Can phages replace pesticides? - Individual Project in Bioinformatics Report

Ida Kups

ABSTRACT

Potatoes are one of the most important carbohydrate sources worldwide, but a significant part of the harvest is lost due to bacterial diseases such as blackleg or soft rot caused by Pectiobacteriae and Dickeya. Research is currently underway to develop a phage cocktail against them. There are many things to consider during a process. One of the many questions that needs to be answered are what the Pectobacteriae and Dickeya resistant to and how the disease is spread. The goal of this project was to investigate them on a small group of a bacteria isolated from an infected potato tubers collected by farmers across Denmark.

Introduction

With potatoes being one of the most widely cultivated carbohydrates source and dietary staple for millions of people worldwide, it is crucial to ensure a bountiful harvest and reduce losses at every step of the process. At the moment, more than one fifth of the yearly potato yield is being lost due to diseases caused by viruses, fungi and bacteria[1]. Developing successful techniques to reduce the loss rate could save millions of tons of potatoes from being wasted. In 2022, yields decreased by 6% compared to 2021, and this trend is expected to continue in 2023[2]. These unfavorable economic and environmental conditions make the reduction of losses an even more pressing issue.

Blackleg and tuber soft rot are among the bacterial diseases having the biggest economical impact on potato production. They are caused by the different species of bacteria belonging to the *Pectobacterium* and *Dickeya* genera. There are several ways how the bacteria are spread including soil transfer or infected seed tubers[3]. In the latter case losses can occur both due to the direct damage of tubers as well as increased number of soft rot and blackleg cases in the next growing season. Currently, there are no protection protocols that can effectively prevent tuber infection, and current strategies mainly involve avoiding contamination by using a seed certification system, proper plant nutrition, breeding for resistance, and physical or chemical seed treatment[1]. However, none of these approaches provide satisfactory results, leading to the a constant search for alternative methods to address the problem. The use of bacteriophages appears to be the one of the most promising.

Bacteriophages are viruses capable of infecting and lysing bacterial cells. When applied to infected tubers, they could provoke their target death (i.e. eliminate the bacteria) while being completely safe for the plant. This approach offers several potential advantages, such as host-specificity which means phages are unable to infect any other organisms and therefore completely harmless to humans. Their persistence and self-replication ability allows one time induction to the environment and the biodegradability makes the approach suitable for all types of crops, including ecological ones. As they are safe to eat and have no impact on smell or taste of the tubers[4] phages can be applied on tubers also after harvest in a controlled conditions[5].

One of the main problems of the strategy is the fact that the target bacteria acquire resistance very quickly. To address this issue, it is necessary to understand exactly what the bacteria are resistant to and how they acquire immunity. To fight the phages and other mobile genetic elements bacteria developed their own, primitive immune system. It is build up of two components localized next to each other in the genome- genes encoding Cas proteins and Clustered Regularly Interspaced Short Palindromic Repeats (CRISPRs)[6]. The system targets foreign nucleic acids invading the cell and cuts them into pieces, preventing infection. Cas genes encode proteins belonging to the nucleases family. They act like a molecular scissors that cut nucleic acids. CRISPR array is made up of a series of short repeated sequences separated by unique spacers that are usually fragments of the genetic information of the invaders with whom the bacteria (or its ancestor) had contact. During reinfection, thanks to the information encoded in spacers, the bacteria has an advantage over the invader as the immune system is activated. Cas proteins are expressed and spacers are transcribed into long precursor CRISPR RNA (pre-crRNA) which is then transformed into mature crRNA. Together, they form the complex designed to recognize and destroy certain type of nucleic acid. As crRNAs are 100% compatible with the predators genetic material, they serve as a Cas proteins navigators, enabling the precise cut of the invaders nucleic acid. For CRISPR systems targeting DNA, there are is an additional element: short activating sequence (PAMs) separated by protospacer just next to the target in bacterial genome[7]. It is used for determination of origin of the nucleic acids and makes it possible to distinguish between one's own and foreign genome. Protospacer is a short 2-6 bp DNA sequence within which the Cas nuclease cuts foreign genome. Unless protospacer is flanked by two PAMs the cut is not possible. As PAM is only present in target genome there is no risk of self targeting[8].

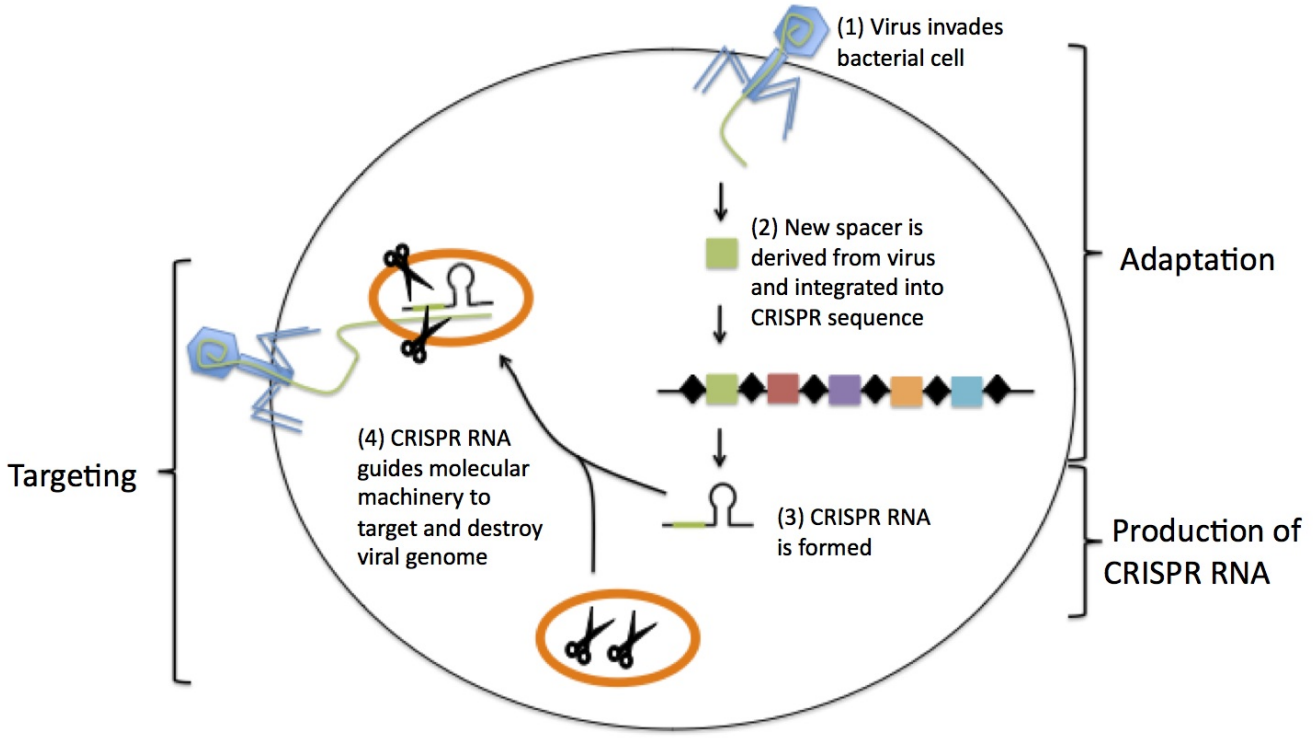

Figure 1. CRISPR mediated immunity mechanism. Retrieved from: *https://sitn.hms.harvard.edu/flash/2014/crispr-a-game-changing-genetic-engineering-technique/*

New spacers are added in the process involving Cas1 and Cas2 proteins cutting both DNA strands to enable their integration[9]. They are usually acquired on one end of the array, after the AT rich leader sequence. Although there is no specific number of the spacers in the CRISPR array, it is rare for its length to exceed 30. This is likely because a high spacer content can reduce the system's effectiveness. It becomes sub optimal for bacteria to keep the spacers targeting the invader they had not encountered for a long time as it may slow down immune reaction against the new "predator" that have just appeared in the environment. Therefore spacers are lost, especially in the presence of the selective pressure caused for example by the introduction of the unknown phage to the field. Interestingly, the system does not get rid of the oldest spacers, but instead removes ones in the middle[9]. There are many reasons for this, such as the fact that when being periodically exposed to the same phages, spacers remain useful, even when being relatively old[10]. Another reason could be the so called primed adaptation, which is the second possible way of spacer acquisition. In this process, the crRNP complex recognizes the protospacer target and "builds" the new spacer from the neighbouring DNA sequence. Priming occurs when, for some reason, there is no perfect match between spacer and its target[11]. As mutations accumulate over time, old spacers are usually no longer matching the target with 100%. Because of the priming, even imperfect, mutated spacers may remain useful in the CRISPR array and getting rid of them would be unfavorable. As for the trailer end spacers, they seem to be lost less frequently than it would be expected. The reason for that is not yet known, but it is probably hidden in a recombination mechanism rather than in their role in immune response.

Because spacers farthest from the leader sequence are the oldest, they are shared between relatively wide range of strains, while the closest and newest are usually unique. It quickly became clear that their order can give the information about when the bacteria had the contact with the invader[9]. Although there are some suspicions, it is not clear what parts of invaders genome are more likely to be chosen. The mechanism is definitely complicated and there are many potential targets, therefore the probability of acquiring exactly the same spacers by bacteria in independent events is close to zero. This property makes spacers and their order a perfect tool to track evolutionary relations between the species and event the specific bacteria.

To get the overall idea about how the bacterial immune system works and what information it could provide to the researchers one must not only focus on the system itself, but also on the mobile genetic elements it actually targets. There are three main groups of potential invaders: bacteriophages, prophages and plasmids. As bacteriophages have already been discussed above, I will jump straight to the second group. Prophages are simply the bacteriophages incorporated into the host genome. Integration is possible thanks to the integrase genes encoding the enzyme responsible for cutting the host genome and inserting the phage

inside the break. Prophages are using the host genetic machinery for replication and they remain latent in the genome until being exposed to one of the activation factors. That provokes their entering into the lytic cycle when the cell undergoes lysis and new phages are released. The third possible spacer target are plasmids. As well as phages, they are usually mobile circular genetic elements. They replicate independently and do not integrate into host genome. They are spread between bacteria in a cell-to-cell process called conjugation[12], and often carry information that is useful, but not necessary for bacteria survival such as antibiotic resistance genes.

I was part of the project intending to create a phage cocktail able to fight bacteria causing soft rot and blackleg,that could be applied onto potato tubers to reduce the waste caused by these two diseases. As I had very little time, my role was limited to answering two questions:

- What the bacteria causing the blackleg and soft rot in the Danish potato crops are resistant to and what mobile genetic elements they already had the contact with?
- What are the evolutionary relations between the bacteria collected form the different fields in different parts of Denmark?

As long as the potential benefits of answering the first question are relatively obvious (knowing what bacteria are resistant to prevents us from trying to use it as a bactericide), the second one is slightly less straightforward. Figuring out the relations between different bacteria strains and combining it with the metadata telling us where and from whose field they were collected, would enable understanding how the soft rot and blackleg diseases are spread. This information in turn, would help to limit the number of infections. As most of the bacteria belong either to *Pectobacterium* or *Dickeya* genera it was clear that assigning them to one of these groups will not be enough. To answer the second question, I had to to decode the species and then even find the relations between its different representatives. As the differences at this level of kinship are very small, it makes it very tricky to distinguish the important from the "noise" caused for example by the sequencing errors prophages/plasmids which are thousands of kbp long insertions in the host genomes.

Methods and results

Side note : I have decided to merge Methods and Results section as I believe that makes my report clear and easy to understand. I divided my work into several steps, many of which follows from the previous one. I will describe the methods and results separately for each step keeping the chronological order

To answer the questions I was equipped with 59 assembled bacterial genomes isolated from soft rot/ blackleg infected potatoes collected by the Danish farmers from the yields across the whole country. Bacteria were named from J1 to J59 and I had no further information about their species and place of collection. Additionally, I received 48 assembled phage genomes that have been isolated with the intention of using them as a bactericides.

Step 1

Overview: The goal of the first step was to extract the spacers from the bacterial genomes.
Methods: I have decided to use two tools to do it:

- CrisprCasTyper (CCT)[13]: widely available software for identifying the whole CRISPR-Cas systems in the genome. It uses minced algorithm and "blasting" against known repeats stored in a database to find the spacer-repeat arrays. It only performs at the "alphabetic level" simply scanning the genome to find repeated sequences of the same length separated by unique strings of letters of similar length.
  advantages: clear FASTA output ready to use in next steps
  disadvantages: may have a problems with distinguishing CRISPRs from other arrays as it only looks for a patterns in a genome. Tends to shorten the array when there are some mutations in the middle spacers (mainly insertions and deletions)

- CrisprDetect (CD)[14]: program written strictly for detection of CRISPR arrays. It uses biological information about the arrays and places great emphasis on assigning directionality.
  advantages: deals better with mutations inside the spacers, assigning directionality and identifying short CRISPR arrays
  disadvantages: complicated text output and no commonly available tools for converting it into FASTA format .

Results: I performed spacer extraction for all bacterial genomes using both tools. Out of 59 sequences spacers were identified in 49 using CD and 50 using CCT. CCT did not find the spacers in 9 bacteria and CD did not find any in the same 9 plus in J14.
bacteria without spacers when using CCT: {J5, J15, J23, J46, J50, J51, J53, J55, J57}
bacteria without spacers when using CD {J5, J15, J23, J46, J50, J51, J53, J55, J57, **J14**}

Step 2

Overview: In the next step I tried to find out whether any of the spacers target our isolated phages.
Methods: I used two tools to perform the analysis:

- BLAST The most popular local alignment tool. It finds similarities between sequences at a nucleotide level.

- SpacePHARER[15] tool made specifically for host-phage identification using CRISPR spacers. In order to improve the alignment quality it first translates the sequences using all six ORFs and then compare the phage vs spacer on the amino acids level.

I used BLAST to test CCT spacers against the phages and SpacePHARER to check both CCT & CD outputs.
Results: I got no hits for any of the performed analysis.

Step 3

Overview: In this step I wanted to identify the spacers targets.
Methods: I first blasted each spacer sequence against whole NCBI database to see if I get any interesting hits against the mobile genetic elements.
I also wanted to find out if any of the spacers targets plasmids/ prophages hidden in our bacterial genomes. I first wrote a script (*extract_spacers.py* that extracted the spacers from CRISPRDetect output files and saved them into FASTA format, each CRISPR system separately. From now on I decided to limit the analysis only to CD output files as the results from both tools were similar and CD should in theory outperform CCT. Later, I blasted all the extracted spacers against all the bacterial genomes. I identified 'self-hits' (i.e. hits against the spacers in the genome the spacer was extracted from) and removed them using my own script *remove_hits.py*. In the next step I created a 59 maps of prophages and plasmids in the bacterial genomes that which could be later used to identify whether a certain spacer targets a sequence inside a plasmid/prophage in one of the bacteria. For identifying mobile genetic elements I used different tools:

- Prophages: VIBRANT[16](it was recommended to me as a tool with the best effort/result ratio). Result were saved to the excel file (*annotated_bacteria.xlsx*.

- Plasmids: It is way more complicated to identify plasmids than prophages as they do not differ that much from the host genome. All currently available tools are based either on homology search against a database, finding regions of high/low coverage in raw sequencing data or both methods combined. I first tried to use PlasmidFinder which is a homology tool for searching against plasmid database. As I got no hits (which was not really surprising as there is very limited number of plasmids that were isolated, sequenced and stored in a database) I knew it does not make sense to use any tools based on the homology search. The information about the contigs coverage was already included in the assembled sequence (depth parameter), so I have decided to write my own script (*find_plasmids.py*) to identify the plasmids, rather than use the existing tools. All contigs with depth <= 0.5x and >=5x were assigned as plasmids. Results were saved into pandas DataFrame.

After creating the maps, I performed analysis using the script I've written myself (*final.py*). It was reading in each BLAST hit for each bacteria separately and comparing it with a map to find whether the spacer target can be identified (i.e. matches a genome part assigned as prophage/plasmid). Only positive results (i.e. spacers with known target) were saved to Excel file (*SPACERS_known.xlsx)*)

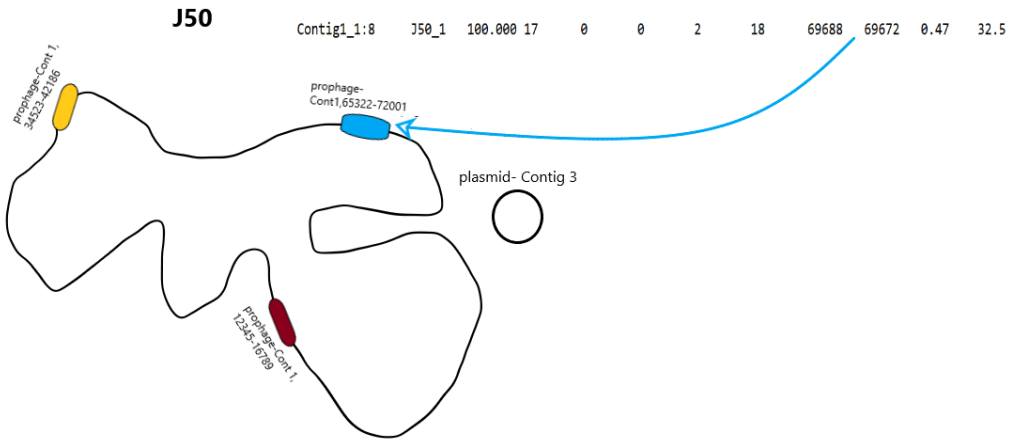

Figure 2. Mapping mechanism scheme. Each BLAST hit was compared to each bacteria map to identify the spacers target

Results: Out of 631 unique spacers I managed to identify the target for 152 (24%). *I have decided to update the description of the results with the data collected in step 4 to make the report more transparent* Scanning a whole NCBI database I got 0/64 phage hits for spacers in orange group (P. atrosepticum), 3/206 for P. parmentieri, 1/96 for P. brasiliense and three identified hits in total for the smaller groups.
For P. atrosepticum only 1 out of 64 spacers was not identified in a database at all (i.e. the sequence was not present in any genome stored in a database including all of the P. atrosepticum sequences), for P. parmentieri it was 12/206, for P. brasiliense it was 47/96. For J48 all, for J54 15/64, for J35 0, for J17/J18/J19 23/35, for J2/J3 25/39. Most of the hits were identified only in the genomes of the analyzed bacteria species (e.g. when analyzing bacteria from orange group 54/64 spacers were identified only in P. atrosepticum genomes stored in NCBI). Those that have been identified in more than one species, were usually, but not always, found in the other Pectobacterium (of course only for the spacers originally extracted from the Pectobacterium genomes).

Step 4

Overview: In step 4 I grouped the bacteria basing on their CRISPR spacers.
Methods: I used FASTA files from the previous step to find the relations between bacteria. I have written two scripts that helped me with the task:

- *compare_spacer_sets.py* - Used for comparing spacers between two systems. Gives an information about which spacers are shared and which are not. I used this script for finding big similarities (i.e. whole CRISPR shared systems).
- *find_same_spacers.py* - checks every spacer of a CRISPR system and returns an information whether it is shared by any of the bacteria and if yes, then at what position. Good at catching slight similarities between distantly related bacteria.

I later grouped the bacteria based on the information obtained from the scripts. It allowed me to distinguish three main groups and I used that information to construct the separate trees for each. Bacteria were assigned to the same group if they shared at least one spacer.

Results:

1. Bacteria grouped based on their CRISPR content

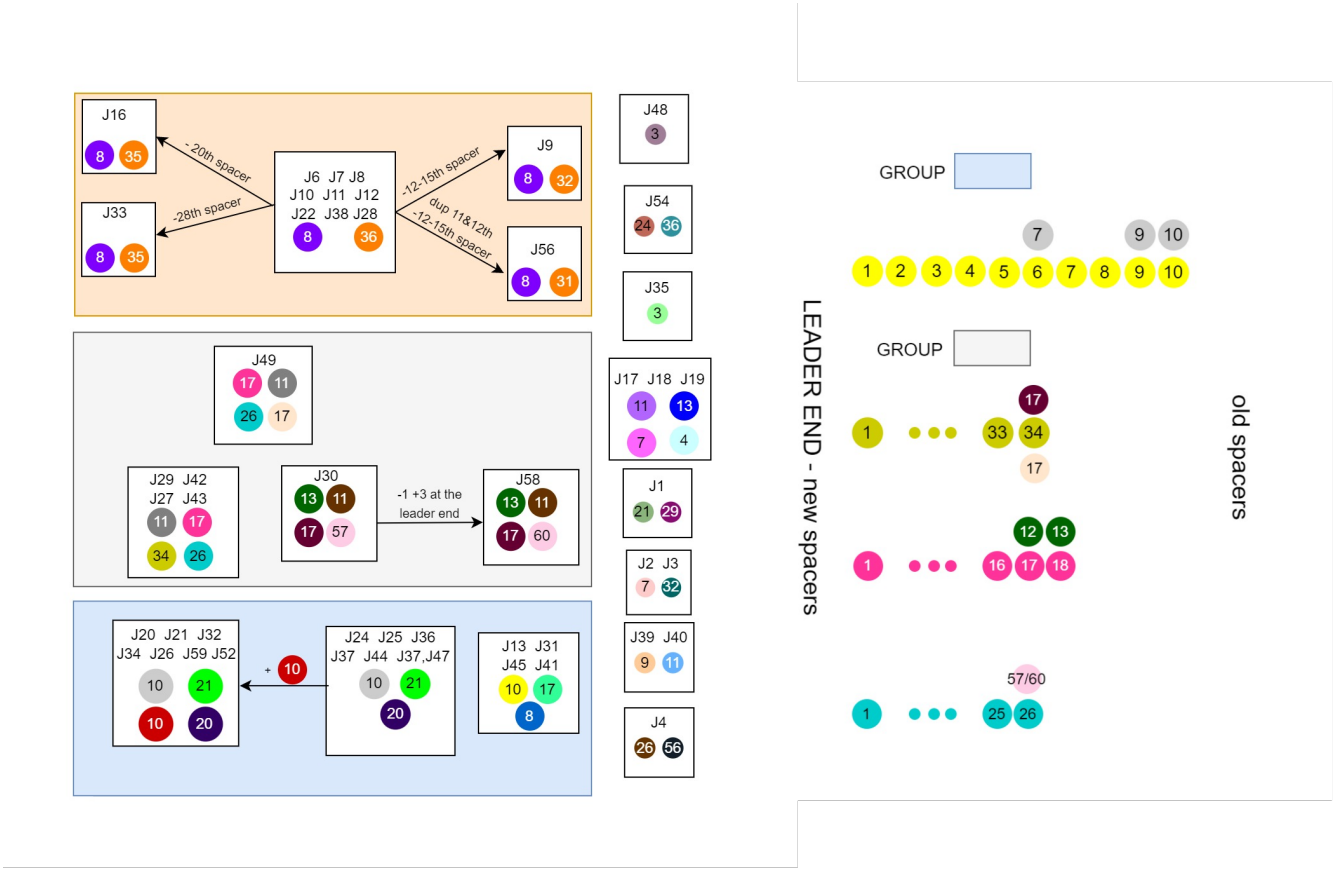

Figure 3. Bacteria grouped based on their CRISPR system. Dots indicate a CRISPR with unique color for each system. Number on a dot gives an information about the amount of spacers and the arrows indicate the direction of the evolution. To make the picture clearer, I have presented relationships between spacers in different CRISPR systems in a separate graphic. In this case the the dots refer to one spacer inside a CRISPR and number is a spacer position when counting from the leader end.

2. Phylogenetic data

I later compared my results with the phylogenetic data obtained by one of my colegues (it was prepared using the approach suitable for grouping the bacteria into species but not precise enough to catch intra-species dependencies).Our results overlapped in 97%. The only difference was that I grouped J39 and J40 as a separate subgroup while according to his data it belonged to P. atrosepticum species (orange group in my analysis). It is worth noting that bacteria mentioned above were flanking the left group border, which means that even with the analysis aimed to only distinguish the species they appeared to differ from the rest of the group which could explain completely non-overlapping CRISPR systems.

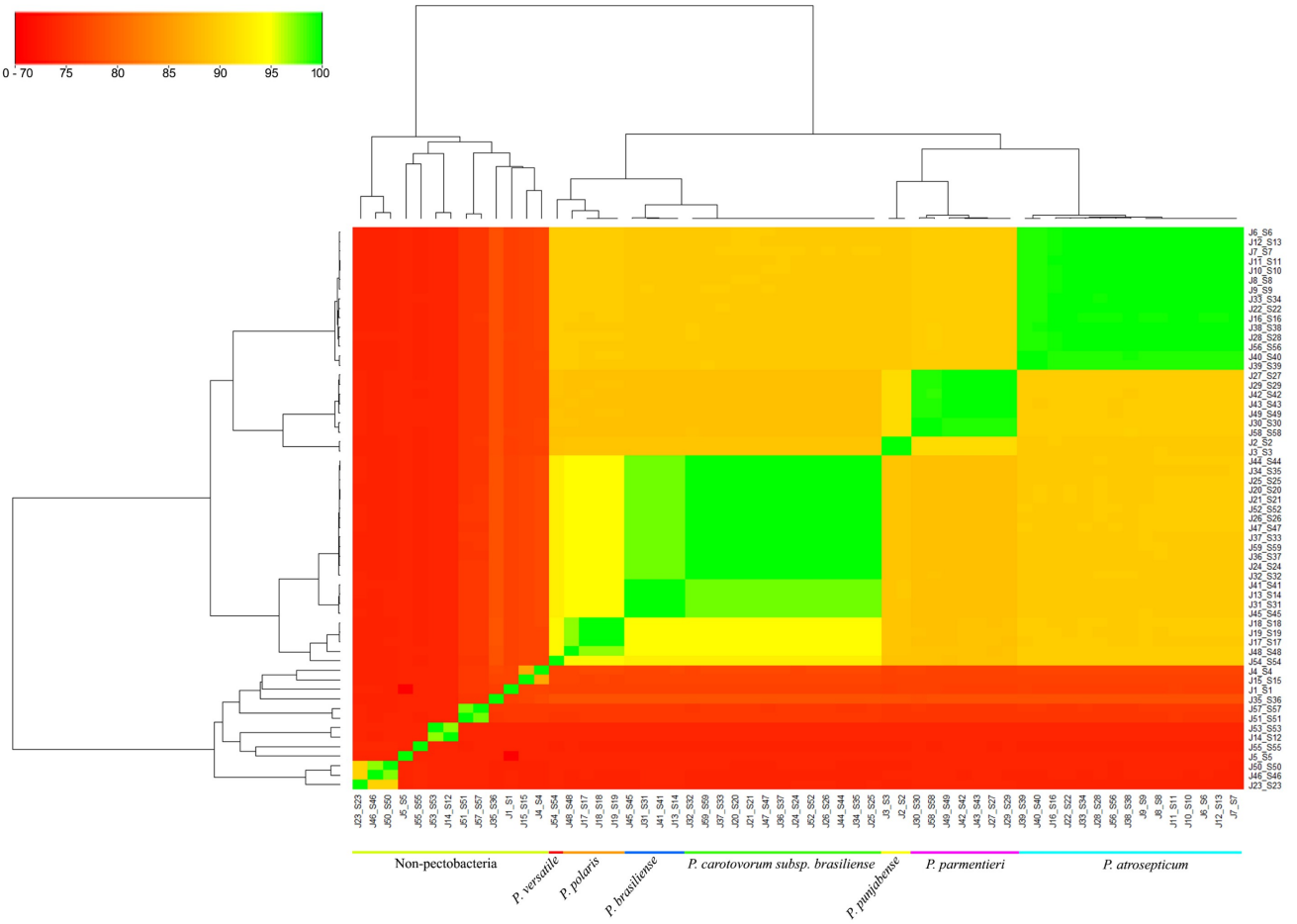

Figure 4. groups with assigned species. Prepared by one of my colleagues.

3. Phylogenetic tree based on the CRISPR information.

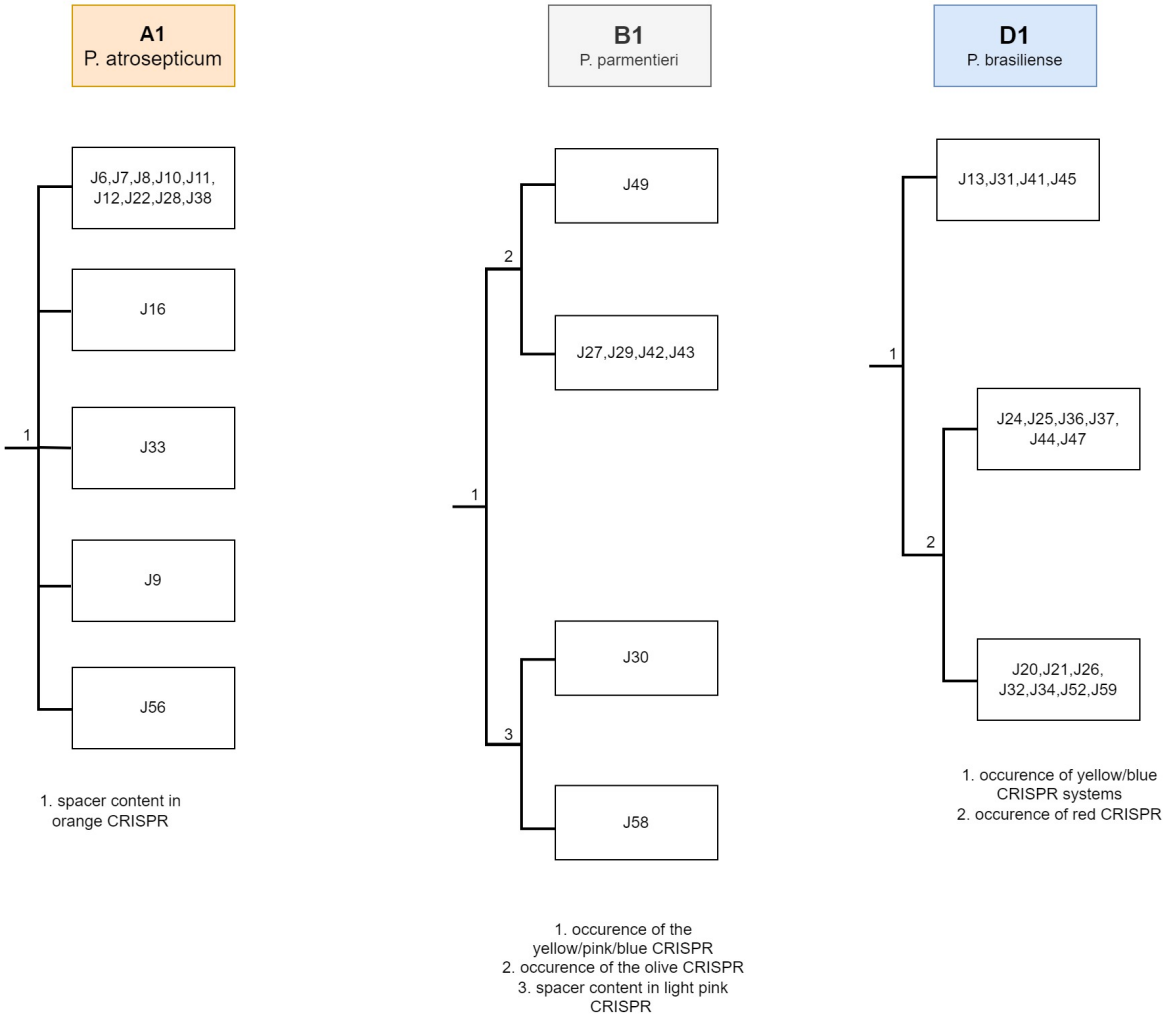

Figure 5. Trees build up on the CRISPR information separately for each big group. Colorful rectangle above the tree refers to the group it represents while the letter code and species name is connected to the phylogeny data I received

Step 5

Overview: As some of the tree leaves contained big bacteria groups, I have decided to update the trees I created with the information about the prophage content of each of the individuals. As, in contrast to the spacers, same prophages can be incorporated independently into different genomes, especially if bacteria originate from the same regions, I made the assumption that information on CRISPR overrides information on prophages.

Methods: I first used script *compare_prophages.py* to extract prophage sequences (I found using VIBRANT in the previous step) from bacterial genomes and save them into separate files with prophage ID number serving as filename. I later performed all possible combinations of pairwise alignments against prophages and clustered those who had more that 97% similarity together assuming that the differences are not statistically significant. I kept using the for grouping the bacteria based on their phage content. After I performed grouping I was informed that VIBRANT tends to miss the prophages in some genomes, even though it found them in the others. Hence, to correct the analysis I BLASTed all prophage genomes against a database build from bacterial sequences. I wrote a script *prophage_map.py* to analyze whether a hit is significant or not. It finds all the

hits targeting the bacteria and decides on their importance based on the two conditions. If % identity is greater than 99 and alignment length covers at least 95% of the prophage sequence, then start & stop positions, together with the phage ID are added to the pandas DataFrame (separate for each bacteria). All 59 dataframes are later converted into excel sheets and saved to the Excel file. I corrected the groups based on the information stored in the Excel sheets and constructed an updated tree incorporating the prophage content information into the tree build in the previous step. I also repeated step 3 using an updated prophage information.

Results

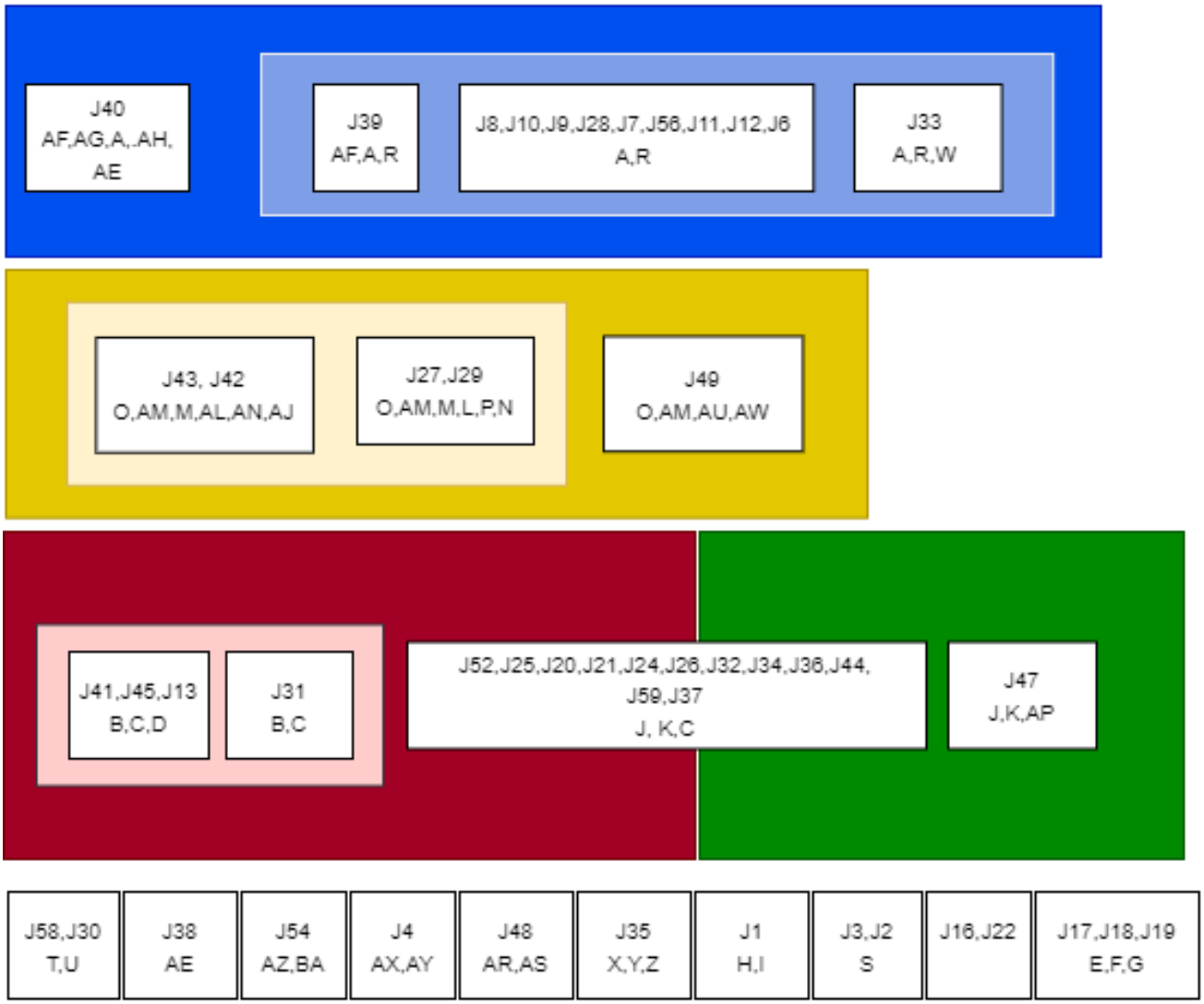

Figure 6. Bacteria grouped based on their prophage content.

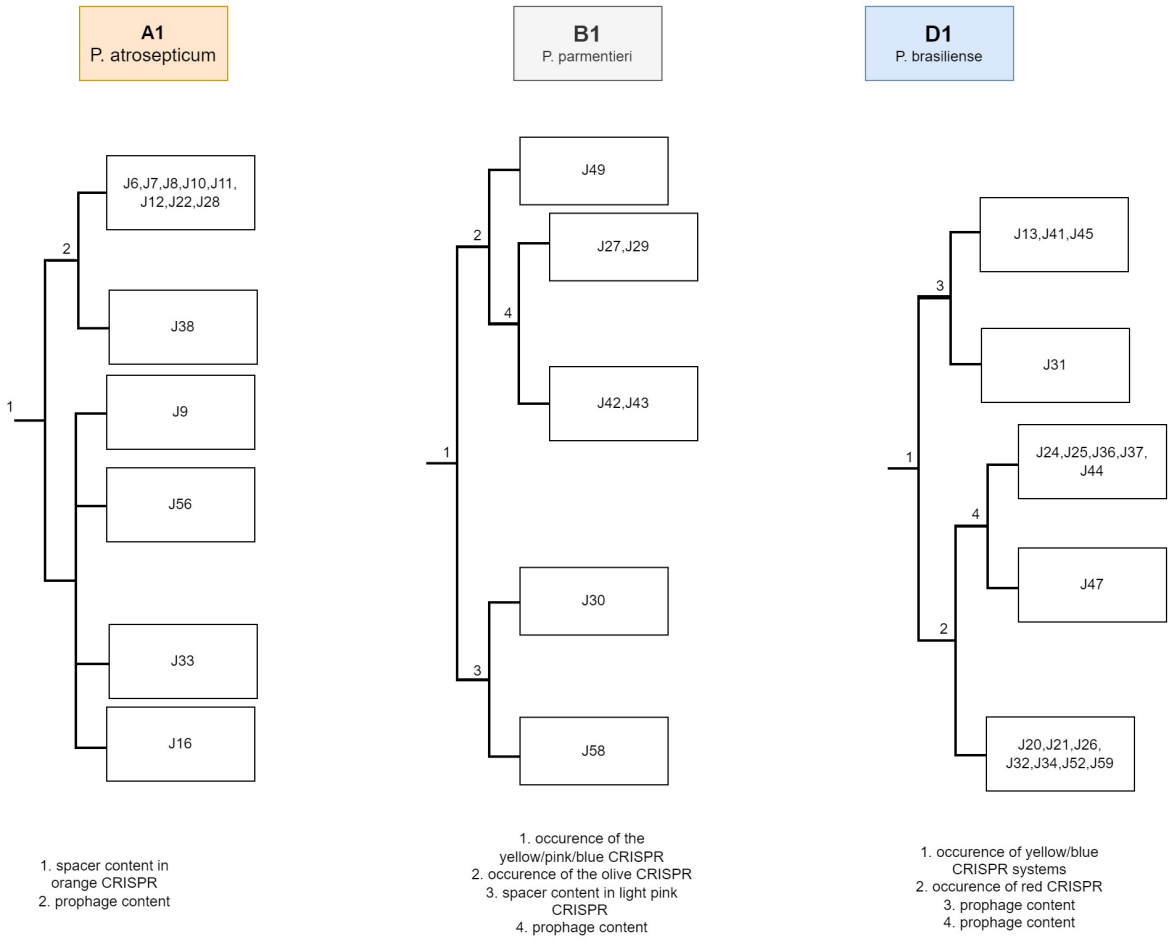

Figure 7. phylogenetic tree updated with prophage content data

Step 6

Overview In the last step I tried to compare the full genomes.

Methods I already knew that my bacteria form three main groups, each representing a different species, therefore I have decided to only compare those belonging to the same species. As the genomes are very simillar with just a tiny differences, it was challenging to find a right tool for comparison. I have decided to use GSAlign[17], which is specifically designed for intra-species genome comparison. It aligns two sequences and returns the descriptions of insertions, deletions and SNVs in a .vcf file. It also returns the whole sequence alignment in .maf format. I wrote a bash script *run_gsalign.sh* that was used to run GSAlign between all possible sequence pairs inside of the group. Output (.vcf file) was later passed as an argument into a python script called *get_GSAlign_out.py*. It was summing up all the mutation for a certain pair and later writing it together with the bacteria IDs' in a separate line in *meg_file.txt* file. Once the looping was over, *meg_file.txt* together with a list of bacteira IDs' forming the group were passed to another python script called *make_distance_matrix.py*. Its role was to create a lower triangular distance matrix and save it into a .meg file that could be later used to construct the tree.

For tree construction I used the MEGA11 program. I used construct Neighbor-Joining tree option and a default settings. I created the trees for all three big groups.

Results Each big group tree made with MEGA is paired with the one build on prophage/CRISPR data, so that it is easier to compare the results.

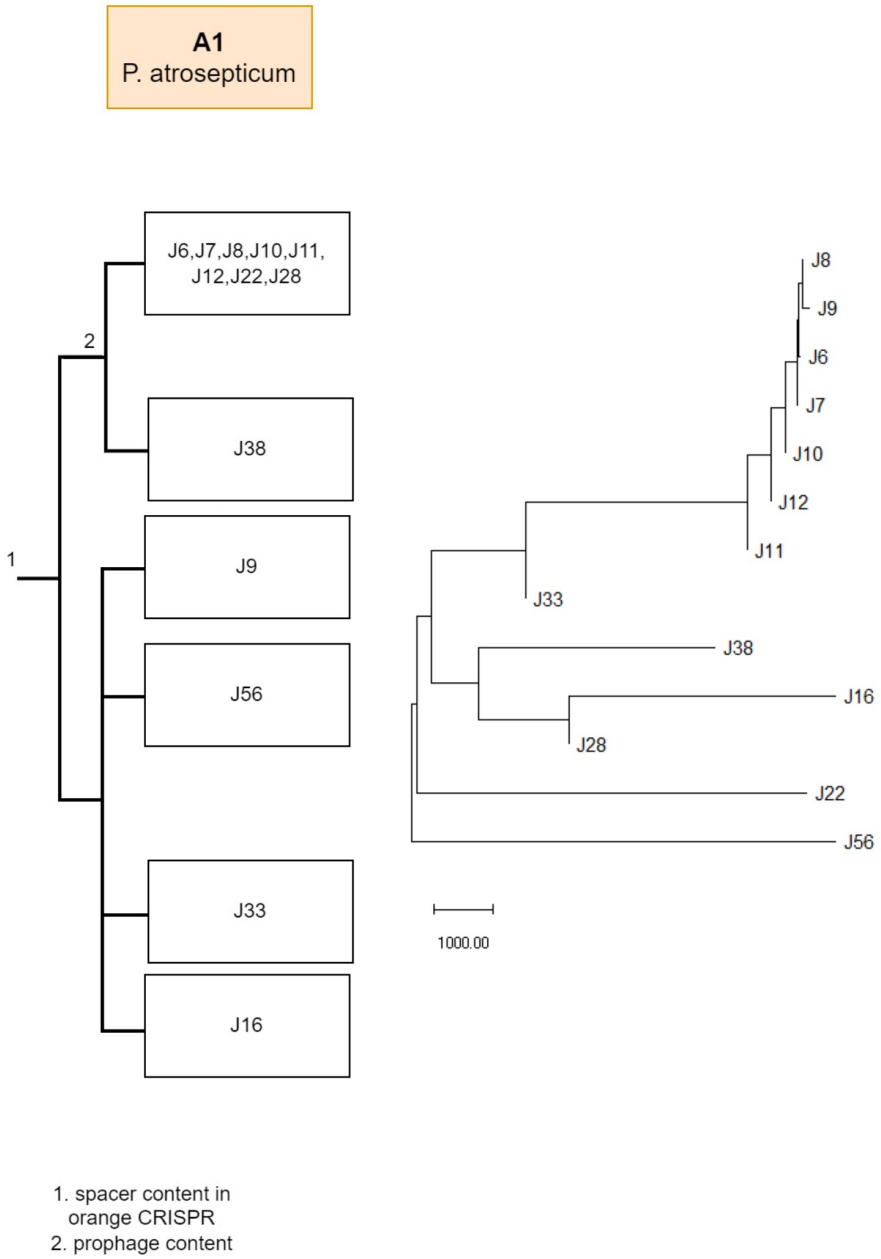

Figure 8. Trees for the orange group - P. atrosepticum

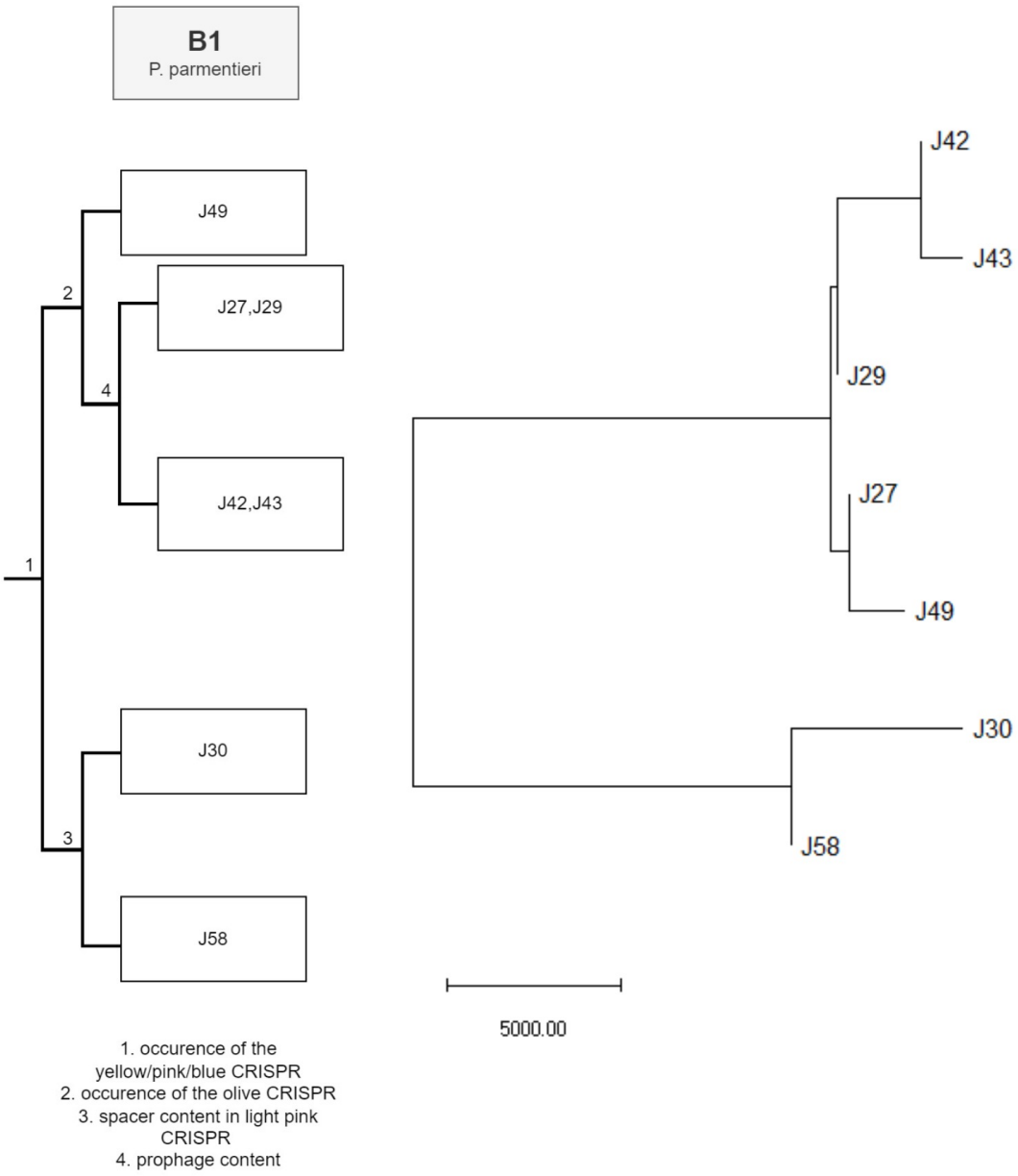

Figure 9. Trees for the grey group - P. parmentieri

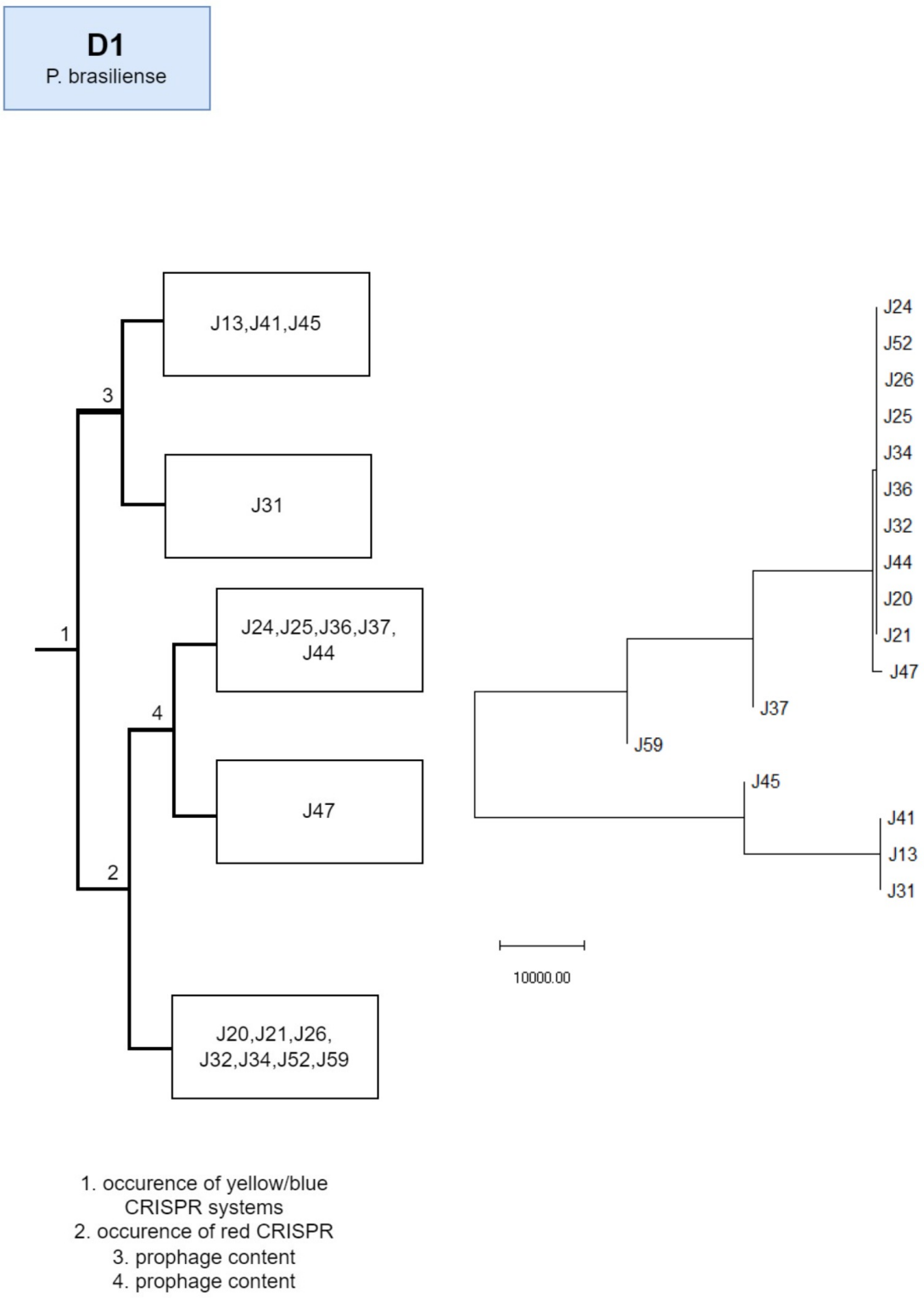

Figure 10. Trees for the blue group - P. brasiliense

Discussion

The main purpose of the first question was to find out whether assembled phages that are thought to be used in a cocktail aiming to fight soft rot and blackleg disease are actually useful. After researching the question with several methods, I could say with a high dose of certainty that none of the bacteria is resistant to none of the phages I worked with.
As it did not fully answer the first question, I moved to the second part when I tried to find the actual targets for all the extracted spacers.
Phages are a very common entities, but they are not well studied yet. With a development of a NGS methods it becomes much easier to sequence their genomes and the amount of data stored in databases is constantly increasing. In 2020 NCBI database contained 8239 complete phage genomes[18], but only a small percentage has a Pectobacterium hosts (or does not have a host identified at all). That being said, even before "blasting" the spacers against the NCBI, it seemed highly improbable that some groundbreaking discoveries will be made. As suspected, there were just a very little hits that allowed me to assign a spacer to its target, but it turned out that BLAST results may be the source of many valuable information about the age and "place-specificity" of identified CRISPR systems. Most of the spacers were identified as being a part of one of the NCBI Pectobacterium genomes of the species a specific bacteria was assigned to (e.g. for bacteria in orange group 99% of the spacers were found in one of the P. atrosepticum genomes stored in the database). The fact that there were some unidentified spacers means, that there are no bacteria containing such a fragment in their genome stored in a database yet. Determination of the cause of that was not actually a part of the research question, but as it turned out to shed light on some evolutionary relations inside a groups, I have decided to include this thread in the discussion. There could be several reasons for getting no hits when "blasting" against database and I will try to explain them on the example of individual groups.
P. atrosepticum was the first bacteria identified as a soft rot pathogen[19], therefore it is relatively well studied. For this species, out of all the spacers only the last one in shorter CRISPR of J39/J40 was not identified. CRISPR systems, as all the other parts of bacteria genome are subject to mutation and the longer some fragment is present in a genome the more likely it is to be mutated. As the spacer of interest is the furthest (starting from the leader end), probably the oldest one and all newer ones have been found, it seems very unlikely that it was not identified because of the unique invader exposure. such contact would have to have happened a long time ago, and the spacer for some reason has not been passed on to subsequent generations in contrast to all the others. It is much more likely that the spacer mutated relatively recently and therefore is not shared with none of the genomes stored in NCBI.
There are 15 complete, assembled P. parmentieri genomes stored in NBCI, but more than a half were submited by the same group and collected in the same country what surely introduces some bias. Out of 9 CRISPR systems identified in this group, one stands out as extremely "unrecognized" (7/17 spacers) while the rest include no more than one "unmatched" spacer per system (mostly the most recent e.g. closest to leader end ones indicating fresh exposure to the unknown invader). The "unrecognized" system was only found in J49, what means only 14% of the group contains it. These things suggest it is a relatively new, specific for the small group system. The fact, that 6/17 (35%) spacers were identified targeting prophages identified in our bacteria (11, 18 and 15 % for other J49 CRISPRs), lends credence to our supposition that this system is specific for bacteria occuring in the area from where the potato tubers were collected.
P. brasiliense is thought to be one of the most virulent among Pectobacteriae[20]. It is causing soft rot not only in potato, but also in many other vegetables and therefore its genomes are less homogeneous that these of the other Pectiobacteriae subgroups[20]. P. brasiliense was first detected in potato in Brasil and was thought to be the main soft rot causing agent in warmer parts of globe. More recent studies show it's introduction to the colder regions of Europe which is linked with a climate changes. Having said that, it does seem logical that out of the three big subgroups, blue one is relatively the worst studied. There is a clear division between bacteria belonging to the yellow/blue/green CRISPR group (J13,J31,J45,J41) and the others. While spacers for the first were identified in NCBI in 100%, for CRISPR systems of the second group it was only 23%. When combined with the information about percentage of spacers identified as targeting prophages in our bacteria: 43% first and 23% second group, this may suggest that the first group was introduced to the Danish fields earlier and was much better studied while the second is more of an "latest addition".
The idea to identify prophages/plasmids inside our bacteria genomes was an attempt to find the answers that "blasting" against NCBI failed to give us. As all the bacteria share the same host they are very prone to be infected with the same invaders.It therefore seemed very reasonable to suspect if the bacteria did not have a certain prophage, it is because they acquired resistance not because they were not exposed. After analyzing the results, it seems that the idea is reflected in reality as we got much more hits when blasting "against" identified prophages than when using NCBI database. Once again, it would be naive to expect that such an approach will answer all the questions, but it certainly brought us closer to solving the first of two main project questions.
Chosen approach for sure has some shortcomings, both when it comes to methods used for creating the "maps" as well as group chosen for analysis. Starting off with a first, there is no way for avoiding bias introduced when identifying plasmids and prophages. VIBRANT is a tool that, using neural-network, looks for viral proteins in bacterial genomes. It performs HMM

search against three different databases to distinguish between host and prophage proteins and eliminate most of the non-viral scaffolds. Rest of the potential viral proteins are trimmed and passed into a neural network which separates viral scaffold from the remaining non-viral ones. As most of the machine learning approaches this one tends to be a little unpredictable as well. During my project I realized that it was identifying a prophage in one bacteria while missing exactly the same one in another. What was also strange is the fact, that although it works on a protein level, therefore it surely knows the direction in which the phage genes are transcribed it does not save this information in the output while returning exactly the same phage in 5' -> 3' direction in one bacteria and 3' -> 5' in another. That being said, it is highly probable that the results I obtained using this tool were far from the reality. The same thing applies to the script I have written for identifying the plasmids. As the homologous approach was not possible in this case, everything was based on an assumption that the plasmid coverage is either significantly smaller or significantly bigger than the chromosomal one. If any of the plasmids would have number of copies close to the chromosomal one, it would end up being classified as a part of the chromosome. Another achilles heel is the fact that I've arbitrary chosen the "boundary" numbers. There were no experimental basis justifying the choice and it could be easily possible that, when slightly adjusting the thresholds, I would end up with a significantly different results. The thing is, that to my knowledge there is no way of defining the numbers without identifying the plasmids first, and then the numbers would be useless. It leaves us without any reasonable approaches for improving the script and once again we must work with the imperfect tool once again.

It is also worth noting that the spacers must not actually target a mobile genetic element. Research shows that sometimes spacers can target their own genomes without inducing autoimmune response[21] (I would not be able to notice such an events with my experimental stup as I was eliminating self-hits and this events would be counted as self hits as well), they could also not target anything when they were acquried in a primed adaptation process[22].

The last, probably most important reason for identifying so little spacer targets was the fact that I only created a map for 59 of the collected bacteria, which is just a small section of genomes that could potentially host the prophages we are interested in. That being said, to increase the numbers on identified targets, one should not only improve the tools, but most importantly increase the number of "maps" that the spacers could be compared to. I believe that the results of NCBI search can also be used as an indicator of undiscovered prophages. Although most of the spacers tend to match only a genome they were extracted from, there is also a big group of more "popular" ones shared between different species. It is unlikely a consequence of CRISPR conservation as it only applies to several spacers not a whole system. This made me thing that the popular spacers actually target "popular" prophages shared between several species.

As already mentioned in the introduction, answering the second question was not straightforward. We were expecting the vast majority of isolated bacteria to belong to the Pectobacteriae family and therefore the differences between the individual genomes to be very small. It was clear that simple genome comparison will not give us enough information for several reasons: first of all, it is already really computationally challenging to compare two whole bacterial genomes, and having to expand it to 59 genomes that have to be compared each with each would be impossible with our time and hardware resources. Even if, we would somehow manage to solve the first problem, there are also some other significant issues such as assessing whether a certain mutation actually occurred or is just an sequencing error. Except of deciding on the source of origin, we would need to redefine a scoring system. For example, prophage build up into a genome would result in two bacteria being categorized as highly dissimilar even if the rest of their genomes alligned in 100%. This is because prophage genomes are usually few kbp long and therefore have a huge impact on the assembly score. There are also some minor challenges such us "how shall we deal with mutations on the border of the two contigs?" - they would be treated as two separate events although it certainly does not reflect the reality. Having said that, it was clear that we must search for some bypass approaches to answer the question.

Investigating and comparing CRISPR systems was one of the possible approaches. As the spacers and also the whole systems are acquired after the exposure to the invader and later passed to the next generations, they could serve as a kind of a stamp allowing us to cath the significant differences between a very closely related genomes. Analysis of the CRISPR systems allowed me to distinguish three main groups and several small ones consisting of up to three bacteria. After learning more about the mechanisms underlying spacers "reshuffles" I was even able to make some splits inside the groups.

Orange group turned to be the most homogeneous with just a slight reshuffles inside the orange CRISPR system. There were four such an events observed and they were all deletions of the middle spacers combined with duplication in one of the cases. They were probably caused by the strong selective pressure that have appeared in the environment, but looking only at the CRISPR data it is impossible to say if the events were somehow correlated or locate them one a timeline. That is why, when building a CRISPR-based tree, they were all treated equally. I divided blue group into two subgroups (D1.1 and D1.2 in phylogenetic data). I was able to make one more split inside of the D1.1 subgroup. As it is much more likely that the CRISPR system is acquired than lost, I assumed that bacteria containing the red one evolved from these having only three others. Grey group could be divided into two main branches, which gave us information we could not get from the phylogenetic data. When analyzing light pink CRISPR in J30 and J58 I could tell that J58 actually evolved from J30 and not vice versa. That is the kind of information that may be very useful for tracking the bacteria spread and is itself a strong argument in favor of using this

approach.
Analysis of the prophage content gave me some additional information that allowed me to update the trees build with the CRISPR data. I made an assumption that, in contrast to the spacers, the same prophages can be build up into different genomes in an independent events. As all of the other steps, this one was imperfect as well. I have decided to treat the sequences with more than 97% identity as equal and later, when trying to improve VIBRANT results I was looking for sequences with more than 99% similarity and 95% of the prophage length. When doing this I realized that there are also some relations between prophage groups that I did not studied. Many of the prophages tend to share up to few thousand bp sequences. As the alignment, although usually 100% identical, did not cover more than 95% prophage length they were classified as two different invaders. To get a more realistic picture, I would need to dive deeper into tracking the relations between different phages. Anyway the data I obtained allowed me to update the trees and divide the species into even smaller clusters.
As a last step I wanted to compare the whole genomes and build a competing tree. I have already discussed problems of such an approach and they of course did not disappeared, but I used a tool that helped me to solve most of them. As GSAlign was made specifically for comparing intra-species genomes, it uses the fact they are so similar to speed up the comparison. By default it only takes small indels (<25 bp) into account and the bigger ones are scored as 0 what solves the "prophage problem". Its main disadvantage is the fact that it is still unable to distinguish between sequencing errors and actual mutations. At this level of kinship, when there are just a very few mutations across the genome observed it introduces a lot of bias. It also, in contrast to CRISPR analyisis, does not give us any information about the direction of changes and their evolutionary importance. For this reason, the conclusions we can draw from the analysis are something like "X is that different from Y" not "X is that different from Y, so I can be sure that it evolved from it and not vice versa". Using such an approach, although we can catch much more differences, we are loosing a very important information about the direction of the observed changes. I believe that best we could do is to actually use both approaches and compare the results trying to draw conclusions analyzing both trees separately.
There are significant differences between the trees for all of the groups. The main trends are preserved, although there are significant differences inside the subgroups. None of the trees gives us a straightforward evolutionary informations.
After analyzing the P. atrosepticum trees, it is clear that J6,J7,J8,J9,J10,J11 and J12 are almost identical. It must be confirmed by the metadada, but I would risk the statement that they were collected in a neighbouring fields or at least fields belonging to the same farmers. What is interesting is the fact that J22 and J28 although they not only share the exactly the same CRISPR systems, but even the same prophages were classified as very distant using distance matrix. As I mentioned above, GSAlign ignores the big inserts and is therefore unable to catch differences caused by the prophage content (i.e. we would get the same results when using genomes with and without prophages), therefore one should only focus on CRISPR systems when trying to explain this event. As there were more than 5000 differences noticed between J22 and the core of the group, it is clearly not only due the sequencing errors. With only two contigs, assembly quality also seem to be pretty hig. This leads to the conclusion that the CRISPR systems are actually much more conserved than we expected them to be. Of, course, to confirm that, I would need to look at the distribution of mutations in the genome to make sure that they were not concentrated in one place (this would suggest mistakes during the assembly/ analysis process not necessarily the real mutations). Generally, the trees are not mutually exclusive, although of course, MEGA one distinguish branches that I treated equally. I would not risk translating this into an evolutionary information though.
For grey group (P. parmentieri) the trees are almost completely overlapping. The information from CRISPR./prophages fully alligns with the one collected on the genetic level.
Blue group seems to be the most homogeneous. Using distance matrix ten genomes were assigned as almost equal and eleventh (J47) is also really similar. This in general aligns with the CRISPR/prophage data, although there are some differences. When using CRISPR/prophage approach I managed to divide D1.1 into two main subgroups based on presence of the red CRISPR system. This difference, although crucial from the evolutionary point of view, remained unnoticed in the MEGA trees and almost all bacteria from D1.1 (except of J37 and J59) formed one big group. Two remaining bacteria, although grouped together with the rest when using CRISPR/prophage data, turned out to differ a lot. This once again, leads to the hypothesis that the CRISPR systems are actually more conserved that we expected them to be.
Summarizing the second part: two tested approaches lead to the same main groups and subgroups although they tend to group some of the bacteria inside the subgroups differently. Although distance matrix based trees allow for greater distinction, I would personally choose the CRISPR/prophage ones to track the spread of the bacteria as they are less likely leading to incorrect conclusion (i.e. if there was a split it is very likely to be of an evolutionary importance). It is of course worth keeping the MEGA trees and referring to them for example in matters of dispute. To improve analysis, except of changing the tools, one could search for another genome functional elements that could serve as an evolutionary stamps. Developing a knowledge-based valuation system for different approaches would definitely help with organizing the data and make a results more comparable. I believe that using CRISPR to track evolutionary relationships has potential, but some sorting is definitely needed if it is to become "go to" approach for closely related species comparison.

References

1. Czajkowski, R., Pérombelon, M. C. M., van Veen, J. A. & van der Wolf, J. M. Control of blackleg and tuber soft rot of potato caused by pectobacterium and dickeya species: a review. *Plant Pathol.* **60**, 999–1013, DOI: https://doi.org/10.1111/j.1365-3059.2011.02470.x (2011).
2. Farmers from the nepg zone will globally produce 7 to 11 % less potatoes due to climate change. Online (2022). Accessed on September 12, 2022 from [http://nepg.info/wp-content/uploads/2022/09/220912-NEPG-press-release-GB-final.pdf].
3. Perombelon, M. C. M. & Hyman, L. J. Serological methods to quantify potato seed contamination by erwinia carotovora subsp. atroseptica. *EPPO Bull* **25**, 361–87 (1995).
4. Zaczek, M., Weber-Dabrowska, B. & Gorski, A. Phages in the global fruit and vegetables industry. *Appl Microbiol* **118**, 537–56 (2015).
5. Jensen, M. F., Karlsson, M. & Sarrocco, S. e. a. Biological control using microorganisms as an alternative to disease resistance. *Plant Pathog. Resist. Biotechnol.* 341–63 (2016).
6. Jansen, R. & et. al. Identification of a novel family of sequence repeats among prokaryotes. *OMICS 6* **1**, 23–33 (2002).
7. Deveau, H. *et al.* Phage response to crispr-encoded resistance in streptococcus thermophilus. *J. Bacteriol.* **190**, 1390–1400, DOI: 10.1128/jb.01412-07 (2008).
8. Marraffini, L. A. & Sontheimer, E. J. Crispr interference: Rna-directed adaptive immunity in bacteria and archaea. *Nat. Rev. Genet.* **11**, 181–190, DOI: 10.1038/nrg2749 (2010).
9. McGinn, J. & Marraffini, L. A. Molecular mechanisms of crispr-cas spacer acquisition. *Nat. Rev. Microbiol.* **17**, 7–12, DOI: 10.1038/s41579-018-0071-7 (2019).
10. Sun, C. L., Thomas, B. C., Barrangou, R. & Banfield, J. F. Metagenomic reconstructions of bacterial crispr loci constrain population histories. *ISME* **10**, 858–870, DOI: 10.1038/ismej.2015.162 (2016).
11. Semenova, E. *et al.* Highly efficient primed spacer acquisition from targets destroyed by the escherichia coli type i-e crispr-cas interfering complex. *Proc. Natl. Acad. Sci. U.S.A.* **113**, 7626–7631, DOI: 10.1073/pnas.1602639113 (2016).
12. Chen, C., Fuqua, C. & et. al. Editorial: Plasmid transfer-mechanisms, ecology, evolution and applications. *Front. Microbiol.* **13**, DOI: https://doi.org/10.3389/fmicb.2022.993628 (2022).
13. Russel, J., Pinilla-Redondo, R., Mayo-Munoz, D., Shah, S. & Sorensen, S. Crisprcastyper: Automated identification, annotation, and classification of crispr-cas loci. *The CRISPR J.* **6**, 462–469, DOI: 10.1089/crispr.2020.0059 (2020).
14. Biswas, A., Staals, R., Morales, S., Fineran, P. & Brown, C. Crisprdetect: A flexible algorithm to define crispr arrays. *BMC Bioinforma.* **17**, DOI: https://doi.org/10.1186/s12864-016-2627-0 (2016).
15. Zhang, R. *et al.* Spacepharer: sensitive identification of phages from crispr spacers in prokaryotic hosts. *Bioinformatics* **37**, 3364–3366, DOI: https://doi.org/10.1093/bioinformatics/btab222 (2021).
16. Kieft, K., Zhou, Z. & Anantharaman, K. Vibrant: automated recovery, annotation and curation of microbial viruses, and evaluation of viral community function from genomic sequences. *Microbiome* **8**, DOI: https://doi.org/10.1186/s40168-020-00867-0 (2020).
17. Lin, H. & Hsu, W. Gsalign: an efficient sequence alignment tool for intra-species genomes. *BMC Genomics* **21**, DOI: https://doi.org/10.1186/s12864-020-6569-1 (2020).
18. Zrelovs, N., Dislers, A. & Kazaks, A. Overview of the currently available phage diversity. *Front. Microbiol.* **11**, DOI: 10.3389/fmicb.2020.579452 (2020).
19. Perombelon, M. C. M. & Kelman, A. Ecology of the soft rot erwinias. *Annu. Rev. Phytopathol.* **18**, 361–367, DOI: 10.1146/annurev.py.18.090180.002045 (1980).
20. Said, O. *et al.* Pectobacterium brasiliense: Genomics, host range and disease management. *Microorganisms* **106**, DOI: 10.3390/microorganisms9010106 (2021).
21. Wimmer, F. & Beisel, C. Crispr-cas systems and the paradox of self-targeting spacers. *Front. Microbiol* **10**, DOI: https://doi.org/10.3389/fmicb.2019.03078 (2020).
22. Musharova, O., Medvedeva, S., Savitskaya, E. & et. al. Prespacers formed during primed adaptation associate with the cas1–cas2 adaptation complex and the cas3 interference nuclease–helicase. *PNAS* **118**, DOI: https://doi.org/10.1073/pnas.2021291118 (2021).

Additional information

all the code I have written is stored in this repository: https://github.com/ideczka32/Individual-project-in-Bioinformatics