positive and equal number of negative interactions was constructed. Afterwards, the CDSs and related proteins were gathered from the genomes of the bacteria and phages to construct features. In addition to using protein-protein interaction scores, the authors also included amino acid frequency, chemical composition and molecular weight of the proteins as informative features. Their final predictive models reached accuracy, sensitivity and specificity values of over 90%. The authors do mention several limitations in the followed approach. Firstly, the diversity of the dataset was limited due to the fact that most interactions involved only a single bacterial species, *Mycobacterium smegmatis*. Secondly, the predictions of interaction were made at the level of phage species, while a large number of phages exhibit host specificity at the strain level. As a predictive model only learns through use of data, this illustrates the importance of sufficient quality of the dataset, as well as the nuances to make when datasets are questionable in their quality. Nonetheless, machine learning approaches to predict bacteria-phage interaction *in silico* still have value in increasing our understanding of phage-host specificity (Leite *et al.*, 2017).

In this chapter, a slightly different method is used to study phage-host specificity. Machine learning models are used to predict the bacterial host of phages based on features specifically extracted from tail fiber and tail spike protein data. The machine learning models in this work attempt to classify proteins in three different categories, representing the host of their phage: *Escherichia coli*, *Salmonella enterica* or *Klebsiella pneumoniae*. From the coding and protein sequences of these tail fiber and tail spike proteins, features are extracted which represent typical DNA- and protein characteristics. These features are then used as input for the predictive models. In addition, after classification itself, the features that are most important during classification are identified and further interpreted. This method is explained in detail in the section below, after which the obtained results are interpreted and discussed.

4.2 Use of tail fiber and tail spike protein data to infer phage-host specificity

4.2.1 Tail fiber and tail spike protein data acquisition

Tail fiber and tail spike protein data was gathered from UniProt KB (The UniProt Consortium, 2017). To start with, several keywords and Gene Ontology (GO) annotations were identified that could indicate tail fiber or tail spike proteins. An overview of these is given in Table 4.1. However, not all keywords and GO annotations were eventually used in constructing the final dataset. Proteins annotated with GO:0098004 were discarded because proteins related to the assembly of the tail fiber were not of interest. Additionally, the GO:0046718 annotation completely overlapped with other GO annotations. However, this GO might still be relevant if new protein entries are added to Uniprot in the future. The other keywords and GO annotations were used to manually query UniProt. Queries were restricted to the Caudovirales group of viruses to enhance the relevance of the results. Tail fiber and tail spike proteins are only expected in tailed phages. On the other hand, to further enlarge the protein dataset resulting from these queries, the results were clustered together with other proteins having at least 90% sequence identity by mapping the results from UniProt to UniRef (Suzek *et al.*, 2015). These clusters, now containing the original protein sequences plus new sequences, were mapped back to UniProt to be able to download the protein data. In this way, new proteins were found that were not annotated to any of the gene ontologies or keywords that were searched for. Additionally, several tail fiber and tail spike proteins related to *K. pneumoniae* were found in literature and manually searched for on UniProt (Latka *et al.*, 2017). After manually filtering out unwanted proteins that were not of interest, a total of 425 protein sequences with unique identifiers were downloaded from UniProt.

Several proteins did not have a specified bacterial host related to them. In a third step, these missing hosts were added by searching for the phages (which these proteins originate from) in the GenomeNet virus-host database, in the NCBI GenBank database or in literature (Benson *et al.*, 2013; Mihara *et al.*, 2016). Subsequently, the corresponding phages and their bacterial hosts were added to the dataset for each protein. Bacterial hosts were only characterized at the species level due to lack of sufficient strain information about the hosts. Afterwards, CDSs of the proteins were added to the dataset by using the EMBL IDs of the entries and querying the GenBank database from within Python using BioPython functionalities (Cock *et al.*, 2009). Finally, a manual check of the dataset was performed to delete entries with undeter-

Table 4.1: Keywords and Gene Ontologies that could be useful in the construction of a tail fiber and tail spike protein dataset.

Keyword or Gene Ontology	Description
KW-1161	Viral attachment to host cell
KW-1230	Viral tail fiber protein
GO:0019062	Virion attachment to host cell
GO:0046718	Viral entry into host cell
GO:0098004	Virus tail fiber assembly
GO:0098024	Virus tail, fiber
GO:0008233	Peptidase activity
GO:0008236	Serine-type peptidase activity
GO:0016798	Hydrolase activity, acting on glycosyl bonds
GO:0004553	Hydrolase activity, hydrolyzing O-glycosyl compounds

Table 4.2: Brief overview of the constructed tail fiber and tail spike protein dataset.

Dataset descriptor	Example entry
Protein ID	Q04830
Taxonomic ID	344021
EMBL ID	AJ505988
Protein name	Tail spike protein
Protein sequence	MSTITQFPSGNTQYRIEFDYLARTFVVVTLVNSSNPTLNRVLEVGR...
Coding sequence	ATGTCCACGATTACACAATTCCCTTCAGGAAACACTCAGTACAG...
Organism name	Bacteriophage K1F
Host name	Escherichia coli

mined amino acids (represented as the letter X) in their protein sequence and entries that either did not have a reliable source for their CDS (e.g. supposedly, a tail fiber protein was present in Drosophila species) or that had DNA sequences that did not adequately match the protein sequence. The final dataset consisted of 411 entries from various phages. A brief overview of the dataset is given in Table 4.2. The entire dataset is given in a digital appendix C.

Finally, the dataset was explored to get a better sense of the diversity of the dataset. A bar plot was generated to show the number of instances assigned to different bacterial hosts. The three largest groups of hosts in this dataset comprised *Escherichia coli*, *Salmonella enterica* and *Klebsiella pneumoniae*. As these three bacteria are also important human pathogens, the analyses below focused solely on the proteins corresponding to these bacteria (Giske *et al.*, 2008; Fabrega and Vila, 2013). The proteins corresponding to other bacterial hosts were filtered out of the dataset. The remaining dataset consisted of 330 entries. In this way, a multi-class classification could be performed with three classes.

4.2.2 Machine learning methods

The method used to construct machine learning models was implemented in Python. More specifically, machine learning models were constructed using the Scikit-learn package in Python (Pedregosa *et al.*, 2011). The Python script is given in digital appendix C.

After collection of relevant data, the second step was the construction of several features that characterize the DNA and protein sequences in the dataset. An overview of the different features is given in Table 4.3. In total, 133 features were constructed based on the raw DNA sequences. These features included nucleotide frequencies, GC-content, codon frequencies and codon usage bias. Codon usage bias was computed by counting the occurence for each codon and subsequently dividing by the total number of counts from synonymous codons (i.e. codons that correspond to the same amino acid). Furthermore, 38 features were constructed based on the primary protein sequence. More specifically, 20 features described the relative abundance of amino acids. Fifteen more features desribed various physicochemical properties of the sequences including molecular weight, iso-electric point, aromaticity and others. Finally, three features described the secondary structure in terms of the fractions of amino acids that are predicted to be present in an α-helix, β-sheet or turn. All together, every protein entry in the dataset is now described by 171 features.

To explore the features and the classes they are related to, a principal component analysis (PCA) was performed (Bishop, 2006). This is a dimensionality reduction technique that allows to visualize high-dimensional data by projecting the data orthogonally in a lower-dimensional linear space that maximizes the variance of the projected data. To avoid biases due to differences in scale, all features were standardized befor the analysis. Afterwards, the first three components were plotted in two-dimensional spaces to get a sense of how well all three classes were separable in a low number of dimensions. Additionally, linear discriminant analysis (LDA) was used to compute the linear discriminants on the standardized set of features. This can be seen a supervised (i.e. using label information) dimensionality reduction technique to explore how well data is separable in a low number of dimensions by plotting the linear discriminants. The biggest difference between PCA and LDA is that LDA uses the different classes to compute the linear discriminants, while PCA does not use class label information (i.e. unsupervised) in computing the principal components. As a final exploratory analysis, local pairwise sequence alignment was performed with every protein sequence in the dataset. The resulting pairwise alignment scores were clustered and visualized in Python. This provided an extra look at the raw protein sequences and how they are related.

Table 4.3: Overview of the different features used for classification of the tail fiber and tail spike proteins.

(a) Features derived from DNA sequence.

Description	# of features	Reference
Nucleotide frequency	4	/
GC-content	1	Zhou and Liu, 2008[a]
Codon frequency	64	Sastry *et al.*, 2017
Codon usage bias	64	Roux *et al.*, 2015

(b) Features derived from protein sequence.

Description	# of features	Reference
AA frequency	20	Al-Shahib *et al.*, 2007
Molecular weight	1	Al-Shahib *et al.*, 2007[a]
Protein length	1	/
Iso-electric point	1	Hobohm and Sander, 1997[a]
Aromaticity	1	Lobry and Gautier, 1994[a]
Instability	1	Guruprasad *et al.*, 1990[a]
Flexibility	1	Vihinen, 1994[a]
Aliphatic AA fraction	1	Sastry *et al.*, 2017
Uncharged polar AA fraction	1	Sastry *et al.*, 2017
Polar AA fraction	1	Sastry *et al.*, 2017
Hydrophobic AA fraction	1	Sastry *et al.*, 2017
Positively charged AA fraction	1	Sastry *et al.*, 2017
Negatively charged AA fraction	1	Sastry *et al.*, 2017
Sulfur containing AA fraction	1	Sastry *et al.*, 2017
Amide containing AA fraction	1	Sastry *et al.*, 2017
Alcohol containing AA fraction	1	Sastry *et al.*, 2017
Fraction of AAs in α-helix	1	Sastry *et al.*, 2017[a]
Fraction of AAs in β-sheet	1	Sastry *et al.*, 2017[a]
Fraction of AAs in turn	1	Sastry *et al.*, 2017[a]

a: calculated via Biopython ProteinAnalysis utilities.

After the features were constructed and explored, several machine learning models were selected to learn from the tail fiber and tail spike protein data. Both linear and non-linear models were adopted as it was not known beforehand whether the different classes could be separated by a linear boundary. Two linear models were selected: logistic regression and LDA. Logistic regression was trained using L_1 and L_2 regularization. These linear methods were chosen for their general applicability in classification as well as their ease of interpretation. Furthermore, the two adopted non-linear models were Random Forests (RFs) and Gradient Boosting (GB). These non-linear methods can model more complex patterns in the dataset, which can possibly result in more accurate predictions.

Subsequently, these models were trained using the available protein data. However, all methods except for LDA have hyperparameters that can be adjusted (or tuned) for optimal performance of the models. Training, tuning and testing of the different models was done using nested four-fold cross-validation. A visual representation of

this is given in Figure A.4 of appendix A. Four-fold cross-validation implies splitting the dataset in four equal parts. Training and tuning is done using three parts of the dataset, while the fourth part is kept for testing (i.e. to measure performance). This can be repeated four times, each time measuring performance using a different part of the dataset. Here, the cross-validation was also nested, consisting of an inner loop and outer loop. In the outer loop, the data is split into four parts as explained above. In the inner loop, the three parts allocated to training and tuning are split up again in four subparts. Here, three of the subparts are used for training, while the fourth subpart is used for tuning. For each split in the outer loop of the cross-validation, this inner loop (the split into subparts) can also be repeated four times. Hence the cross-validation is nested.

For logistic regression, the only hyperparameter that was tuned was the strength of regularization (i.e. the magnitude of shrinkage of the parameter values to prevent overfitting and reduce variance). In RFs, the number of trees in the forest and the number of features to consider when making a split were the two hyperparameters that were tuned. In GB, the number of boosting stages to perform was tuned as a hyperparameter. Every hyperparameter was tuned using accuracy as performance measure. However, in each testing phase, four performance measures were computed: accuracy (Acc), precision (P), recall (R) and the F1-measure. After cross-validation, the performance measures were averaged over the different folds of the cross-validation. Subsequently, the different machine learning models were interpreted based on the obtained performance measures.

As a final analysis, features that were important in the classification were identified using three methods: logistic regression with L_1 regularization, RF and a decision tree. The L_1 regularization applied in logistic regression shrinks the weights of different features, potentially to zero. Applying a strong regularization will shrink most of the weight to zero, which enables to look at the non-zero weights as a measure of feature importance. Because all features were standardized, these coefficients can carefully be interpreted as a measure of feature importance. By increasing the strength of regularization, more and more weights of features are forced to zero. Because of this property of L_1 regularization, only the most important features remain having non-zero coefficients at strong regularization. Additionally, feature importance was measured by an RF model with 2000 trees. In the RF model, feature importance is calculated as the Gini importance (Menze *et al.*, 2009). The coefficients of the logistic regression and the Gini importance for each feature were plotted and compared. Finally, a pruned decision tree was constructed in R using the standardized features. This pruned tree was plotted to visualize the most important features in the tree. The R script is given in digital appendix C.

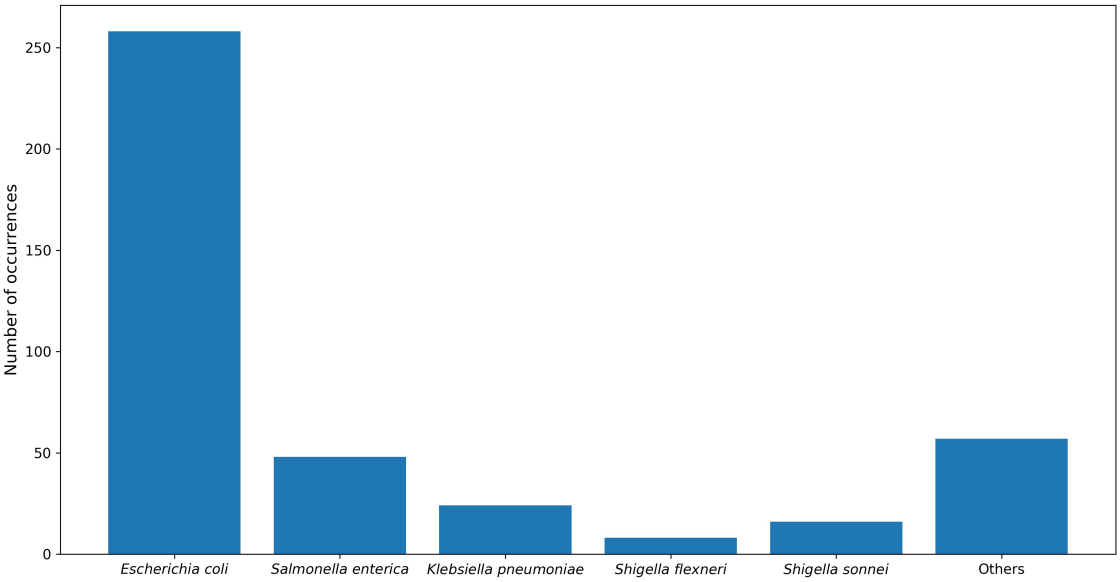


Figure 4.1: Bar plot of different bacterial hosts in the dataset.

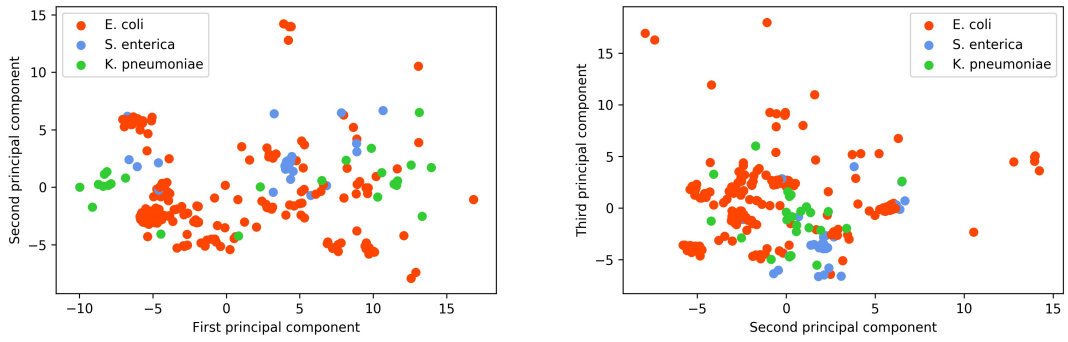
The bar plot shows the number of occurrences of each of the five largest bacterial species linked to the tail fiber and tail spike proteins in the dataset. The occurrences of various other bacterial species were all grouped in 'others'.

4.3 Results and discussion

4.3.1 Data exploration

Figure 4.1 shows the bar plot of the different bacterial hosts related to the tail fiber and tail spike proteins present in the dataset. Clearly, most of the proteins are related to *E. coli* hosts. The four other most abundant hosts are *S. enterica*, *K. pneumoniae*, *Shigella flexneri* and *Shigella sonnei*. Various other bacterial species occur as well, which were grouped in 'others'. As mentioned above, the proteins corresponding to the three most abundant hosts were kept for further analysis and to train the machine learning models on. In this way, the machine learning models were trained to discriminate between three classes, corresponding to three major human pathogens. All other proteins were omitted from the dataset.

Figure 4.2 shows the first three principal components of the data after applying PCA on the entire (standardized) set of features. Plot (a) displays the first and second principal components on the x-axis and y-axis, respectively. Plot (b) displays the second and third principal components on the x-axis and y-axis, respectively. Both



(a) **(b)**

Figure 4.2: Plots of the first, second and third principal components computed using PCA on the entire set of features.

Plot (a) shows the first and second principal components on the x-axis and y-axis, respectively. Plot (b) shows the second and third principal components on the x-axis and y-axis, respectively. In both plots, the proteins were colored accoring to their related host.

plots show some variation among the different classes (hosts), but not enough to make an adequate separation between the classes. The dimensionality reduction obtained by PCA does not result in a proper separation of the different classes. This indicates that the largest sources of variance are not the differences between the classes. One reason why this might be expected is that some features show high correlation among each another. For example, as tryptophan is only encoded by the UGG codon, both these features are perfectly correlated. Including such highly correlated features increases the contribution of the common underlying property to the PCA. This results in the PCA overemphasizing this contribution. Removing these highly correlated features from the data and subsequently repeating the PCA could result in a better separation between the classes.

Figure 4.3 displays the first linear discriminant of LDA on the x-axis and the second linear discriminant on the y-axis. The linear discriminants were computed using the entire set of standardized features. Surprisingly, the linear discriminants of LDA are able to almost perfectly separate the different classes in two dimensions. This is because LDA explicitely models the different classes as separate multivariate Gaussian distributions and computes the linear discriminants in order to maximize the separation between those distributions (James *et al.*, 2013). As a result, the separation between the three classes is almost perfect. However, there are three proteins that correspond to *S. enterica* but appear much closer to proteins related to *E. coli* hosts. These three are all proteins of Salmonella phage SG1. Additionally, no other proteins of Salmonella phage SG1 are present in the dataset. Two of the three proteins are predicted tail fiber proteins, while the other is a predicted tail connector protein. BLASTP results reveal that both predicted tail fiber proteins are remarkably similar to tail fiber

proteins of various phages including *Shigella* phages, *Escherichia* phages, *Yersinia* phages and a *Citrobacter* phage. More importantly, both tail fiber proteins are similar to two proteins of the Enterobacteria phage T4. The first is short tail fiber gp12 (identity score: 93.7%, E-value of approximately zero) and the second is long tail fiber proximal subunit gp34 (identity score: 63.6%, E-value of approximately zero). Gp12 is an adhesion protein that binds irreversibly to lipopolysaccharides (LPS) on the cell surface of *E. coli* (Weigele *et al.*, 2003). Gp34 forms the proximal half of the long tail fiber that connects to the baseplate of the virion (Cerritelli *et al.*, 1996). The predicted tail connector protein shows significant similarity to long tail fiber protein gp35 of Enterobacteria phage T4, with an identity score of 96.2% and an E-value of approximately zero. Gp35 proposedly forms the hinge connecting the proximal and distal half of the long tail fiber protein of T4. Potentially, the identified proteins of Salmonella phage SG1 either target components that are shared across bacterial genera, or the proteins constitute parts of tail fiber proteins that are not determinant for host-specificity. Cerritelli *et al.* mention that there is evidence that gp34 is indeed conserved among tail fiber genes of coliphages (Cerritelli *et al.*, 1996). The same could be true for the other identified proteins of Salmonella phage SG1. Another possibility is that these proteins are falsely annotated to be Salmonella phage SG1 proteins. However, the reliability of the host and protein annotation could not be verified. Therefore, these are merely speculations. Further research should be conducted to characterize these proteins in more detail.

Pairwise sequence alignment scores of all proteins in the dataset were used to perform hierarchical clustering. The results are visualized in Figure 4.4. The colors in the heat map indicate the alignment scores. Higher alignment scores correspond to higher similarity between the proteins. This cluster analysis indicates several things. The proteins in cluster one seem considerably more similar than proteins in other clusters. In this cluster, 37 proteins are related to *E. coli*, while one protein is related to *S. enterica*. The latter is one of the tail fiber proteins from Salmonella phage SG1, which was also identified previously using LDA. Cluster two also contains one protein related to *S. enterica*, while the rest belongs to *E. coli*. This is the tail connector protein of Salmonella phage SG1, also previously mentioned. Cluster three shows subclusters. This could indicate subspecies specificity, in which slightly different proteins could interact with slightly different bacterial receptors. In this cluster as well, one protein is present related to *S. enterica*. This is the second tail fiber proteins of Salmonella phage SG1. Both LDA and cluster analysis thus identified the same three proteins of Salmonella phage SG1 as being closely related to proteins that are linked to *E. coli* as host. Most of the other proteins related to the host *S. enterica* cluster together without observable subclusters. Strangely, the proteins related to *K. pneumoniae* do not cluster together well. This could indicate that some bacterial

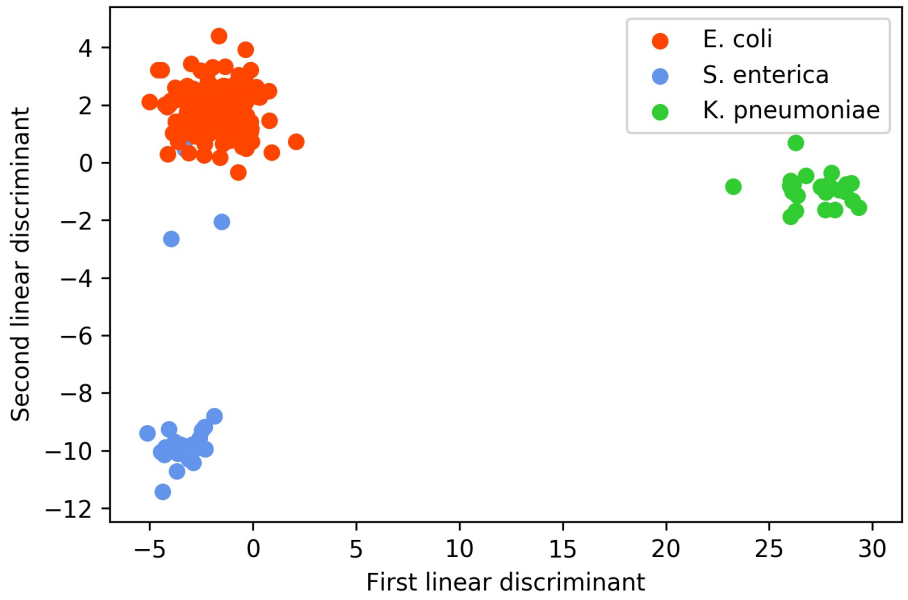


Figure 4.3: Plot of the linear discriminants of LDA using the entire set of features.

Plot of the first linear discriminant of LDA on the x-axis and the second linear discriminant on the y-axis. The linear discriminants were computed using the entire set of standardized features. Proteins were colored according to their related host.

receptors of *K. pneumoniae* are similar to receptors of *E. coli*, which corresponds to the similarity in the proteins necessary for their recognition. Looking more generally at the differences between clusters, the results indicate that phages infecting *E. coli* can do so using different proteins, probably corresponding to different bacterial receptors (discussed in Section 3.2). One possible explanation is that these different bacterial receptors are characteristics of different *E. coli* strains, further indicating subspecies specificity. However, as strain info was not included in the analysis, no conclusions can be made regarding subspecies specificity. Another possibility is that the corresponding tail fiber or tail spike proteins are involved in different stages of the adsorption process (as described in Section 3.3). Furthermore, as *S. enterica* is closely related to *E. coli*, subclusters of proteins related to *S. enterica* would also be expected, corresponding to different bacterial receptors on the surface of *S. enterica*. This could be the result of a lack of diversity in the dataset. Including more proteins related to both *S. enterica* and *K. pneumoniae* would allow for a more comprehensive analysis of the clusters that they form.

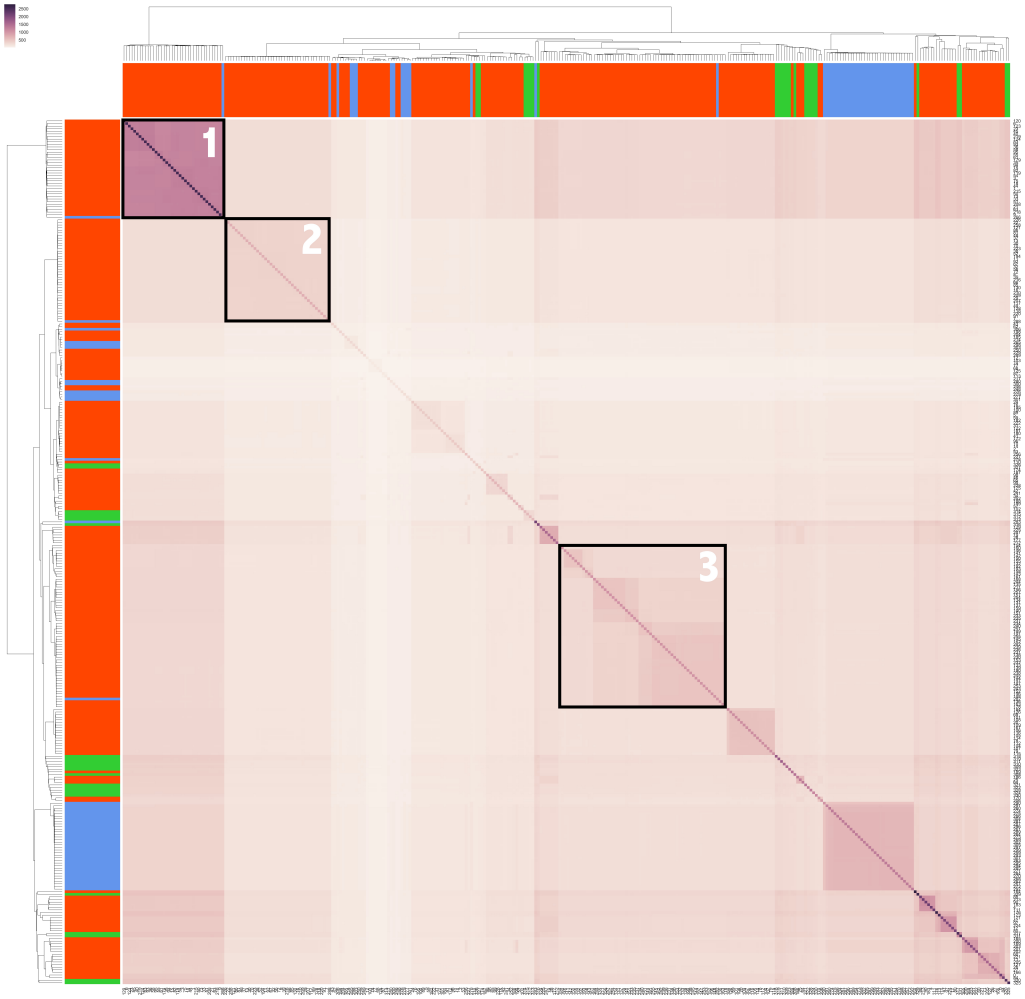


Figure 4.4: Hierarchically clustered heat map of the proteins in the dataset based on their pairwise alignment scores.

The figure displays the protein sequences in the dataset after hierarchical clustering based on their pairwise alignment scores. The colors in the heat map indicate the alignment scores. Higher alignment scores correspond to more similar proteins. The colors at each branch of the tree correspond to the bacterial host which the protein is related to. Red instances correspond to *E. coli* as host, blue instances correspond to *S. enterica* as host and green instances correspond to *K. pneumoniae* as host.

4.3.2 Model performance

Table 4.4 summarizes the resulting performance metrics for the different models used in this method. Overall, all models show good and similar performances. Every model scores above 88% across all performance metrics. These results demonstrate that tail fiber and tail spike protein data is indeed appropriate to predict the bacterial host that these proteins are related to. The best performing model is LDA. It consistently outperforms the other models on all performance metrics. This indicates that a simple linear decision boundary is sufficient to separate the different classes. On the other hand, logistic regression is also a linear model though it has a lower performance,

Table 4.4: Measured performance metrics for the different machine learning models after four-fold nested cross-validation.

	Accuracy	Precision	Recall	F1
GB	0.90	0.91	0.90	0.89
LDA	0.93	0.93	0.93	0.92
RF	0.90	0.90	0.92	0.90
logistic (L1)	0.89	0.91	0.88	0.89
logistic (L2)	0.89	0.91	0.89	0.90

especially with L_1 regularization. The difference between logistic regression and LDA is that logistic regression models the probabilities of belonging to a particular class by using the logistic function. Logistic regression does not assume a specific density function for the different classes. On the contrary, LDA models the different classes as separate multivariate Gaussian distributions. If these distributions are good approximations of the actual distributions, it can be expected for LDA to perform better than logistic regression. Furthermore, the non-linear methods also perform slightly worse than LDA. If a linear boundary is able to separate the classes, then linear methods model this decision boundary more easily, which can result in them outperforming non-linear methods (Hastie *et al.*, 2001).

To further examine the performance of the best performing model, a confusion matrix was computed for predictions with LDA, again using four-fold cross-validation. A confusion matrix summarizes the number of instances in each actual class (corresponding to the rows of the table) versus the number of instances predicted by the model to be in each class (corresponding to the columns) (James *et al.*, 2013). In this way, the confusion matrix provides a simple way to identify the classes that are most difficult to predict. Table 4.5 gives this confusion matrix for LDA. As an example, of the 259 instances actually related to *E. coli*, LDA correctly predicts 253 instances while falsely classifying four instances in the *S. enterica* class and one instance in the *K. pneumoniae* class. Looking at the predicted class labels (columns), LDA mostly has difficulty with classes *E. coli* and *S. enterica*, falsely classifying sixteen instances to the *E. coli* class and six instances to the *S. enterica* class. Only two instances were falsely classified as belonging to the *K. pneumoniae* class. Looking at the actual classes, most errors were made for classes *S. enterica* and *K. pneumoniae*. Twelve out of fourty-eight instances from the *S. enterica* class were falsely classified in another class. Likewise, seven of twenty-four instances from *K. pneumoniae* were falsely classified in another class. Both classes are not abundant in the dataset. These results indicate that a large number of instances (as is the case for the *E. coli* class), results in a higher predictive power. Adding more proteins related to *S. enterica* and *K. pneumoniae* in the dataset would likely reduce the number of missclassified instances in these

Table 4.5: Confusion matrix of predictions made using LDA. Rows represent the actual class labels, while columns represent the predicted class labels.

	E. coli	*S. enterica*	*K. pneumoniae*
E. coli	253	4	1
S. enterica	11	36	1
K. pneumoniae	5	2	17

classes. For this, better data annotation resulting from experimental characterization of these proteins is needed.

Additionally, some remarks can be made regarding to the obtained results. As machine learning models only learn patterns from the data that were used to train these models, limitations in the generalizability of the models mainly reside in the quality and diversity of the dataset. Here, the dataset is both limited in size and biased towards proteins needed for infection of a single species, *E. coli*. Therefore, it is likely for the models to be limited in their generalizability due to limited diversity, especially for the classes of *S. enterica* and *K. pneumoniae* that are only represented by a small number of entries in the dataset. One factor that could add to this limited generalizability is the fact that the dataset was extended by mapping the originally found tail fiber and tail spike proteins to UniRef. Here, new proteins that were not annotated properly were found by clustering the originally found proteins to new proteins that were at least 90% identical. This biases the dataset towards sequences that are higly alike and thus are more easily separable from other sequences. On the other hand, as described above, performing BLASTP for the tail fiber proteins of Salmonella phage SG1 identified several tail fiber proteins from phages infecting different bacterial genera. Clustering the originally found proteins might have had the same effect, thus improving the diversity of the dataset. It is not clear to what extent these biases in the dataset influence the performance of the models, though it is possible that the models do not generalize well to new data. In any case, a more diversified dataset would contribute further to a better understanding of phage-host specificity. Furthermore, the classification performed here predicts bacterial hosts at the species level. As explained before, a considerable number of phages exhibit specificity at the strain level. However, public databases generally only mention a phage's host at the species level. Therefore, it was chosen to perform this classification at the species level.

4.3.3 Feature importance

The adopted machine learning approach can also be of value by looking at the features that are most important in the classification. In doing so, the tail fiber and tail

spike proteins used in these methods can be characterized based on the features used to describe them. This could further increase the understanding of how phage-host specificity works at the protein level or be used for the *in silico* design of tail fiber and tail spike proteins for various applications.

The upper plot of Figure 4.5 displays the absolute values of the coefficients of logistic regression with L_1 regularization for every class. The lower plot displays the Gini importance (from the RF model) for every feature. The magnitude of these coefficients and Gini importances represent the importance of that particular feature in the classification. A larger magnitude indicates a higher importance of that feature in the classification. Based on this result, the features identified as most important for each of the models were the following:

- **Features important in logistic regression:** ACA codon frequency, CCC codon frequency, GGG codon frequency, CCT codon frequency, GAG codon frequency, glutamine (Q) frequency, isoleucine (I) frequency, fraction of AAs in α-helices and the fraction of positively charged AAs.
- **Features important in RF:** ACA codon frequency, ATA codon frequency, CCC codon frequency, GGG codon frequency, ACA codon usage bias, ACC codon usage bias, GGG codon usag bias, leucine (L) frequency, isoleucine (I) frequency, fraction of AAs in α-helices and fraction of polar AAs.

As a third analysis, feature importance was also visualized by constructing a decision tree in R. By pruning the tree, only the most informative features are considered for making decisions. The pruned decision tree is visualized in Figure 4.6. Every node in the tree represents a specific decision made by the decision tree in order to discriminate between the different classes. Here, the most informative features were the ATA codon frequency, the A nucleotide frequency, the fraction of uncharged polar AAs, the fraction of AAs in α-helices and the G nucleotide frequency. The numerical cut-off values are less interpretable as all features were standardized beforehand. These cut-off values can thus only be interpreted as the magnitude of deviation from the mean value of that particular feature in the dataset.

All three methods identify the fraction of AAs in α-helices as an important discriminating feature. Furthermore, two out of three models identify the isoleucine (L) frequency, the ACA codon frequency, the ATA codon frequency, the CCC codon frequency and the GGG codon frequency as important discriminating features. These six features are visualized both as histograms and in a pairwise manner in Figure 4.7. Due to the large size of this figure, it is also given in digital appendix C. From the

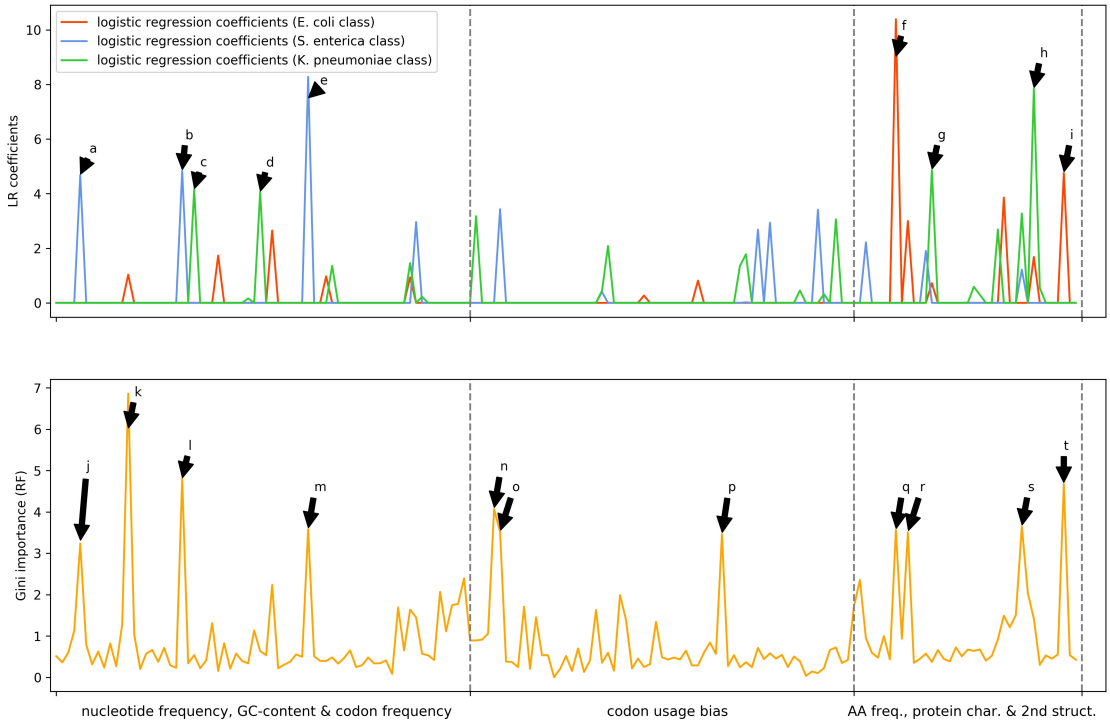


Figure 4.5: Plot of the feature importance calculated by logistic regression (blue) and RF (orange).

The figure displays the feature importance as given by the coefficients of logistic regression with L_1 regularization in blue and the Gini importance of the features calculated by RF in orange. The magnitude of these coefficients and Gini importances represent the importance of that particular feature in the classification. After applying an arbitrary cut-off of 4, the peaks from logistic regression represent the following features: ACA codon frequency (a), CCC codon frequency (b), CCT codon frequency (c), GAG codon frequency (d), GGG codon frequency (e), isoleucine (I) frequency (f), glutamine (Q) frequency (g), the fraction of positively charged AAs (h) and the fraction of AAs in α-helices (i). Applying an arbitrary cutoff of 2.5, the features identified as important in RFs were ACA codon frequency (j), ATA codon frequency (k), CCC codon frequency (l), GGG codon frequency (m), ACA codon usage bias (n), ACC codon usage bias (o), GGG codon usage bias (p), isoleucine (I) frequency (q), leucince (L) frequency (r), the fraction of polar AAs (s) and the fraction of AAs in α-helices (t).

histograms, it is clear that a single feature is not able to properly separate the three different classes. However, some features might be adequate to properly separate two of the classes. For example, the isoleucine frequency allows to separate proteins related to *E. coli* from proteins related to *S. enterica* to a proper extent. Additionally, the CCC codon frequency can also be used to separate the *E. coli* and *S. enterica* classes to a certain extent. Considering the pairwise plots, again, none of the pairs of features was able to separate the three different classes appropriately. This indicates that biological rules are complex. A protein function is too complex to be described

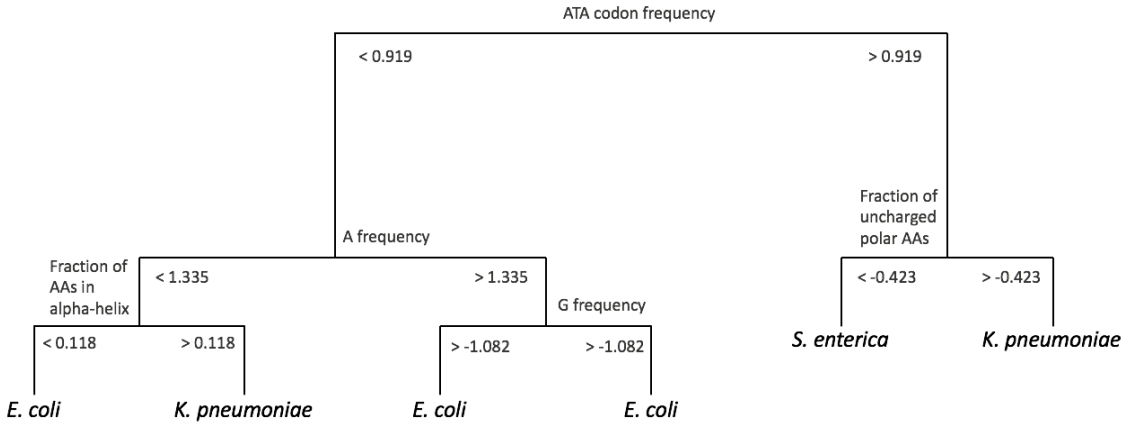


Figure 4.6: Plot of the pruned tree computed using the standardized features.

The figure shows the pruned tree obtained in R using the standardized features. Every node constitutes a specific decision made by the decision tree to discriminate between the different classes.

Table 4.6: Measured performance metrics for the different machine learning models after four-fold nested cross-validation using only the four most important features.

	Acc	P	R	f1
GB	0.90	0.89	0.90	0.89
LDA	0.90	0.87	0.90	0.88
RF	0.92	0.90	0.91	0.91
logistic (L1)	0.85	0.88	0.81	0.86
logistic (L2)	0.86	0.90	0.86	0.86

with simple features. Even though these features are important in classification, a single characteristic or only a pair of characteristics of the protein or DNA sequence will never be informative enough to explain the difference in host specificity.

Finally, the five models considered in Section 4.3.2 were trained, tuned and tested again using only the four features identified as most important. The results are given in Table 4.6. The results indicate that these features alone are capable of discriminating between the three different classes quite well, with performances varying from around 81% to 92% across different metrics.

Together, this section shows that machine learning can be used to identify features that are important in the discrimination of proteins related to the infection of different hosts. However, as phage-host specificity is complex, one single feature or even a set of two features is not informative enough to explain the difference in phage-host specificity. Decision trees might be an informative visualization for which combination of different features is used to disriminate between different classes. However, it is harder to explain how these features biologically contribute to a difference in host specificity.

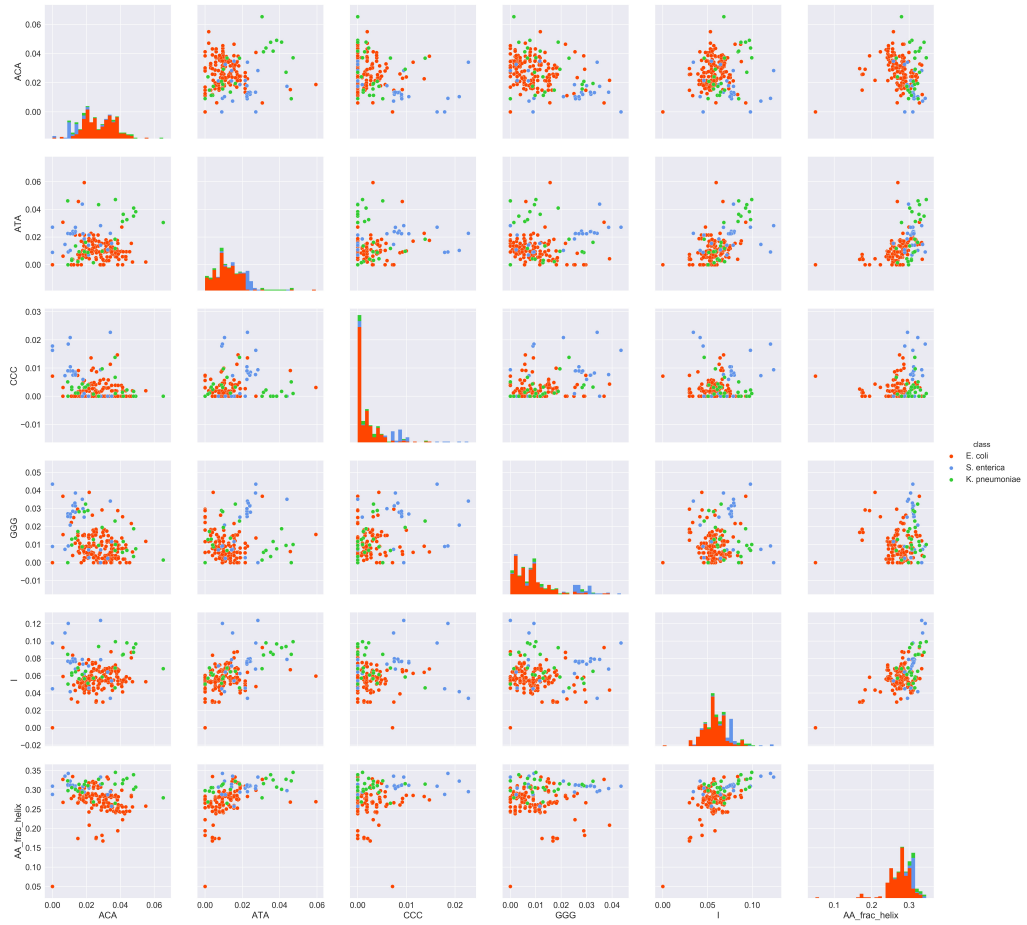


Figure 4.7: Pairwise plots of the four features identified as most important by several models.

Visualization of the ACA codon frequency, ATA codon frequency, CCC codon frequency, GGG codon frequency, isoleucine (I) frequency and the fraction of AAs in α-helices, both as histograms and as pairwise plots. Every instance was colored according to the related bacterial host.

4.4 Conclusion

In conclusion, this chapter introduced several machine learning methods that were capable of discriminating between proteins related to the infection of three different bacterial species based on features derived from the protein sequences and CDSs. While this approach exhibits some limitations, the results above show that machine learning can be used effectively to study phage-host specificity in an way that is both original and informative. Accurate *in silico* predictions can further increase the understanding of phage-host specificity. The biggest drawback is the lack of a diversified, large dataset from which the machine learning models learn patterns. A second drawback is the lack of host annotation at the strain level. In addition, phage-host specificity cannot completely be explained based on a limited set of features. On the other hand, decision trees can still be practical to visualize the features that are hi-

erarchically used to separate different classes. This could give a more detailed view of which combination of characteristics explains the difference in host specificity to a certain extent. Finally, a potential application of this analysis might also be the discovery of falsely annotated proteins, or phages with tail fibers that could potentially infect previously unknown hosts.

CHAPTER 5

Conclusions and future perspectives

5.1 Conclusions

In this work, phage-host specificity was studied in a computational context. One of the central questions of this project was whether phage-host specificity could be identified and explained at the proteome and protein level. Subsequently, the aim of this project was to identify the factors determining phage-host specificity using computational tools. More specifically, two computational techniques were used to study phage proteomes and proteins: optimal transport and machine learning.

In Chapters two and three, optimal transport was used as a mathematical framework to identify similarities and differences in biological data by representing these data as probability distributions. As a proof-of-concept, the technique was first applied to complete phage proteomes in order to construct a phage tree. The results indicate that Sinkhorn distances can be a good measure for similarity among phages, provided that appropriate values are chosen for the parameters k and λ. In Chapter three, optimal transport was applied again at the level of specific proteins to reveal the differences between three members of the T7 group of phages. Here, optimal transport was combined with a threshold for statistical significance and a threshold for biological significance to ultimately identify the proteins that were unique in phages with similar proteomes but a different host. Although this method provides an original strategy to assess the differences between related phages, several shortcomings still have to be overcome to put this method to better use. A comparison between the proteomes of Enterobacteria phage T7 and Erwinia phage vB_EamP-L1 with optimal transport was able to identify the tail fiber protein and EPS depolymerase as unique in both phages, respectively. On the other hand, some proteins were identified as unique while BLASTP revealed significant matches to proteins in the phage proteome that these were compared to. Potentially, false results can be prevented by using a higher value for k and a more negative cut-off value for effect size. Additionally,

repeating the analysis with more T7-like phages (that have a different host) and only considering proteins as unique if they are unique among all comparisons could avoid falsely identifying a protein as unique, while it is not necessarily linked to a difference in host specificity.

In Chapter four, tail fiber and tail spike protein data were used to train several machine learning models. These models were used to predict the bacterial host related to these proteins in a three-class classification setting. In addition, the proteins were characterized based on the features that were considered as most important in classification. The results indicate that using tail fiber and tail spike data and machine learning models are indeed informative in predicting bacterial hosts of the phages that encode these proteins, more so than an alignment based approach. To conclude, this method forms an interesting basis for future applications of phage-host predictions.

Furthermore, machine learning models can be used to characterize protein data based on the features that were constructed to train the models. Here, it became clear that a simple set of features is not informative enough to discriminate between proteins related to differents hosts. Although phages arguably constitute the most simple biological entities, the way their host specificity works can be complex. Nevertheless, visualization tools such as decision trees can help discover what combination of features discriminates proteins that are related to different bacterial hosts.

A final conclusion is that both an inadequate amount as well was an improper quality of data (and annotation) is a serious bottleneck. Both the lack of protein annotation and the absence of strain information regarding the bacterial hosts hampered a more comprehensive biological interpretation of the analyses. A large proportion of the identified unique proteins had no specified function, indicating a lack of proper annotation in biological databases. Therefore, these proteins could not be linked to a difference in phage-host specificity. This lack of knowledge on the biological function of phage proteins also limited the number of tail fiber proteins to train machine learning models. Here, a more diversified dataset, particularly for the classes that were not well represented, would likely result in a broader understanding of the difference of host specificity between these classes of phage proteins. However, as data on phages and their proteins become increasingly abundant, the tools developed in this work will help to provide even more insight into phage-host specificity in the future. To that extent, the next section explores some ideas and suggestions for future research that build on the developed methods and analyses done in this project.

5.2 Future perspectives

The two computational approaches used in this work provide new ways of exploring phage proteomes and proteins to study phage-host specificity. As a result, there are several possible extensions of this work. Below, three of those are elaborated upon.

5.2.1 Improving the developed computational methods and datasets

A straightforward first step is to improve the developed methods. Several steps can be optimized. Regarding optimal transport, k-mer length is a parameter that should be studied in more detail. Although correlation between alignment scores and Sinkhorn distances is higher for low values of k, it is hypothesized that higher values of k would work better in the identification of unique proteins among phage proteomes. Possibly, this could also result in a tree that is even more congruent to the current references. Furthermore, the cut-off value for effect size used to identify biologically relevant proteins with a significant match should be tuned to give more relevant results. However, the identification of correct unique proteins depends on the result of the value of k in combination with the cut-off value for effect size and will differ according to the particular choice of phages. It would be interesting to repeat the analysis using a larger number of T7-like phages with different hosts. Instead of comparing proteomes in a pairwise manner, every protein in one proteome could instead be compared directly to every protein in all other proteomes with different hosts. Finally, this work did not make use of the transportation matrix P^*. This matrix could be a useful visualization tool to further study the (dis)similarities between the k-mer distributions of proteins. For example, similar k-mers could be identified and then mapped back to the protein sequences, which could prove useful in identifying the determinants of host specificity at the level of specific domains.

With regard to the machine learning methods, one of the most important aspects will be to improve on both the quantity and quality of the dataset. To start, a more in-depth review of literature that characterizes tail fiber proteins could provide extra proteins for the dataset. Furthermore, as biological databases are frequently updated, automating the process of data gathering and preprocessing would be advantageous and would also result in the expansion of the dataset. Another improvement would be the addition of extra features based on secondary and tertiary structure of these proteins. In general, a better representation of proteins could be advantageous for predictions. One possible alternative representation is the use of embeddings, as explained by Yang *et al.* (2018). The end goal would be to be capable of predicting

the host based on new tail fiber sequences, for a large number of hosts. Additionally, the goal would be to fully understand how tail fiber and tail spike proteins differ among phages, both with the same host and different hosts. Machine learning models as well as other tools such can help reach this goal.

5.2.2 An approach to identify tail fiber proteins in viral metagenomics data

As there is no single homologous gene present in all phages, viral taxonomy is challenging. This is particularly problematic in a metagenomics context, in which only sequence data is available. Most metagenomics studies use homology searches to cope with this. However, viral metagenomics is often characterized by a large number of sequences having no significant similarity to any other sequence due to heterogeneity of viral genomes and low coverage of the global virome (Mokili *et al.*, 2012; Ren *et al.*, 2017). As a result, it is not straightforward to identify viral sequences in metagenomic data and dicriminate them from prokaryotic sequences. However, several studies developed methods to be able to correctly distinguish viral sequences from non-viral sequences.

Deaton *et al.* (2017) developed machine learning models that identify unknown sequences from a metagenomic sample as phage or non-phage using tetranucleotide frequencies (Deaton *et al.*, 2017). Another study by Ren *et al.* (2017) also used *k*-mer frequency profiles to predict whether the contig (i.e. overlapping sequence reads in a metagenome) represents a viral sequence or not (Ren *et al.*, 2017). An interesting addition to both approaches would be to predict tail fiber and tail spike genes from metagenomic data. Identifying these genes could not only be useful to identify viral sequences in metagenomic data but also to predict the bacterial host(s) from these tail fiber and tail spike genes. Such a tool would require the construction of machine learning models able to discriminate between tail fiber or tail spike genes and other viral and non-viral genes. Subsequently, this tail fiber predictor tool could be coupled to the machine learning models used in this work, to predict the bacterial host related to this gene or protein.

One potential disadvantage of this method is the fact that it is still gene-based. If no gene can be predicted from a contig, the method cannot make a prediction about whether a tail fiber gene could be present in that contig. In addition, it remains an open question whether tail fiber or tail spike genes or proteins could be sufficiently distinguished from other viral and non-viral genes and proteins. Another challenge is extending the developed machine learning models to make predictions for a larger number of bacterial species. A less complex extension would be to make predictions

at higher taxonomic levels. However, if this proves successful, potentially a lot of new tail fiber proteins can be identified in otherwise unexplored data. These new tail fiber sequences could be used to develop synthetic viruses and construct a phage bank of synthetic phages, annotated with their predicted host. This would be advantageous both to applications in phage therapy as well as the development of customized tailocins.

5.2.3 *In silico* design of synthetic tail fibers

Tail fiber and tail spike proteins have several applications, including specific detection of pathogens[1] or the development of synthetic viruses to edit microbial populations (Ando *et al.*, 2015). Therefore, another possible extension of this work is the use of the developed machine learning methods for *in silico* protein engineering. The goal here would be to design proteins with optimal characteristics for a specific application. One way to aid in this development would be to use patterns and rules (e.g. distilled from decision trees) to design proteins *in silico*. Another approach would be to optimize proteins using machine learning models as cost function in optimization methods, for example using Bayesian optimization (Cui and Yang, 2018). As these machine learning have already learned patterns regarding the data that was used to train them, using them as cost function in optimization looks like a promising approach to design proteins with optimal characteristics *in silico*.

[1] www.biomerieux.com.au/industrial-microbiology/food/pathogen-detection

Bibliography

Ackermann, H. W. (2007). 5500 Phages examined in the electron microscope. Archives of Virology, 152(2), 227243. http://doi.org/10.1007/s00705-006-0849-1

Adriaenssens, E. M., Edwards, R., Nash, J. H. E., Mahadevan, P., Seto, D., Ackermann, H. W., Kropinski, A. M. (2015). Integration of genomic and proteomic analyses in the classification of the Siphoviridae family. Virology, 477, 144154. http://doi.org/10.1016/j.virol.2014.10.016

Ahmed, S., Saito, A., Suzuki, M., Nemoto, N., and Nishigaki, K. (2009). Host-parasite relations of bacteria and phages can be unveiled by Oligostickiness, a measure of relaxed sequence similarity. Bioinformatics, 25(5), 563570. http://doi.org/10.1093/bioinformatics/btp003

Al-Shahib, A., Breitling, R., and Gilbert, D. R. (2007). Predicting protein function by machine learning on amino acid sequences - A critical evaluation. BMC Genomics, 8. http://doi.org/10.1186/1471-2164-8-78

Ando, H., Lemire, S., Pires, D. P., and Lu, T. K. (2015). Engineering Modular Viral Scaffolds for Targeted Bacterial Population Editing. Cell Systems, 1(3), 187196. http://doi.org/10.1016/j.cels.2015.08.013

Bateman, A., Martin, M. J., ODonovan, C., Magrane, M., Alpi, E., Antunes, R., Zhang, J. (2017). UniProt: The universal protein knowledgebase. Nucleic Acids Research, 45(D1), D158D169. http://doi.org/10.1093/nar/gkw1099

Benson, D. A., Cavanaugh, M., Clark, K., Karsch-Mizrachi, I., Lipman, D. J., Ostell, J., and Sayers, E. W. (2013). GenBank. Nucleic Acids Research, 41(D1), 3642. http://doi.org/10.1093/nar/gks1195

Cenens, W., Makumi, A., Mebrhatu, M. T., Lavigne, R., and Aertsen, A. (2013). Phage-host interactions during pseudolysogeny. Bacteriophage, 3(1), e25029. http://doi.org/10.4161/bact.25029

Cerritelli, M. E., Wall, J. S., Simon, M. N., Conway, J. F., and Steven, A. C. (1996). Stoichiometry and domainal organization of the long tail-fiber of bacteriophage T4: A

hinged viral adhesin. Journal of Molecular Biology, 260(5), 767780. http://doi.org/10.1006/jmbi.1996.0436

Chaturongakul, S., and Ounjai, P. (2014). Phage-host interplay: Examples from tailed phages and Gram-negative bacterial pathogens. Frontiers in Microbiology, 5(AUG), 18. http://doi.org/10.3389/fmicb.2014.00442

Clokie, M. R. J., Millard, A. D., Letarov, A. V., and Heaphy, S. (2011). Phages in nature. Bacteriophage, 1(1), 3145. http://doi.org/10.4161/bact.1.1.14942

Cock, P. J. A., Antao, T., Chang, J. T., Chapman, B. A., Cox, C. J., Dalke, A., De Hoon, M. J. L. (2009). Biopython: Freely available Python tools for computational molecular biology and bioinformatics. Bioinformatics, 25(11), 14221423. http://doi.org/10.1093/bioinformatics/btp163

Cuervo, A., Pulido-Cid, M., Chagoyen, M., Arranz, R., González-García, V. A., Garcia-Doval, C., Carrascosa, J. L. (2013). Structural characterization of the bacteriophage T7 tail machinery. Journal of Biological Chemistry, 288(36), 2629026299. http://doi.org/10.1074/jbc.M113.491209

Cui, J., Yang, B. (2018). Graph Bayesian Optimization: Algorithms, Evaluations and Applications. Journal of Machine Learning Research, 18, 1-24.

Cuturi, M. (2013). Sinkhorn Distances: Lightspeed Computation of Optimal Transportation Distances, 113. Retrieved from http://arxiv.org/abs/1306.0895

Das, S., Deb, T., Dey, N., Ashour, A. S., Bhattacharya, D. K., and Tibarewala, D. N. (2017). Optimal choice of k-mer in composition vector method for genome sequence comparison. Genomics, (November), 01. http://doi.org/10.1016/j.ygeno.2017.11.003

Daudén, M. I., Marti-Benito, J., Sánchez-Ferrero, J. C., Pulido-Cid, M., Valpuesta, J. M., and Carrascosa, J. L. (2013). Large terminase conformational change induced by connector binding in bacteriophage T7. Journal of Biological Chemistry, 288(23), 1699817007. http://doi.org/10.1074/jbc.M112.448951

Davies, J., and Davies, D. (2010). Origins and Evolution of Antibiotic Resistance. Microbiology and Molecular Biology Reviews, 74(3), 417433. http://doi.org/10.1128/MMBR.00016-10

Deaton, J., Yu, F. B., and Quake, S. R. (2017). PhaMers identifies novel bacteriophage sequences from thermophilic hot springs, (September), 131. http://doi.org/10.1101/126847

Díaz-Muñoz, S. L., and Koskella, B. (2014). Bacteria-Phage interactions in natural environments. Advances in Applied Microbiology (1st ed., Vol. 89). Elsevier Inc. http://doi.org/10.1016/B978-0-12-800259-9.00004-4

Doss, J., Culbertson, K., Hahn, D., Camacho, J., and Barekzi, N. (2017). A review of phage therapy against bacterial pathogens of aquatic and terrestrial organisms. Viruses, 9(3). http://doi.org/10.3390/v9030050

Duplessis, M., and Moineau, S. (2001). Identification of a genetic determinant responsible for host specificity in Streptococcus thermophilus bacteriophages. Molecular Microbiology, 41(2), 325336. http://doi.org/10.1046/j.1365-2958.2001.02521.x

Dupont, K., Vogensen, F. K., Neve, H., Bresciani, J., and Josephsen, J. (2004). Identification of the receptor-binding protein in 936-species lactococcal bacteriophages. Applied and Environmental Microbiology, 70(10), 58185824. http://doi.org/10.1128/AEM.70.10.5818-5824.2004

Edwards, R. A., McNair, K., Faust, K., Raes, J., and Dutilh, B. E. (2016). Computational approaches to predict bacteriophage-host relationships. FEMS Microbiology Reviews, 40(2), 258272. http://doi.org/10.1093/femsre/fuv048

Egley, L. E. W., and Breitbart, M. (2003). Use of Fluorescently Labeled Phage in the Detection and Identi cation of Bacterial Species. Applied Spectroscopy, 57(9), 7. http://doi.org/10.1366/00037020360696008

Eickholt, J., and Wang, Z. (2014). PCP-ML: Protein characterization package for Machine Learning. BMC Research Notes, 7(1), 15. http://doi.org/10.1186/1756-0500-7-810

Fàbrega, A., and Vila, J. (2013). Salmonella enterica serovar Typhimurium skills to succeed in the host: Virulence and regulation. Clinical Microbiology Reviews, 26(2), 308341. http://doi.org/10.1128/CMR.00066-12

Fineran, P. C., Blower, T. R., Foulds, I. J., Humphreys, D. P., Lilley, K. S., and Salmond, G. P. C. (2009). The phage abortive infection system, ToxIN, functions as a protein-RNA toxin-antitoxin pair. Proceedings of the National Academy of Sciences, 106(3), 894899. http://doi.org/10.1073/pnas.0808832106

Finn, R. D., Attwood, T. K., Babbitt, P. C., Bateman, A., Bork, P., Bridge, A. J., Mitchell, A. L. (2017). InterPro in 2017-beyond protein family and domain annotations. Nucleic Acids Research, 45(D1), D190D199. http://doi.org/10.1093/nar/gkw1107

Fokine, A., and Rossmann, M. G. (2014). Molecular architecture of tailed double-stranded DNA phages. Bacteriophage, 4(2), e28281. http://doi.org/10.4161/bact.28281

Ghequire, M. G. K., and De Mot, R. (2015). The Tailocin Tale: Peeling off Phage Tails. Trends in Microbiology, 23(10), 587590. http://doi.org/10.1016/j.tim.2015.07.011

Giske, C. G., Monnet, D. L., Cars, O., and Carmeli, Y. (2008). Clinical and economic impact of common multidrug-resistant gram-negative bacilli. Antimicrobial Agents and Chemotherapy, 52(3), 813821. http://doi.org/10.1128/AAC.01169-07

Guruprasad, K., Reddy, B. V. B., and Pandit, M. W. (1990). Correlation between stability of a protein and its dipeptide composition: a novel approach for predicting in vivo stability of a protein from its primary sequence. Protein Engineering, Design and Selection, 4(2), 155161. http://doi.org/10.1093/protein/4.2.155

Hobohm, U., and Sander, C. (1995). A sequence property approach to searching protein databases. Journal of Molecular Biology, 251(3), 390399. http://doi.org/10.1006/jmbi.1995.0442

Hoess, R. H., and Landy, A. (1978). Structure of the lambda att sites generated by int-dependent deletions. Proceedings of the National Academy of Sciences, 75(11), 54375441. http://doi.org/10.1073/pnas.75.11.5437

Horvath, P., Barrangou, R. (2010). CRISPR/Cas, the Immune System of Bacteria and Archaea. Source: Science, New Series, 327(5962), 167170. http://doi.org/10.1126/science.1179555

Hyman, P., and Abedon, S. T. (2010). Bacteriophage host range and bacterial resistance. Advances in applied microbiology (1st ed., Vol. 70). Elsevier Inc. http://doi.org/10.1016/S0065-2164(10)70007-1

Jiang, S. C., and Paul, J. H. (1996). Occurrence of lysogenic bacteria in marine microbial communities as determined by prophage induction. Marine Ecology Progress Series, 142, 2738. http://doi.org/10.3354/meps142027

Kiro, R., Molshanski-Mor, S., Yosef, I., Milam, S. L., Erickson, H. P., and Qimron, U. (2013). Gene product 0.4 increases bacteriophage T7 competitiveness by inhibiting host cell division. Proceedings of the National Academy of Sciences, 110(48), 1954919554. http://doi.org/10.1073/pnas.1314096110

Koskella, B., and Meaden, S. (2013). Understanding bacteriophage specificity in natural microbial communities. Viruses, 5(3), 806823. http://doi.org/10.3390/v5030806

Kwan, T., Liu, J., DuBow, M., Gros, P., and Pelletier, J. (2005). The complete genomes and proteomes of 27 Staphylococcus aureus bacteriophages. Proceedings of the National Academy of Sciences, 102(14), 51745179.
http://doi.org/10.1073/pnas.0501140102

Latka, A., Maciejewska, B., Majkowska-Skrobek, G., Briers, Y., and Drulis-Kawa, Z. (2017). Bacteriophage-encoded virion-associated enzymes to overcome the carbohydrate barriers during the infection process. Applied Microbiology and Biotechnology, 101(8), 31033119. http://doi.org/10.1007/s00253-017-8224-6

Lavigne, R., Seto, D., Mahadevan, P., Ackermann, H. W., and Kropinski, A. M. (2008). Unifying classical and molecular taxonomic classification: analysis of the Podoviridae using BLASTP-based tools. Research in Microbiology, 159(5), 406414.
http://doi.org/10.1016/j.resmic.2008.03.005

Le, S., He, X., Tan, Y., Huang, G., Zhang, L., Lux, R., Hu, F. (2013). Mapping the Tail Fiber as the Receptor Binding Protein Responsible for Differential Host Specificity of Pseudomonas aeruginosa Bacteriophages PaP1 and JG004. PLoS ONE, 8(7), 18. http://doi.org/10.1371/journal.pone.0068562

Leite, D. M. C., Brochet, X., Resch, G., Que, Y.-A., Neves, A., and Pena-Reyes, C. (2017). Computational Prediction of Host-Pathogen Interactions Through Omics Data Analysis and Machine Learning, 10209, 1530. http://doi.org/10.1007/978-3-319-56154-7

Lenski, R. E. (1984). Coevolution of bacteria and phage: Are there endless cycles of bacterial defenses and phage counterdefenses? Journal of Theoretical Biology, 108(3), 319325. http://doi.org/10.1016/S0022-5193(84)80035-1

Lévy, B., and Schwindt, E. L. (2018). Notions of optimal transport theory and how to implement them on a computer. Computers and Graphics (Pergamon), 72, 135148. http://doi.org/10.1016/j.cag.2018.01.009

Lindberg, A. A. (1973). Bacteriophage Receptors.

Lipman, D. J., Wilbur, W. J., Smith, T. F., and Waterman, M. S. (1984). On the statistical significance of nucleic acid similarities. Nucleic Acids Research, 12(1 Pt 1), 215226. Retrieved from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC320998/

Lobry, J. R., and Gautier, C. (1994). Hydrophobicity, expressivity and aromaticity are the major trends of amino-acid usage in 999 escherichia coli chromosome-encoded genes. Nucleic Acids Research, 22(15), 31743180.
http://doi.org/10.1093/nar/22.15.3174

Mahmood, K., Webb, G. I., Song, J., Whisstock, J. C., and Konagurthu, A. S. (2012). Efficient large-scale protein sequence comparison and gene matching to identify orthologs and co-orthologs. Nucleic Acids Research, 40(6).
http://doi.org/10.1093/nar/gkr1261

Maxwell, K. L., and Frappier, L. (2007). Viral Proteomics. Microbiology and Molecular Biology Reviews, 71(2), 398411. http://doi.org/10.1128/MMBR.00042-06

Mendelman, L. V, Notarnicola, S. M., and Richardson, C. C. (1992). Roles of bacteriophage T7 gene 4 proteins in providing primase and helicase functions in vivo. Proceedings of the National Academy of Sciences of the United States of America, 89(22), 1063842. http://doi.org/10.1073/pnas.89.22.10638

Menze, B. H., Kelm, B. M., Masuch, R., Himmelreich, U., Bachert, P., Petrich, W., and Hamprecht, F. A. (2009). A comparison of random forest and its Gini importance with standard chemometric methods for the feature selection and classification of spectral data. BMC Bioinformatics, 10, 116. http://doi.org/10.1186/1471-2105-10-213

Middelboe, M., Chan, A. M., and Bertelsen, S. K. (2010). Isolation and life cycle characterization of lytic viruses infecting heterotrophic bacteria and cyanobacteria. Manual of Aquatic Viral Ecology, (May 2018), 118133. http://doi.org/10.4319/mave.2010.978-0-9845591-0-7.118

Mihara, T., Nishimura, Y., Shimizu, Y., Nishiyama, H., Yoshikawa, G., Uehara, H., Ogata, H. (2016). Linking virus genomes with host taxonomy. Viruses, 8(3), 1015.
http://doi.org/10.3390/v8030066

Modi, S. R., Lee, H. H., Spina, C. S., and Collins, J. J. (2013). Antibiotic treatment expands the resistance reservoir and ecological network of the phage metagenome. Nature, 499(7457), 219222. http://doi.org/10.1038/nature12212

Mokili, J. L., Rohwer, F., and Dutilh, B. E. (2012). Metagenomics and future perspectives in virus discovery. Current Opinion in Virology, 2(1), 6377.
http://doi.org/10.1016/j.coviro.2011.12.004

Moldovan, R., Chapman-McQuiston, E., and Wu, X. L. (2007). On kinetics of phage adsorption. Biophysical Journal, 93(1), 303315.
http://doi.org/10.1529/biophysj.106.102962

Monge, G. (1781). Mémoire sur la théorie des déblais et des remblais. Histoire de lAcadmie Royale des Sciences (1781), 666704, 1784. (Original reference unatainable, cited from Lévy and Schwindt, 2017).

Nishimura, Y., Yoshida, T., Kuronishi, M., Uehara, H., Ogata, H., and Goto, S. (2017). ViPTree: The viral proteomic tree server. Bioinformatics, 33(15), 23792380. http://doi.org/10.1093/bioinformatics/btx157

ONeill, J. (2016). Tackling drug-resistant infections globally: final report and recommendations. The Review on Antimicrobial Resistance, (May), 84. http://doi.org/10.1016/j.jpha.2015.11.005

Park, M., Lee, J. H., Shin, H., Kim, M., Choi, J., Kang, D. H., Ryu, S. (2012). Characterization and comparative genomic analysis of a novel bacteriophage, SFP10, simultaneously inhibiting both Salmonella enterica and Escherichia coli O157:H7. Applied and Environmental Microbiology, 78(1), 5869. http://doi.org/10.1128/AEM.06231-11

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Duchesnay, É. (2012). Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research, 12, 28252830. http://doi.org/10.1007/s13398-014-0173-7.2

Pride, D. T., Wassenaar, T. M., Ghose, C., and Blaser, M. J. (2006). Evidence of host-virus co-evolution in tetranucleotide usage patterns of bacteriophages and eukaryotic viruses. BMC Genomics, 7, 113. http://doi.org/10.1186/1471-2164-7-8

Rakhuba, D. V., Kolomiets, E. I., Szwajcer Dey, E., and Novik, G. I. (2010). Bacteriophage receptors, mechanisms of phage adsorption and penetration into host cell. Polish Journal of Microbiology, 59(3), 145155. http://doi.org/10.1016/j.micres.2015.01.008.1.94

Ren, J., Ahlgren, N. A., Lu, Y. Y., Fuhrman, J. A., and Sun, F. (2017). VirFinder: a novel k-mer based tool for identifying viral sequences from assembled metagenomic data. Microbiome, 5(1), 69. http://doi.org/10.1186/s40168-017-0283-5

Rohwer, F., and Edwards, R. (2002). The Phage Proteomic Tree : a Genome-Based Taxonomy for Phage, 184(16), 45294535. http://doi.org/10.1128/JB.184.16.4529

Rohwer, F., Prangishvili, D., and Lindell, D. (2009). Roles of viruses in the environment. Environmental Microbiology, 11(11), 27712774. http://doi.org/10.1111/j.1462-2920.2009.02101.x

Roock, S. De, and Steven, M. (2014). A historical overview of bacteriophage therapy as an alternative to antibiotics for the treatment of bacterial pathogens, 5(1), 110.

Ross, A., Ward, S., and Hyman, P. (2016). More is better: Selecting for broad host range bacteriophages. Frontiers in Microbiology, 7(SEP), 16. http://doi.org/10.3389/fmicb.2016.01352

Rossolini, G. M., Arena, F., Pecile, P., and Pollini, S. (2014). Update on the antibiotic resistance crisis. Current Opinion in Pharmacology, 18, 5660. http://doi.org/10.1016/j.coph.2014.09.006

Roux, S., Hallam, S. J., Woyke, T., and Sullivan, M. B. (2015). Viral dark matter and virus host interactions resolved from publicly available microbial genomes. eLife, 4(January), 120. http://doi.org/10.7554/eLife.08490

Saitou, N., and Nei, M. (1998). The neighbor-joining method: a new method for reconstructing phylogenetic trees. Molecular Biology and Evolution, 4(4), 406425. Retrieved from https://academic.oup.com/mbe/article/4/4/406/1029664

Samson, J. E., Magadán, A. H., Sabri, M., and Moineau, S. (2013). Revenge of the phages: Defeating bacterial defences. Nature Reviews Microbiology, 11(10), 675687. http://doi.org/10.1038/nrmicro3096

Sastry, A., Monk, J., Tegel, H., Uhlen, M., Palsson, B. O., Rockberg, J., and Brunk, E. (2017). Machine learning in computational biology to accelerate high-throughput protein expression. Bioinformatics. http://doi.org/10.1093/bioinformatics/btx207

Schmelcher, M., Donovan, D. M., and Loessner, M. J. (2012). Bacteriophage endolysins as novel antimicrobials. Future Microbiology, 7(10), 11471171. http://doi.org/10.2217/fmb.12.97

Scholl, D., Kieleczawa, J., Kemp, P., Rush, J., Richardson, C. C., Merril, C., Molineux, I. J. (2004). Genomic Analysis of Bacteriophages SP6 and K1-5, an Estranged Subgroup of the T7 Supergroup. Journal of Molecular Biology, 335(5), 11511171. http://doi.org/10.1016/j.jmb.2003.11.035

Simmonds, P., Adams, M. J., Benk, M., Breitbart, M., Brister, J. R., Carstens, E. B., Zerbini, F. M. (2017). Consensus statement: Virus taxonomy in the age of metagenomics. Nature Reviews Microbiology, 15(3), 161168. http://doi.org/10.1038/nrmicro.2016.177

Simon, E. J., Reece, J. B., Dickey, J. (2010). Campbell Essential Biology: Fourth Edition. Benjamin Cummings. ISBN 0321772601.

Stern, A., Mick, E., Tirosh, I., Sagy, O., and Sorek, R. (2012). CRISPR targeting reveals a reservoir of common phages associated with the human gut microbiome. Genome Research, 22(10), 19851994. http://doi.org/10.1101/gr.138297.112

Steven, A. C., Trus, B. L., Maize, J. V, Unser, M., Parry, D. A. D., Wal, J. S., Studier, F. W. (1988). Molecular substructure of a viral receptor recognition protein The gp17 tail fiber of bacteriophage T7, 0, 351365.

Studier, F. W. (1975). Gene 0.3 of bacteriophage T7 acts to overcome the DNA restriction system of the host. Journal of Molecular Biology, 94(2), 283295. http://doi.org/10.1016/0022-2836(75)90083-2

Susskind, M. M., and Botstein, D. (1980). Superinfection exclusion by prophage in lysogens ofSalmonella typhimurium. Virology, 100(1), 212216. http://doi.org/10.1016/0042-6822(80)90571-1

Suzek, B. E., Wang, Y., Huang, H., McGarvey, P. B., and Wu, C. H. (2015). UniRef clusters: A comprehensive and scalable alternative for improving sequence similarity searches. Bioinformatics, 31(6), 926932. http://doi.org/10.1093/bioinformatics/btu739

Tabor, S., and Richardson, C. C. (1989). Selective inactivation of the exonuclease activity of bacteriophage T7 DNA polymerase by in vitro mutagenesis. Journal of Biological Chemistry, 264(11), 64476458. Retrieved from http://www.ncbi.nlm.nih.gov/pubmed/2703498

Tortora, G. J., Funke, B. R., and Case, C. L. (2013). Microbiology: an introduction, Pearson, Boston, 11th edition. ISBN 9780321733603 (student ed.).

Trevors, J. T. (1999). Evolution of gene transfer in bacteria [Review]. World J Microbiol Biotechnol, 15(1), 17. http://doi.org/10.1023/A:1008830914223

Vihinen, M., Torkkila, E., and Riikonen, P. (1994). Accuracy of protein flexibility predictions. Proteins: Structure, Function, and Bioinformatics, 19(2), 141149. http://doi.org/10.1002/prot.340190207

Vinga, S., and Almeida, J. (2003). Alignment-free sequence comparison - A review. Bioinformatics, 19(4), 513523. http://doi.org/10.1093/bioinformatics/btg005

Weigele, P. R., Scanlon, E., and King, J. (2003). Homotrimeric, beta-stranded viral adhesins and tail proteins. Journal of Bacteriology, 185(14), 40224030. http://doi.org/10.1128/JB.185.14.4022-4030.2003

Weinbauer, M. G. (2004). Ecology of prokaryotic viruses. FEMS Microbiology Reviews, 28(2), 127181. http://doi.org/10.1016/j.femsre.2003.08.001

Weitz, J. S., Hartman, H., and Levin, S. A. (2005). Coevolutionary arms races between bacteria and bacteriophage. Proceedings of the National Academy of Sciences, 102(27), 95359540. http://doi.org/10.1073/pnas.0504062102

Wilkins, M. R., Gasteiger, E., Bairoch, A., Sanchez, J. C., Williams, K. L., Appel, R. D., and Hochstrasser, D. F. (1999). Protein identification and analysis tools in the ExPASy server. Methods Molecular Biology, 112(February), 531552.

Wu, G. A., Jun, S.-R., Sims, G. E., and Kim, S.-H. (2009). Whole-proteome phylogeny of large dsDNA virus families by an alignment-free method. Proceedings of the National Academy of Sciences, 106(31), 1282612831. http://doi.org/10.1073/pnas.0905115106

Yang, K. K., Wu, Z., Bedbrook, C. N., and Arnold, F. H. (2017). Learned Protein Embeddings for Machine Learning. Bioinformatics.

Yao, G. W., Duarte, I., Le, T. T., Carmody, L., LiPuma, J. J., Young, R., and Gonzalez, C. F. (2017). A Broad-host-range Tailocin from Burkholderia cenocepacia, 83(10), 117.

Youle, M. (2017). Thinking like a phage.

Yu, Z. G., Chu, K. H., Li, C. P., Anh, V., Zhou, L. Q., and Wang, R. W. (2010). Whole-proteome phylogeny of large dsDNA viruses and parvoviruses through a composition vector method related to dynamical language model. BMC Evolutionary Biology, 10(1). http://doi.org/10.1186/1471-2148-10-192

Zhang, Q. (2016). Strategies for Identifying the Optimal Length of K- mer in a Viral Phylogenomic Analysis using Genomic Alignment-free Method.

Zhou, Q., and Liu, J. S. (2008). Extracting sequence features to predict protein - DNA interactions: A comparative study. Nucleic Acids Research, 36(12), 41374148. http://doi.org/10.1093/nar/gkn361

Zhou, S. T., Liu, R., Zhao, X., Huang, C. H., and Wei, Y. Q. (2011). Viral proteomics: The emerging cutting-edge of virus research. Science China Life Sciences, 54(6), 502512. http://doi.org/10.1007/s11427-011-4177-7

Zillig, W., Fujiki, H., Blum, W., Janekovi, D., Schweiger, M., Rahmsdorf, H., Hirsch-Kauffmann, M. (1975). In vivo and in vitro phosphorylation of DNA-dependent RNA polymerase of Escherichia coli by bacteriophage-T7-induced protein kinase. Proceedings of the National Academy of Sciences of the United States of America, 72(7), 250610. Retrieved https://www.ncbi.nlm.nih.gov/pubmed/1101258

Zinder, N. D. (1958). Lysogenization and superinfection immunity in Salmonella. Virology, 5(2), 291326. http://doi.org/10.1016/0042-6822(58)90025-4

APPENDIX A

Extra figures and tables

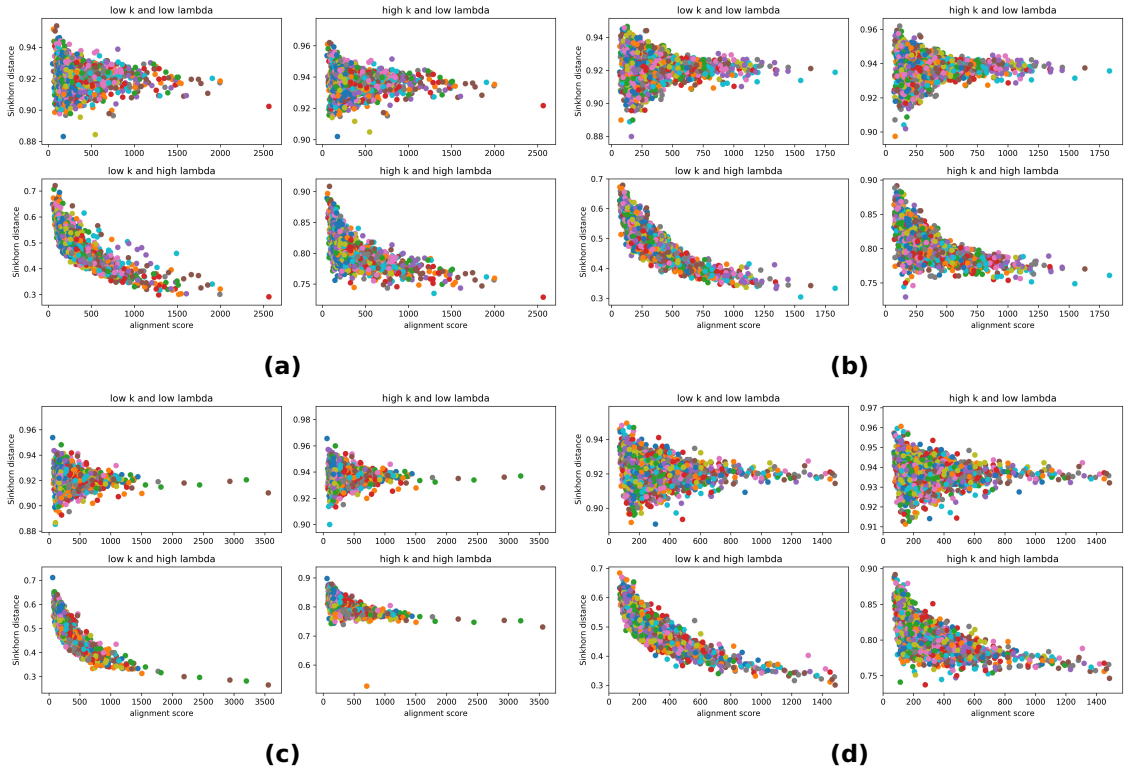


Figure A.1: Relationship between Sinkhorn distances and alignment scores for combinations of low and high values of k and λ, using one hundred proteins sampled at random from the dataset.

Four of nine extra analyses of sampling hundred proteins from the collected dataset. These proteins were compared in a pairwise manner using both optimal transport and local alignment. The figures above show the alignment score in function of the Sinkhorn distance for pairwise comparison of the proteins. In the upper-left plots, λ was equal to 0.1 and k equal to 3. In the upper-right plots, λ was equal to 0.1 and k equal to 15. In the bottom-left plots, λ was equal to 30 and k equal to 3. In the bottom-right plots, λ was equal to 30 and k equal to 15.

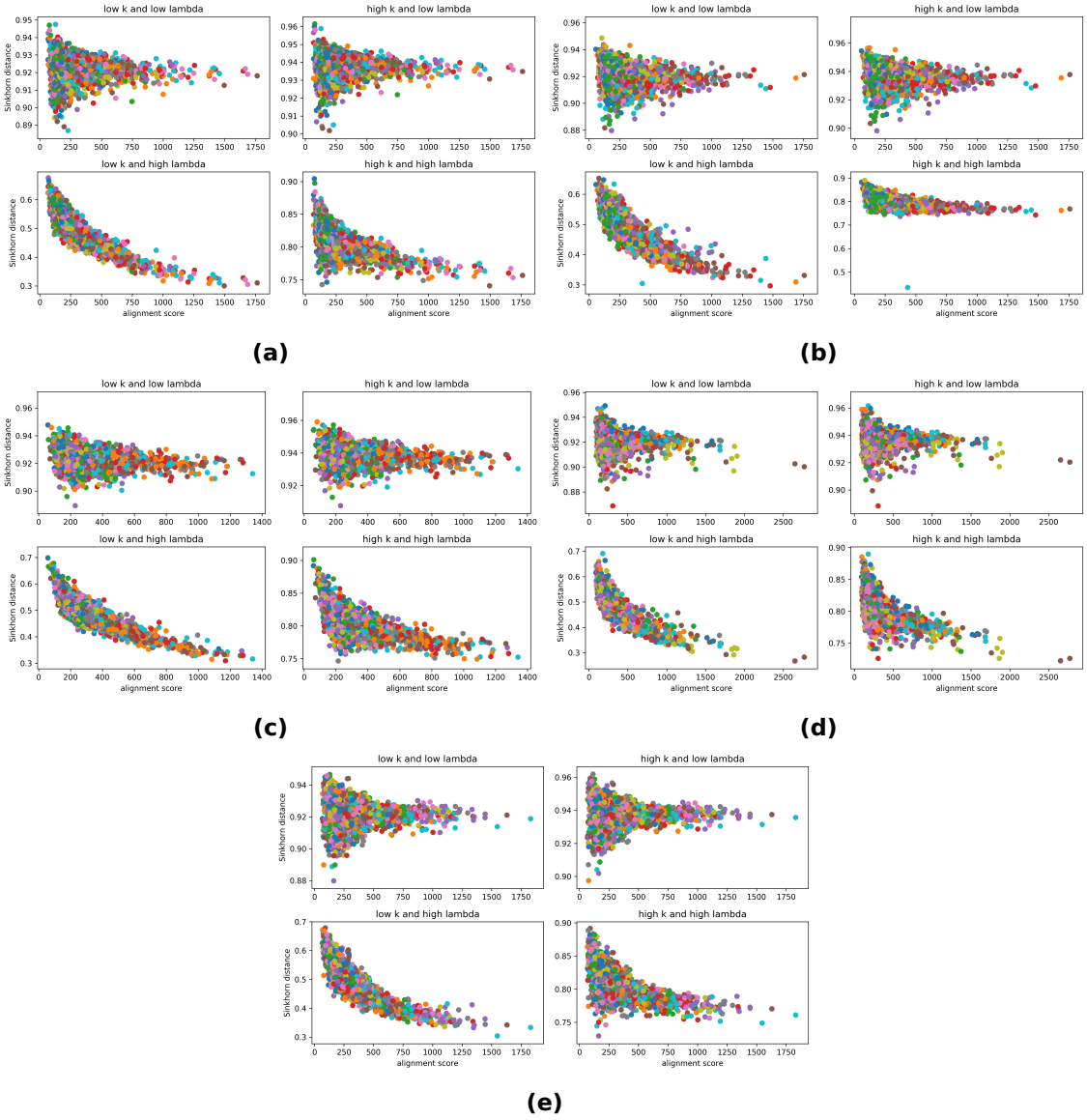


Figure A.2: Relationship between Sinkhorn distances and alignment scores for combinations of low and high values of k and λ, using one hundred proteins sampled at random from the dataset, repeated nine times.

Four of nine extra analyses of sampling hundred proteins from the collected dataset. These proteins were compared in a pairwise manner using both optimal transport and local alignment. The figures above show the alignment score in function of the Sinkhorn distance for pairwise comparison of the proteins. In the upper-left plots, λ was equal to 0.1 and k equal to 3. In the upper-right plots, λ was equal to 0.1 and k equal to 15. In the bottom-left plots, λ was equal to 30 and k equal to 3. In the bottom-right plots, λ was equal to 30 and k equal to 15.

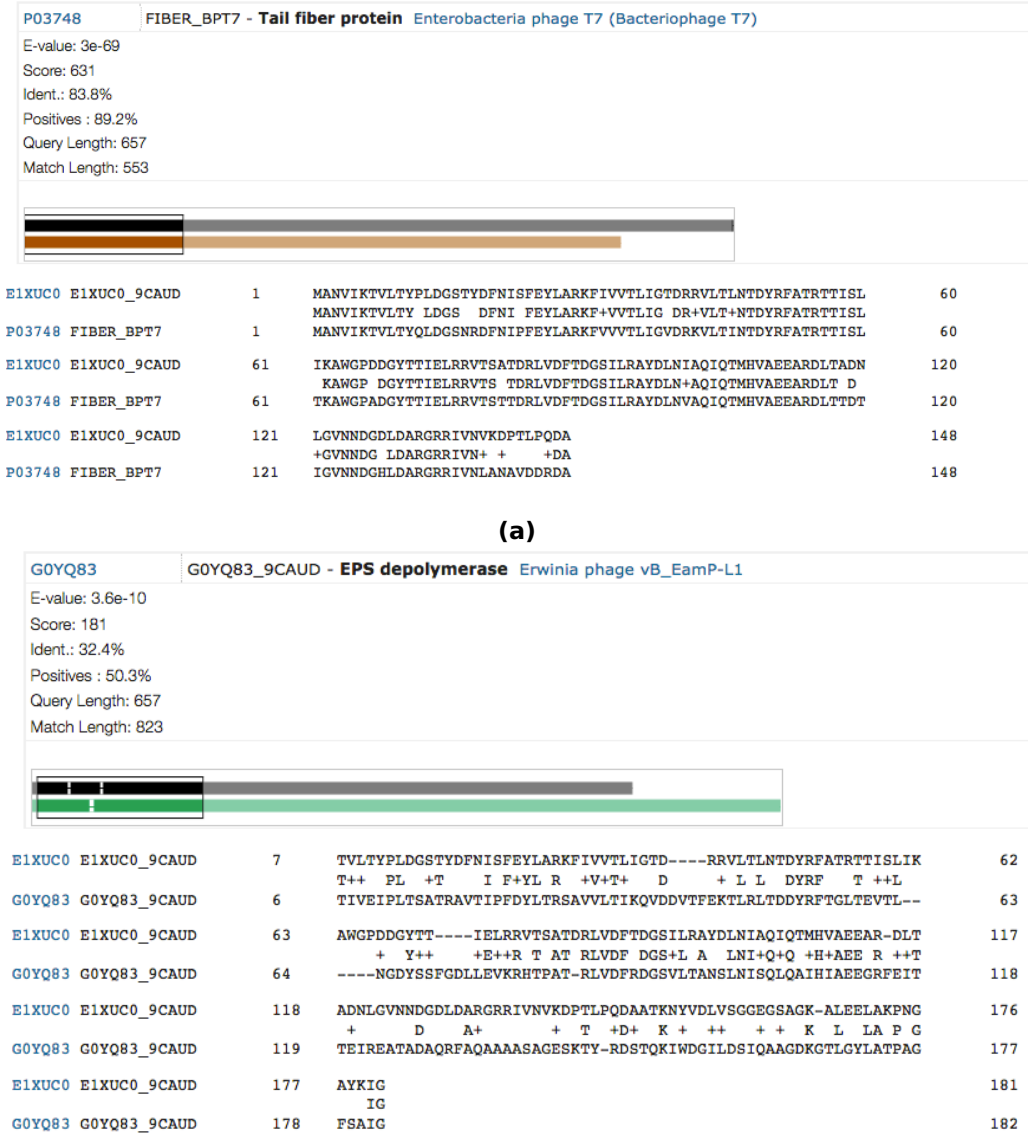


Figure A.3: Pairwise sequence aligments of the tail fiber protein of Salmonella phage Vi06 with the tail fiber protein of Enterobacteria phage T7 (a) and the EPS depolymerase of Erwinia phage vB_EamP-L1 (b).

Output of the pairwise alignments of of the tail fiber protein of Salmonella phage Vi06 with the tail fiber protein of Enterobacteria phage T7 (a) and the EPS depolymerase of Erwinia phage vB_EamP-L1 (b) using UniProt. Both alignments only span across the N-terminal domain, indicating that the C-terminal domain is responsible for the difference in host specificity.

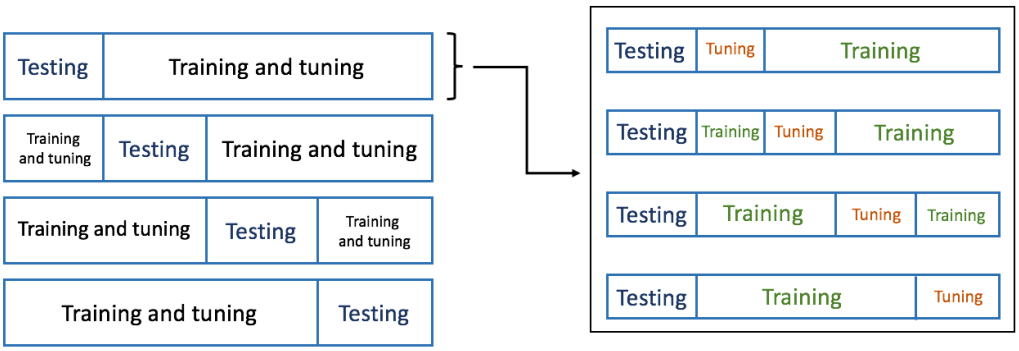


Figure A.4: Visualization of a nested four-fold cross-validation.

In a nested four-fold cross-validation scheme, an outer loop divides the dataset in four parts. Three parts are used for training and tuning, while one part is used for testing. This division can be repeated four times, where each time testing is done on a different part of the dataset. In nested cross-validation, every iteration in the outer loop consists of an inner loop. In this inner loop, the three parts used for training and tuning are split up into four equal parts. Similar to the outer loop, the inner split can be repeated four times. In each iteration of this inner loop, the tuning is done on a different part of the data.

APPENDIX B

Python code used in methods

```python
import numpy as np
import datetime as dt
from numba import jit
import scipy as sp
from Bio import SeqIO

def count_kmers(read, k, counts):
    """
    read is a sequence, k is the k-mer length, counts is a dictionary were all
    k-mers and their counts will be stored in.
    """
    num_kmers = len(read) - k + 1
    for i in range(num_kmers):
        kmer = read[i:i+k]
        if kmer not in counts:
            counts[kmer] = 0
        counts[kmer] += 1
    return counts

@jit
def compute_optimal_transport(M, r, c, lam, epsilon=1e-8):
    """
    Computes the optimal transport matrix and Slinkhorn distance using the
    Sinkhorn-Knopp algorithm

    Inputs:
        - M : cost matrix (n x m)
        - r : vector of marginals (n, )
        - c : vector of marginals (m, )
        - lam : strength of the entropic regularization
        - epsilon : convergence parameter

    Output:
        - P : optimal transport matrix (n x m)
        - dist : Sinkhorn distance
    """
    n, m = M.shape
    P = np.exp(- lam * M)
    P /= P.sum()
```

```
    u = np.zeros(n)
    # normalize this matrix
    while np.max(np.abs(u - P.sum(1))) > epsilon:
        u = P.sum(1)
        P *= (r / u).reshape((-1, 1))
        P *= (c / P.sum(0)).reshape((1, -1))
    return P, np.sum(P * M)

def optimal_transport_phages(org_lst, k, lam):
    """
    input:
        - a list of phage names in the reference proteomes to compare
        - k: length of k-mers
        - lam: value for entropic regularization parameter

    output: a matrix of pairwise Sinkhorn distances
    """

    distance_matrix = np.zeros((len(org_lst), len(org_lst)))

    # first phage
    for i in range(0,len(org_lst)-1):

        phage1 = org_lst[i]
        kmers_phage1 = {}

        # get kmers from protein file
        proteomes = open(r'phage_reference_proteomes.fasta')
        for seq in SeqIO.parse(proteomes, 'fasta'):
            if phage1 in seq.description:
                kmers_phage1 = count_kmers(str(seq.seq), k, kmers_phage1)

        # get probability distribution
        r = [kmers_phage1[x] for x in kmers_phage1.keys()] # counts
        r_norm = [float(i)/sum(r) for i in r]

        # code kmers as numbers for later use
        coded1 = [[ord(y) for y in x] for x in list(kmers_phage1.keys())]
        del kmers_phage1

        # second phage
        for j in range(i+1, len(org_lst)):
            phage2 = org_lst[j]
            kmers_phage2 = {}

            # get kmers from protein file
            proteomes = open(r'phage_reference_proteomes.fasta')
            for seq in SeqIO.parse(proteomes, 'fasta'):
                if phage2 in seq.description:
```

```
                    kmers_phage2 = count_kmers(str(seq.seq), k, kmers_phage2)

            # get probability distributions
            c = [kmers_phage2[x] for x in kmers_phage2.keys()] # counts
            c_norm = [float(i)/sum(c) for i in c]

            # build cost matrix M
            coded2 = [[ord(y) for y in x] for x in list(kmers_phage2.keys())]
            del kmers_phage2
            M = sp.spatial.distance.cdist(coded1, coded2, metric='hamming')

            # optimal transport
            P, d = compute_optimal_transport(M, r_norm, c_norm, lam=lam)
            del P
            del M

            distance_matrix[i,j] = d

    return distance_matrix

# Tree construction using optimal transport for seven phages present in
    phage_reference_proteomes.fasta
now = dt.datetime.now()
seven_phages = ['Pseudomonas phage YuA', 'Pseudomonas phage F116', 'Bdellovibrio
    phage phiMH2K', 'Mycobacterium phage Wonder', 'Stenotrophomonas phage Smp131', '
    Vibrio phage Vc1', 'Streptomyces phage phiSASD1']
distance_matrix = optimal_transport_phages(seven_phages, 3, 30)
print(dt.datetime.now()-now)

distance_matrix = np.loadtxt('distmatrix_7phagesk15.txt')
for i in range(0, distance_matrix.shape[0]-1):
    for j in range(i+1, distance_matrix.shape[1]):
        distance_matrix[j,i] = distance_matrix[i,j]
np.fill_diagonal(distance_matrix, 0)
seven_labels = ['PseudomonasYuA', 'PseudomonasF116', 'BdellovibrioPhiMH2K', '
    MycobacteriumWonder', 'StenotrophomonasSmp131', 'VibrioVc1', '
    StreptomycesPhiSASD1']
dm = DistanceMatrix(distance_matrix, seven_labels)
tree = nj(dm)
ts = ete3.TreeStyle()
ts.show_branch_length = True
ete3.TreeNode.from_skbio(tree).render("%%inline", tree_style=ts)
```