Evolutionary Dynamics, Diversity and Interactions of CRISPR Arrays

Dissertation
der Mathematisch-Naturwissenschaftlichen Fakultät
der Eberhard Karls Universität Tübingen
zur Erlangung des Grades eines
Doktors der Naturwissenschaften
(Dr. rer. nat.)

vorgelegt von
Axel Fehrenbach
aus Herrenberg

Tübingen
2026

Gedruckt mit Genehmigung der Mathematisch-Naturwissenschaftlichen Fakultät der Eberhard Karls Universität Tübingen.

Tag der mündlichen Qualifikation	08.10.2026
Dekan:	Prof. Dr. Thilo Stehle
1. Berichterstatter:	Prof. Dr. Franz Baumdicker
2. Berichterstatter:	Prof. Dr. Daniel Huson

Der Mensch in seinem Wissensdrang
Sinniert und forscht sein Leben lang,
Um dann verzichtend einzusehn:
Im Grunde kann er nichts verstehn.

— *Resignation*, Karl Ritter von Frisch (1967)

Abstract

In the evolutionary arms race between phages and prokaryotes, CRISPR-Cas systems provide adaptive immunity against invading genetic elements. These systems consist of CRISPR-associated (Cas) genes and CRISPR arrays and exhibit remarkable diversity across prokaryotes. During infection, short fragments of foreign DNA, termed spacers, are acquired by Cas proteins and integrated between repeats in the CRISPR array. These spacers are then transcribed and processed into guide RNAs that direct the Cas machinery to recognize and target matching invaders, thereby conferring sequence-specific immunity.

Because new spacers are typically added at the leader end of an array, spacer order forms a chronological record of past infections, with older spacers located progressively farther from the leader end. At the same time, spacer loss through deletions introduces additional evolutionary complexity. As a result, CRISPR arrays provide a unique record of host-phage interactions and microbial adaptation. However, key aspects of CRISPR array evolution remain poorly understood, including deletion mechanisms and rates, array orientation and leader localization, and the coordinated interaction and evolution of multiple arrays and systems within a single genome.

To address these challenges, we developed *SpacerPlacer*, a computational framework for reconstructing the evolutionary history of related CRISPR arrays along a phylogeny while preserving spacer order. We further developed stochastic models of array evolution together with statistical methods to infer evolutionary rates and test hypotheses on spacer deletion mechanisms. Building on this framework, we introduced *CRISPR-evOr*, which predicts the acquisition end of CRISPR arrays and thereby indirectly infers information about their orientation and leader location.

Applying these methods to large-scale CRISPR array datasets revealed conserved and type-specific evolutionary patterns across CRISPR-Cas systems. Spacer deletions are frequent, often involve multiple adjacent spacers, are less common at array boundaries, and are shaped by selective pressures, patterns that are broadly conserved across system types. At the same time, CRISPR-Cas systems exhibit distinct strategies of genomic organization. They show type-specific characteristic patterns in genomic positioning, repeat similarity, and orientation. Some systems occur as single instances within a genome, while others distribute spacers across multiple arrays or co-occur with additional CRISPR systems. These observations suggest that CRISPR systems may act independently or antagonistically in some contexts, while in others they function cooperatively by supporting each other in spacer acquisition, expression, and interference.

Taken together, the results presented in this thesis provide a qualitative and quantitative framework for studying CRISPR array evolution across multiple scales, enabling the analysis of evolutionary patterns and selective forces and yielding insights into microbial ecology and CRISPR-Cas functionality in prokaryote host-phage interactions.

Zusammenfassung

Im evolutionären Wettrüsten zwischen Phagen und Prokaryoten stellen CRISPR-Cas-Systeme eine adaptive Immunabwehr gegen eindringende genetische Elemente dar. Diese Systeme bestehen aus CRISPR-assoziierten (Cas-)Genen und CRISPR-Arrays und weisen eine bemerkenswerte Vielfalt innerhalb der Prokaryoten auf. Während einer Infektion werden kurze Fragmente fremder DNA, sogenannte *spacer*, von Cas-Proteinen aufgenommen und zwischen *repeats* in den CRISPR-Array integriert. Diese *spacer* werden anschließend transkribiert und zu *guide RNAs* verarbeitet, die Cas-Effektorkomplexe zu passenden Zielsequenzen dirigieren und dadurch einen sequenzspezifischen Schutz der Zelle ermöglichen.

Da neue *spacer* typischerweise am *leader*-Ende eines Arrays eingefügt werden, spiegelt ihre Reihenfolge die zeitliche Abfolge vergangener Infektionen wider. Gleichzeitig können *spacer* durch Verlustereignisse verloren gehen, was zu komplexen evolutionären Dynamiken führt. Dadurch stellen CRISPR-Arrays ein einzigartiges Abbild von Wirt-Virus-Interaktionen und mikrobieller Adaptation dar. Trotz ihrer biologischen Bedeutung sind wesentliche Aspekte der Evolution von CRISPR-Arrays nach wie vor nicht vollständig verstanden, darunter Mechanismen und Raten von *spacer*-Verlusten, die Orientierung von Arrays und die Lokalisierung des *leaders*, sowie die koordinierte Interaktion und Evolution mehrerer CRISPR-Arrays und -Systeme innerhalb eines einzelnen Genoms.

Um diese Herausforderungen anzugehen entwickelten wir *SpacerPlacer*, ein computergestütztes Verfahren zur Rekonstruktion der Evolutionsgeschichte verwandter CRISPR-Arrays entlang eines phylogenetischen Baums unter expliziter Berücksichtigung der *spacer*-Reihenfolge. Darüber hinaus formulierten wir stochastische Modelle der Array-Evolution sowie statistische Methoden zur Schätzung evolutionärer Raten und zur Überprüfung von Hypothesen über die Mechanismen von *spacer*-Verlusten. Aufbauend auf *SpacerPlacer* entwickelten wir zudem *CRISPR-evOr*, eine Methode zur Vorhersage des Akquisitionsendes von CRISPR-Arrays, die damit indirekt zur Bestimmung ihrer Orientierung sowie der Position des jeweiligen *leaders* beiträgt.

Die Anwendung dieser Methoden auf große CRISPR-Array Datensätze offenbarte sowohl systemübergreifend erhaltene als auch typspezifische Evolutionsmuster innerhalb der CRISPR-Cas-Systeme. *spacer*-Verluste treten häufig auf, betreffen oft mehrere benachbarte *spacer*, kommen an den Rändern der Arrays seltener vor und werden durch selektive Kräfte geprägt, sind aber über verschiedene Typen hinweg weitgehend ähnlich ausgeprägt.

Gleichzeitig unterscheiden sich CRISPR-Cas-Systeme in ihrer genomischen Organisation. Sie weisen typspezifische Muster in genomischer Positionierung, *repeat*-Ähnlichkeit und Orientierung auf. Einige Systeme treten als einzelne Instanzen innerhalb eines Genoms auf, während andere ihre *spacer* auf mehrere Arrays verteilen oder gemeinsam mit anderen CRISPR-Cas-Systemen vorkommen. Diese Beobachtungen deuten darauf hin, dass CRISPR-Cas-Systeme in bestimmten Bedingungen unabhängig oder antagonistisch agieren können, während sie in anderen kooperativ funktionieren und sich gegenseitig bei Spacer-

Akquisition, Expression und Interferenz unterstützen.

Zusammenfassend liefern die Ergebnisse dieser Arbeit einen qualitativen und quantitativen Rahmen zur Untersuchung der Evolution von CRISPR-Arrays über verschiedene Skalen hinweg. Dieser ermöglicht die Analyse evolutionärer Muster und selektiver Kräfte und liefert zugleich Einblicke in die mikrobielle Ökologie sowie die Funktionalität von CRISPR-Cas-Systemen in Interaktionen zwischen prokaryotischen Wirten und ihren Phagen.

Acknowledgements

First, I would like to thank my supervisor Franz Baumdicker. Throughout my PhD, he provided excellent guidance, generously sharing his expertise and enthusiasm for population genetics and my research, even at times when my own motivation was low. He always made time for discussions and offered valuable advice whenever needed.

I am also deeply grateful to my colleagues and collaborators throughout these projects, in particular my collaborators in Freiburg, especially Alexander Mitrofanov and Rolf Backofen, for insightful discussions, support during the projects, and many enjoyable meetings both in person and online.

I would further like to thank the members of Franz' group, Klara Burger, Johannes Kippnich, Hannah Götsch, and more recently Yuxin Ning and Lukas Tatzel, for creating an excellent working atmosphere despite the challenges of spatial separation, as well as for many enjoyable (online) coffee breaks and get-togethers.

My thanks also go to my friends, who provided invaluable moral support throughout the research and writing process and patiently listened to my frustrations. In particular, I would like to thank the people at the Sand, especially Martin Meßmer and Benjamin Kiefer, for their support and for providing welcome distraction during both of my PhD attempts. I am equally grateful to the Integrative Transkriptomik group and its former and associated members for entertaining lunch and coffee breaks, especially during times when our group offices were less full, for the shared experiences, mutual encouragement, and many fun evenings.

Last but not least, I would like to thank my family, who have been deeply supportive during my PhD (and throughout all my life), patiently listened to my (sometimes unnecessary) complaints, and provided a place of calm and recovery during the most stressful times.

Contents

Abstract i

Zusammenfassung (German Abstract) iii

Acknowledgements v

List of figures and tables ix

1 Introduction 1
- 1.1 Thesis outline and structure . . . 9

2 CRISPR-Cas systems: form, function and ecology 13
- 2.1 Overview and terminology . . . 14
- 2.2 Mechanisms of CRISPR-Cas immunity . . . 18
- 2.3 Ecology of CRISPR-Cas systems . . . 25

3 CRISPR array evolution 31
- 3.1 Spacer acquisitions . . . 32
- 3.2 Spacer (block) deletions . . . 34

4 Mathematical modeling of CRISPR array evolution 37
- 4.1 Representation of spacer arrays . . . 38
- 4.2 Models for spacer insertion and deletion . . . 38
- 4.3 Usage of evolutionary models in and around *SpacerPlacer* . . . 43
- 4.4 Derivation of the likelihood functions and the ratio test . . . 44
- 4.5 Comparison to other models of spacer array evolution . . . 51

5 *SpacerPlacer*: ancestral reconstruction of CRISPR arrays reveals the evolutionary dynamics of spacer deletions 55
- 5.1 Introduction . . . 56
- 5.2 Material & Methods . . . 57

5.3 Results . . . 60
5.4 Discussion . . . 76

6 An evolutionary approach to predict the orientation of CRISPR arrays 81
6.1 Introduction . . . 82
6.2 Methods . . . 88
6.3 Results . . . 91
6.4 Discussion . . . 95

7 The complexity of multiple CRISPR arrays in strains with (co-occurring) CRISPR systems 99
7.1 Introduction . . . 100
7.2 Methods . . . 101
7.3 Results & Discussion . . . 103
7.4 Conclusion . . . 116

8 Applications, perspectives, and conclusion 119
8.1 Data availability and applications in industry, biotechnology and medicine . 119
8.2 Refining *SpacerPlacer* and its statistical framework . . . 121
8.3 CRISPR system interactions and the need for a genome-wide perspective . 122
8.4 Applying CRISPR array models to integron evolution . . . 124
8.5 Conclusion . . . 126

Bibliography 128

List of Figures

1.1 Overview of CRISPR-Cas immunity 5

2.1 Targets and modular organization of CRISPR systems by type 16

3.1 Illustration of CRISPR-phage co-evolution in a bacterial community over time 32
3.2 Illustration of deletion (and duplication) events caused by misalignment during replication 35

4.1 Preprocessing and representation of CRISPR spacer arrays 39
4.2 Separation of acquisitions and deletions by retained spacers 40
4.3 Key differences between the simplified and precise likelihood computations for block deletions 47

5.1 Different deletion and insertion models 61
5.2 Workflow of *SpacerPlacer*: ancestral reconstruction 63
5.3 Partial Spacer Insertion Order based refinement of the guide reconstruction 65
5.4 Likelihood ratio test workflow in *SpacerPlacer* 68
5.5 Parameter estimation and LRT-performance on simulated data 69
5.6 Proportions of deletions with respect to the position along the array 70
5.7 Parameter estimates on CRISPRCasdb data 72
5.8 Deletion frequencies with respect to the proximity to the array boundary . 75

6.1 *CRISPR-evOr* workflow 85
6.2 Example of a group of arrays with high confidence orientation prediction . . 87
6.3 Comparison between acquisition- and PAM-orientation 92
6.4 Comparison with repeat based orientation methods 94

7.1 Single and co-occurring Cas types 103
7.2 Distance to Cas and array orientation 106
7.3 Distribution of the estimated insertion rate weights of arrays 109

7.4 Visualization of repeat similarity by projection into 2-d space with multidimensional scaling 112
7.5 Repeat similarity MDS joint scatterplot and marginal distributions 114

List of Tables

4.1 Comparison of spacer acquisition and deletion models used across publications 52

5.1 Mean and median parameter estimates in the independent and block deletion model 73
5.2 Number and proportions of duplication events 74

6.1 Orientation concepts overview 83

7.1 Summary of findings and possible functional implications proposed/supported in the literature 117

1 Introduction

To survive and reproduce, all organisms, from the largest multicellular lifeforms to the smallest microbes, must compete for limited resources and adapt to changing environments. Over evolutionary timescales, these processes shape and are shaped by genetic variation encoded in their DNA.

As organisms adapt to their environments, they also adapt to one another. This has led to the evolution of diverse ecological interactions, including the exploitation of one organism by another. While such interactions can be beneficial or neutral to the host, they often impose fitness costs and may ultimately lead to disease or death. Consequently, hosts have evolved a remarkable diversity of defense mechanisms to protect themselves against invasion, exploitation, and infection by these opportunistic adversaries.

These defense systems are referred to as *immune systems* and are found across virtually all forms of life. They are broadly classified into *innate immune systems*, which provide rapid, pre-configured responses to diverse threats, and *adaptive immune systems*, which can learn from prior encounters and mount highly specific responses. Remarkably, even the smallest organisms, such as bacteria, possess highly sophisticated immune mechanisms and encode a striking diversity of both innate and adaptive defense systems.

The discovery, characterization, and application of two such bacterial defense systems are widely regarded as among the most significant scientific advances of the late twentieth and early twenty-first centuries and have been recognized with two Nobel Prizes (Nobel Prize Outreach 2026a; Nobel Prize Outreach 2026b).

One such defense mechanism is the *CRISPR-Cas system* (or simply *CRISPR system*), the central focus of this work. Found in bacteria and archaea, CRISPR-Cas systems provide adaptive immunity against viral infections and other invading genetic elements.

These ***C****lustered* ***R****egularly* ***I****nterspaced* ***S****hort* ***P****alindromic* ***R****epeats* (CRISPR) systems consist of two main components. The first is the eponymous, adaptive CRISPR array, composed of short repetitive DNA sequences (*repeats*) that are interspaced by variable *spacer* sequences derived from foreign genetic material. The second is a set of *CRISPR-associated* (Cas) genes, which encode a more conserved protein machinery that does not directly adapt to individual threats but mediates spacer acquisition, RNA processing, and target interference.

Although CRISPR arrays and Cas genes constitute only a small part of prokaryotic genomes, together they form highly coordinated immune systems whose emergent behavior far exceeds that of their individual components.

To understand how these systems operate and evolve, we first consider the biological and

ecological context in which they arise.

***Prokaryotes*: navigating a dynamic genetic landscape.** Depending on where the DNA is located in the cell, life can be roughly split into two taxonomic classes. *Eukaryotes* are cells where the genetic information is found within an membrane-enclosed nucleus. *Prokaryotes* are (usually) single cells with no nucleus (or other organelles enclosed in membranes) and can be split into two subdomains *bacteria* and *archaea*. Both bacteria and archaea are (usually) small (a few micrometers in length) and present almost everywhere on earth. In terms of estimated total biomass on earth, bacteria and archaea are estimated to contribute 70 and 7 gigatons of carbon (Gt C), respectively, only exceeded by plants (450 Gt C), and far exceeding animals (2 Gt C) (Bar-On et al. 2018).

Bacteria and archaea are essential for sustaining the biosphere and play crucial ecological roles, ranging from ecosystem-level functions including carbon fixation, nitrogen cycling, organic compound turnover, and the maintenance of microbial communities. Beyond these ecosystem-level roles they also fulfill important functions within the microbiota of diverse organisms, including the human gut, mouth and skin. Their capabilities have further been harnessed in a wide range of biotechnological applications, including food processing, wastewater treatment, biofuel production, and the pharmaceutical industry.

However, not all bacterial interactions are positive. A small number of species are pathogenic, causing some of the most severe infectious diseases in humans and animals, whereas archaea appear to be far less frequently associated with parasitic interactions with other life forms.

As a consequence of their involvement in so many processes of the biosphere, interest in prokaryotes, particularly in pathogenic bacteria, was and remains high. Since the beginning the modern era of genomics of prokaryotes by the sequencing of the first complete genomes of bacteria and archaea in 1995 and 1996 (Koonin et al. 2008). The rapid advancement of DNA sequencing technologies has led to an exponentially growing catalog of prokaryote genomes, ushering in a new era of large-scale bioinformatic, statistical and comparative analyses of prokaryotic DNA.

The genetic material of prokaryotes is typically organized into one or more circular *chromosomes* which contain many of the genes essential for their survival. In addition, prokaryotes interact with diverse DNA and RNA elements within and around their genomes. These so called *mobile genetic elements* (MGE) include *(bacterio)phages*, *archaeal viruses*, *plasmids*, and *transposable elements*.

Phages and archaeal viruses are viruses that infect and replicate within bacteria and archaea, respectively. They often have antagonistic relationships with their hosts, reducing fitness or causing cell death. At the same time, MGEs, particularly plasmids and transposable elements, but also viruses, facilitate exchange of genetic material between prokaryotes through *horizontal gene transfer* (HGT). HGT is considered a major driver of prokaryotic evolution and can provide substantial benefits for the hosts, for example, by enabling the acquisition of genes conferring antimicrobial resistance.

The diversity and abundance of microbial life are remarkable: prokaryotes are among the most abundant forms of life on Earth, with an estimated population exceeding 10^{30} cells (Bar-On et al. 2018; Mushegian 2020; Whitman et al. 1998). Even so, they are outnumbered by phages whose estimated abundance exceeds that of bacteria by approximately an order of magnitude, with a total population greater than 10^{31} particles (Bar-On et al. 2018; Hendrix et al. 1999; Mushegian 2020; Suttle 2005).

Thus, beyond adapting to environmental change and interacting with other organisms,

prokaryotes constantly navigate a genomic landscape shaped by MGE, balancing their beneficial contributions with the threat of viral predation.

The Microbial Warzone* or *prokaryotes live in highly competitive environments. The interaction between prokaryotes and viruses occurs at an immense ecological scale. In seawater alone, where phage concentrations are estimated at 10^7 – 10^8 per milliliter, approximately 10^{23} phage infections take place every second (Suttle 2005; Suttle 2007). These interactions have a substantial impact on microbial mortality, with phages estimated to be responsible for 15 to 40% of bacterial cell deaths per day (Danovaro et al. 2011; Suttle 1994; Suttle 2007).

Consequently, bacteria developed a plethora of defense systems (Dy et al. 2014; Georjon et al. 2023; Hampton et al. 2020; Labrie et al. 2010; Millman et al. 2022; Van Houte et al. 2016a). In response, phages and plasmids have evolved diverse counter-strategies to evade or overcome these defenses (Georjon et al. 2023; Hampton et al. 2020; Labrie et al. 2010; Mayo-Muñoz et al. 2024; Van Houte et al. 2016a). This reciprocal adaptation drives a continuous evolutionary arms race between prokaryotes and their MGE, resulting in rapid co-evolution. The discovery of new defense and anti-defense mechanisms is ongoing, with novel systems and strategies continuing to be uncovered (Mayo-Muñoz et al. 2024; Tesson et al. 2022).

The interplay between the numerous bacterial defense systems and anti-defense mechanisms forms a complex network of interactions between prokaryotes and their MGEs, the full extent of which is still difficult to fully resolve. Moreover, bacteria-phage interactions drive rapid evolution through HGT and strong selective pressure, providing a powerful framework for understanding prokaryotic evolution and MGE dynamics (Millman et al. 2022; Tesson et al. 2022; Van Houte et al. 2016a; Westra et al. 2016).

A particularly striking recent finding is that MGEs themselves deploy dedicated countermeasures, or co-opt host-encoded systems, to interfere with competing MGEs (Faure et al. 2019; Fillol-Salom et al. 2022; Hampton et al. 2020; Hargreaves et al. 2014; Koonin et al. 2024; Pinilla-Redondo et al. 2020; Pinilla-Redondo et al. 2022; Rousset et al. 2022; Al-Shayeb et al. 2020). As a result, MGE-MGE interactions introduce an additional layer of evolutionary conflict, operating at an even finer genomic and ecological scale.

Layered defenses against phages. Bacteria do not rely on a single defense mechanism, but instead deploy a multilayered combination of systems acting at different stages of infection.

The first line of defense acts before the phage infiltrates the cell. Bacteria can block phage adsorption by altering or masking surface receptors (Manning et al. 2011; Reyes-Robles et al. 2018; Seed et al. 2014), or prevent the injection of phage genomes into the cell (Dy et al. 2014). Additional mechanisms also limit successful attachment or DNA delivery (Hampton et al. 2020; Van Houte et al. 2016a).

If a phage manages to enter the cell, bacteria deploy intracellular defense systems. For such defense systems to be beneficial, they must solve two core problems. First, they need a sensor that is able to reliably detect an infection. One widespread possibility to sense an infection is to detect the foreign DNA/RNA introduced by the phage. Recognition must be specific enough to avoid autoimmunity by targeting the own genome and to avoid inhibiting beneficial HGT significantly, yet flexible enough to be effective and difficult for invaders to evade. Second, they require *effectors*, tools (proteins) capable of locating and destroying the phage, or, alternatively, killing the infected host before the phage proliferates (Georjon

et al. 2023; Hampton et al. 2020).

Systems inducing the death, or dormancy, of the infected cell on catastrophic infection are called *abortive infection systems*. On a population level, this cell death acts as "altruistic suicide", preventing the spread of the phage and reducing the risk of an epidemic (Dy et al. 2014; Georjon et al. 2023; Hampton et al. 2020; Labrie et al. 2010; Van Houte et al. 2016a).

Other systems can specifically target and destroy or silence the phage DNA/RNA with their effectors (without killing the cell). Because these systems are capable of precisely targeting and manipulating DNA/RNA, they are of exceptional interest in both fundamental biology and biotechnology research. In particular, the study of two prominent and widespread defense systems, *restriction modification systems* (RM systems) and *CRISPR-Cas systems*, which we briefly introduce below, has directly contributed to two Nobel prizes related to gene editing (Nobel Prize Outreach 2026a; Nobel Prize Outreach 2026b).

RM systems distinguish non-self DNA from self DNA by recognizing and methylating (marking) short (4–8 bp) specific sequences (restriction sites) and subsequently cleaving the non-self (non-methylized) DNA via restriction endonucleases (Hampton et al. 2020; Loenen et al. 2014). They are very diverse and present in most (90%) bacterial genomes (Oliveira et al. 2014). A multitude of different RM systems have been identified (reviewed in Loenen et al. (2014)) and various similar systems have been found (Hampton et al. 2020).

Naturally, phages have developed a number of strategies to evade RM systems (Hampton et al. 2020; Samson et al. 2013). For example, they mutate the restriction sites, deactivate the restriction endonucleases, or they "disguise" themselves by becoming methylated by the host (or their own) methyltransferase upon entry of the cell (Hampton et al. 2020; Samson et al. 2013).

The ability of restriction enzymes to precisely cleave DNA laid the foundation for modern genetic engineering (Loenen et al. 2014). In recognition "for the discovery of restriction enzymes and their application to problems of molecular genetics", Werner Arber, Dan Nathans and Hamilton Smith were awarded the Nobel Prize in Physiology or Medicine in 1978 (Nobel Prize Outreach 2026b).

***Adaptive* defense: CRISPR-Cas.** CRISPR-Cas systems share some similarities with RM systems, both systems recognize and cleave foreign DNA. However, CRISPR-Cas is significantly more sequence-specific and can adapt to new threats during the cells lifetime. In essence, CRISPR-Cas defense operates through a three-step process: (I) *adaptation*, (II) *expression*, and (III) *interference*; illustrated in Figure 1.1.

First, CRISPR-Cas systems *adapt* to new threats by capturing small snippets (∼30 bp) of invading MGEs, called *spacers*, and inserting them into CRISPR arrays in the hosts genome. The corresponding sequences in the invading MGEs are known as *protospacers*. Spacer acquisition is largely mediated by the conserved Cas proteins Cas1 and Cas2. Importantly, not all protospacers are equally suitable for acquisition: in many CRISPR-Cas systems, protospacers must be associated with short adjacent sequence motifs known as *protospacer adjacent motifs* (PAMs), which help define the pool of potential acquisition targets.

Next, the spacers within the CRISPR array are transcribed and processed into guide RNAs, called *CRISPR-RNA* (crRNA). Once mature, these crRNAs are incorporated into effector complexes composed of one or more Cas proteins. Then, during the ongoing or a subsequent invasion of the respective MGE, the effector complexes use the crRNA as guide to recognize and cleave target DNA or RNA in a sequence-dependent manner and thereby provide immunity. Successful defense, in turn, can trigger the uptake of new spacers through a process known as *primed* acquisition. Unlike the relatively conserved

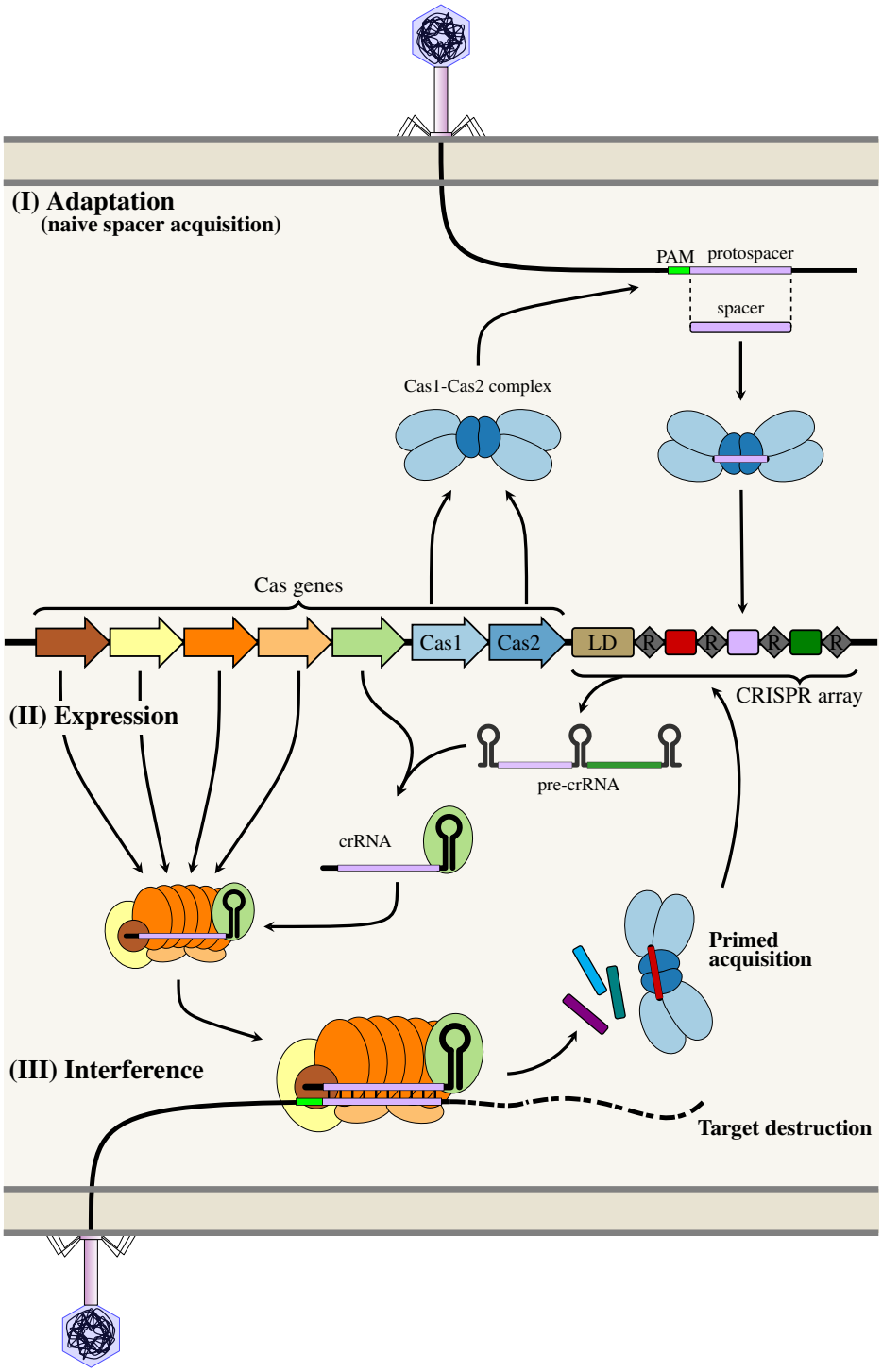


Figure 1.1: Overview of CRISPR-Cas immunity (type I system). The CRISPR-Cas locus (center) encodes the CRISPR array, composed of a leader (LD), spacers (colored rectangles), repeats (diamonds), and Cas genes for crRNA processing (green), interference (brown/yellow/orange), and acquisition (blue). **(I)** A Cas1-Cas2 complex selects a new spacer (lavender) from sequences with a suitable PAM (lime) and integrates it at the leader end. **(II)** The array is transcribed into precursor crRNAs (pre-crRNAs) and processed by Cas proteins (e.g. Cas6) into mature crRNAs, which assemble with other Cas proteins into interference complexes. **(III)** crRNA-guided complexes target and destroy matching DNA. During interference, new substrates for spacer acquisition are generated and integrated into the array (red spacer). The illustration is simplified for clarity; see Chapter 2.2 for a more detailed description of the underlying processes. Adapted from figures in Nussenzweig et al. (2020).

Cas genes involved in adaptation, those responsible for crRNA processing and interference can vary considerably across systems.

Remarkably, spacer acquisition is polarized: new spacers are consistently inserted at one end of the CRISPR array, most commonly associated with a characteristic sequence known as the *leader*. As a result, the array is temporally structured and preserves a record of past infections. In addition to acquiring new spacers, CRISPR arrays also lose spacers over time, most likely due to replication errors within the repeat-spacer structure. Consequently, the array forms a continuously evolving genetic record that functions as an inheritable memory of past encounters, while *adapting* to changing threats through the ongoing processes of spacer acquisition and deletion.

Remark

CRIPSR-Cas systems are one of few clear-cut examples of *Lamarckian evolution* in prokaryotes, since MGEs (the environment) stimulate and directly modify the genome (acquisition of spacers), the modifications directly affect the MGE (through interference), and the CRISPR arrays, filled with new spacers during the prokaryotes lifetime, are subsequently inherited (Koonin et al. 2009).

Although CRISPR-Cas is now widely recognized as a powerful gene editing tool, its original role as an adaptive immune system and history of discovery is relatively unknown outside the scientific community (Ishino et al. 2018; Lander 2016; Morange 2015a; Morange 2015b). We provide a short excursion into the discovery CRISPR-Cas as an adaptive immune defense mechanism in prokaryotes and its first uses as gene editing tool. For more details, see one (or more) of the numerous reviews documenting the discovery of CRISPR-Cas and CRISPR-Cas technology (Barrangou et al. 2013; Ishino et al. 2018; Lander 2016; Mojica et al. 2016; Morange 2015a; Morange 2015b).

Excursion: A short history of CRISPR-Cas

The discovery of CRISPR and associated genes begins in laboratories in Japan at the end of the 1980s (Ishino et al. 1987; Nakata et al. 1989). In 1987, Ishino et al. published a paper discussing the genes involved in the metabolism of *Escherichia coli*. They identified an "unusual structure" in the non-coding region of a *E. coli* (K-12) laboratory strain composed of five highly homologous 29-nucleotide sequences arranged as direct repeats, separated by 32-nucleotide spacers that were all of equal length yet highly diverse – a structure now known as a CRISPR array (Ishino et al. 1987). They found this structure so noteworthy that they included it in their discussion stating that: "so far, no sequence homologous to these has been found elsewhere in prokaryotes, and the biological significance of these sequences is not known" (Ishino et al. 1987), a mystery that would persist for many years.

Advances in DNA sequencing in the 1990s allowed rapid access to (complete) genomes, starting with the first complete genome of a free-living bacterium in 1995 (Ishino et al. 2018; Mojica et al. 2016). Interest in CRISPR grew substantially as similar repeat structures were identified in an increasing number of bacterial and archaeal species, reaching at least 19 species by 2000 (Hermans et al. 1991; Mojica et al. 1993; Mojica et al. 1995; Mojica et al. 2000).

In 2002, Jansen et al. showed that most CRISPR arrays are accompanied by nearby genes which they termed Cas genes. They proposed that the Cas genes might be

involved in DNA repair, recombination, regulation and chromosome segregation (Ishino et al. 2018; Jansen et al. 2002; Makarova 2002). Other Cas genes, e.g. Cas1 and Cas2, which are now known to be largely responsible for spacer acquisition, were found to have no similarity to functional domains of known proteins. After consulting Francisco Mojica, Jansen et al. introduced the now universally adopted name CRISPR, replacing the many earlier abbreviations (SRSR, DR, TREP, LTRR, SR, LCTR, SPIDR) (Barrangou et al. 2013; Jansen et al. 2002).

In 2005, a breakthrough in CRISPR-Cas research occurred that was largely underappreciated at the time, as all authors faced rejections by prominent journals (Ishino et al. 2018; Lander 2016; Mojica et al. 2016; Morange 2015a). Three groups nearly simultaneously discovered that the sequences interspersed between CRISPR repeats correspond to extrachromosomal bacteriophage and plasmid DNA (Bolotin et al. 2005; Mojica et al. 2005; Pourcel et al. 2005). Based on a review of available data and literature, Mojica et al. (2005) and Pourcel et al. (2005) proposed that the presence of matching spacer sequences in CRISPR arrays provides immunity against phages and plasmids carrying the corresponding sequences. This observation was confirmed by Bolotin et al. (2005), who showed that the degree of phage resistance correlates with the number of corresponding spacers. Furthermore, Mojica et al. (2005) and Pourcel et al. (2005) proposed that CRISPR could acquire new spacers from foreign DNA, forming a kind of memory of past encounters. It was also observed that the acquisition of new spacers is polarized, with spacers being added at one end of the array (Lillestøl et al. 2006; Pourcel et al. 2005). Both Pourcel et al. (2005) and Bolotin et al. (2005) hypothesized that Cas genes are directly involved in the acquisition and function of CRISPR system.

Subsequent experiments by Barrangou et al. (2007) and Marraffini et al. (2008) confirmed the defense capabilities against phages and plasmids, respectively. These studies demonstrated the essential role of Cas genes in the defense process and confirmed that new spacers are added to CRISPR arrays in a polarized manner. Notably, the conferred immunity was lost immediately when matching spacers were removed from the CRISPR array (Barrangou et al. 2007). Metagenomic studies provided the first evidence that CRISPR arrays are highly dynamic, showing that spacer content can change within only a few months through both spacer acquisition and deletion events (Andersson et al. 2008; Horvath et al. 2008). At the same time, it became apparent that such rapid turnover may be necessary, as phages can evade CRISPR-mediated immunity through recombination of targeted sequences (Andersson et al. 2008).

With the discovery of the core functionality of CRISPR-Cas, research increasingly focused on understanding the underlying biochemical mechanisms. Due to their conservedness and limited role in interference, Cas1 and Cas2, were quickly implicated to be involved in acquisition of new spacers (Barrangou et al. 2007; Brouns et al. 2008) which was confirmed soon after (Nuñez et al. 2014; Yosef et al. 2012). Additionally, Brouns et al. (2008) discovered that (in type I systems) multiple Cas proteins assemble into an interference complex that uses crRNA, transcribed from individual spacers in the array, to target foreign DNA.

Soon after the three breakthrough papers, Makarova et al. (2006) discussed the similarity of Cas genes to eukaroytic RNA interference (RNA silencing) and first proposed its use for gene silencing. The concept of employing CRISPR-Cas for genome modification was further supported by Garneau et al. (2010), who experimentally demonstrated

that Cas proteins interfere with the foreign DNA by cleaving DNA at specific sites.

Makarova et al. (2011) proposed a classification system that grouped CRISPR-Cas systems into three classes and ten subtypes based on the composition and evolutionary relationships of Cas genes and repeats, a framework that still underlies the current classification system. Their discussion of class II systems contributed to the recognition of their potential for genome editing applications.

A group led by Emmanuelle Charpentier, and independently Sapranauskas et al., had shown that functional interference in type II(-A) systems requires only one Cas protein, Cas9 (Deltcheva et al. 2011; Sapranauskas et al. 2011). They also showed that the only other essential component is a type-specific *trans*-encoded small RNA (tracrRNA), which is involved in crRNA maturation (alongside RNase III) (Deltcheva et al. 2011). Consequently, type II systems represent a more streamlined interference architecture compared with the multi-protein complexes of type I systems.

Charpentier then initiated a collaboration with Jennifer Doudna, and together they demonstrated that Cas9-crRNA-tracrRNA complexes could cleave DNA *in vitro* (Jinek et al. 2012), a finding that was confirmed by Gasiunas et al. (2012). Furthermore, the group around Charpentier and Doudna showed that Cas9-crRNA-tracrRNA-complexes could be repurposed as "programmable" genome editing tools, with target specificity determined simply by swapping the crRNA-tracrRNA guide (Jinek et al. 2012).

In 2013, the first reports of successful CRISPR-Cas9-based genome editing in human and mouse cells were published (Cong et al. 2013; Ishino et al. 2018; Mali et al. 2013). These proof-of-concept studies demonstrated that DNA could be altered with unprecedented ease with CRISPR-Cas, establishing that CRISPR-Cas would become a staple of the genome editing toolbox. Starting in 2012, patents for CRISPR-Cas9-based eukaryotic genome editing were filed by the involved universities and researchers. These filings have led to still ongoing patent disputes (Nature Biotechnology 2025).

In recognition of the transformative impact of CRISPR-Cas on the precision and accessibility of genome editing, Emmanuelle Charpentier and Jennifer Doudna were awarded the 2020 Nobel prize in Chemistry "for the development of a method for genome editing" (Nobel Prize Outreach 2026a).

Applications of CRISPR-Cas across biotechnology and microbiology. CRISPR-Cas has become one of the most widely used tools for genome editing. The CRISPR-Cas9 protein complex remains the most broadly applied system, although alternative Cas effectors have also been successfully adapted for genome engineering. Today, CRISPR-Cas technologies are ubiquitous in genome editing due to their versatility, programmability, and efficiency (reviewed in Wang et al. (2023)).

CRISPR-based gene knockouts and large-scale screening are widely used to study gene function in vivo. Beyond double-strand breaks, catalytically inactive or modified Cas proteins enable transcriptional regulation, targeted single-strand nicking, and gene activation or repression. Fusion of Cas proteins with effector domains further extends their applications to base editing, sequence insertion, imaging, and diagnostics (Wang et al. 2023). Despite this versatility, major challenges remain, including efficient delivery in vitro and in vivo, cellular uptake and nuclear access, target specificity, and minimizing off-target effects (Wang et al. 2023). For details about CRISPR-Cas as genome editing tools, their protein structure, and applications, we refer the reader to available reviews (Foss et al.

2019; Komor et al. 2017; Koonin et al. 2023; Shivram et al. 2021; Villiger et al. 2024; Wang et al. 2023; Wang et al. 2022).

The impact of CRISPR technology as a tool for genome editing and screening is unmistakable. Moreover, current applications likely represent only the beginning, as CRISPR-based genome editing is expected to play an increasingly important role in biotechnology and medicine in the coming years (Wang et al. 2023).

Beyond genome editing, they also provide a powerful framework for studying bacterial evolution and bacteria-phage interactions, offering insights into the ecological and evolutionary dynamics that shape microbial communities.

Building on this dual role as both a biological system and an analytical tool, CRISPR-Cas systems and their influence on microbiomes have important applications across a wide range of domains, including fermented food production, human and animal health (e.g. livestock and aquaculture), the spread of antimicrobial resistance, wastewater treatment, and pharmaceutical and medical contexts, including emerging approaches such as CRISPR-Cas- and phage-based therapies aimed at the targeted manipulation of bacterial communities (Avershina et al. 2026; Benz et al. 2025; Dedrick et al. 2019; Ganote et al. 2025; Guerrero et al. 2021; Laanto et al. 2017; Lam et al. 2019a; Landor et al. 2026; Li et al. 2021; López-Beltrán et al. 2024; Martínez Arbas et al. 2021; O'Sullivan et al. 2019; Palmnäs-Bédard et al. 2023; Pertziger et al. 2025; Seed et al. 2014; Somerville et al. 2022a; Somerville et al. 2022b; Wang et al. 2023).

Accordingly, extensive research has focused on CRISPR-Cas systems both as tools for genetic engineering and as model systems for studying the molecular mechanisms of Cas effector complexes, including their roles in interference and spacer acquisition. In addition, the ecological role of CRISPR-Cas systems and their impact on bacteria-phage interactions have been the subject of substantial experimental and theoretical investigation.

In this context, a mechanistic understanding of CRISPR array evolution, viewed as spacer repertoires shaped by repeated acquisition and deletion events, is particularly important, as it underpins both microbiome dynamics and Cas system function, and highlights the need for further systematic study of array-level evolutionary processes.

1.1 Thesis outline and structure

Despite its importance, several key aspects of CRISPR array evolution remain poorly understood. The following chapters address some of these aspects, which are briefly introduced below.

(1) While comparative analysis of spacer content across closely related strains has long been recognized as valuable for strain subtyping and for understanding prokaryotic ecology, the available computational tools for efficiently analyzing CRISPR arrays and reconstructing their ancestral histories, while accounting for their characteristic deletion and polarized acquisition mechanisms, remain limited.

(2) Spacer acquisition has been studied in considerable detail, whereas the biological mechanisms and bioinformatic characterization of spacer deletions remain comparatively poorly understood.

(3) Predicting which end of a CRISPR array serves as the site of new spacer insertion, as well as determining related features such as leader location, transcription orientation, and their interrelationships, remains a challenging and unresolved problem.

(4) Finally, although individual CRISPR systems have been investigated extensively, the co-evolution and functional interactions of multiple CRISPR arrays and systems within a single genome have received relatively little attention.

The manuscript is organized into three main parts: an introduction to CRISPR-Cas systems and CRISPR array evolution (Chapters 2–4); the three primary contributions of this work, presented in publications Fehrenbach et al. (2024), Fehrenbach et al. (2025), and Fehrenbach et al. (2026) (Chapters 5–7); and a final discussion of the main findings, potential applications, and future perspectives (Chapter 8).

We begin by giving an overview over the components and terminology of CRISPR-Cas systems in Chapter 2. Followed by a description of the mechanisms underlying CRISPR adaptation, expression and interference, highlighting the extensive diversity among systems of different types. We then review relevant theoretical and experimental work on differences between CRISPR-Cas systems, their ecological roles, and their involvement in bacteria-phage interactions.

In Chapter 3, we review the current state of knowledge on CRISPR spacer array evolution, shaped by spacer acquisitions, deletions, and more complex evolutionary events. We further summarize relevant literature, with a particular focus on experimentally observed and theoretically predicted dynamics that any evolutionary model or reconstruction method must account for and that are reflected in the available data.

Building on these insights, Chapter 4 develops a mathematical framework for modeling CRISPR arrays as sequences of spacers and repeats shaped by acquisitions and deletions. We review existing models of array evolution, discuss their underlying assumptions, and introduce extensions (particularly for deletions) that better capture experimentally observed dynamics. We then derive both simplified and precise likelihood functions, incorporate bias corrections, and formulate a likelihood ratio test for comparing alternative deletion models. This framework further enables the reliable estimation of deletion rates, and indirectly acquisition rates, by analyzing ancestral reconstructions of spacer array evolution along phylogenetic trees. This chapter incorporates material from Fehrenbach et al. (2024) and its supplementary information. Readers less interested in the details of the mathematical modeling may choose to skip this chapter on a first reading and consult it during Chapter 5 as needed for additional technical detail and methodological depth.

Chapter 5, adapted from Fehrenbach et al. (2024), applies the theoretical framework of Chapter 4 to address (1) by developing *SpacerPlacer*, a tool for ancestral reconstruction of CRISPR array evolution, using the introduced evolutionary models and exploiting the polarized acquisition of spacers to produce an accurate reconstruction. We validate *SpacerPlacer* through simulations and apply it to a large real-world dataset to investigate (2) the rates and mechanisms of spacer deletions using the mathematical framework introduced in Chapter 4. We gain insights into deletion mechanisms and observe unexpected patterns in deletion frequency along the array.

Chapter 6, adapted from Fehrenbach et al. (2025), addresses (3) inferring the acquisition orientation of CRISPR arrays, which strongly impacts *SpacerPlacer* reconstructions. The existing methods, relying on repeats, leaders, or Cas genes, are often unreliable. We develop a method for predicting acquisition orientation based on reconstructions from *SpacerPlacer*, the mathematical framework presented in Chapter 4, and an associated likelihood ratio. We benchmark *CRISPR-evOr* on simulated and real datasets, comparing it to existing orientation tools and approaches, and reveal new insights into orientation patterns across Cas types.

Lastly, in Chapter 7, adapted from Fehrenbach et al. (2026), we examine (4) the (co-)occurrence

and diversity of multiple CRISPR arrays and systems within and across genomes. Using a comprehensive CRISPR array dataset, we statistically analyze Cas type co-occurrence, identifying types that frequently co-occur and others that appear to avoid each other. We examine array-Cas distances, orientation, lengths, and repeat similarity, revealing significant type-specific differences. We also investigate spacer acquisition patterns across genomes with multiple arrays, leveraging the mathematical relationship between acquisitions, deletions, and array length. Overall, our findings highlight the complex interactions, cooperation, and competition among CRISPR systems within a single host.

In Chapter 8, we summarize the main findings of this thesis, discuss their broader implications, and outline potential extensions of *SpacerPlacer* and the statistical framework, including applications to integrons, which share structural and evolutionary similarities with CRISPR arrays. Finally, we place the results in a broader biological and methodological context.

2

CRISPR-Cas systems: form, function and ecology

In this chapter, we introduce the abundance, components and terminology of CRISPR-Cas systems, followed by an overview of their function as adaptive immune systems in prokaryotes, and conclude with a discussion of relevant literature and CRISPR-Cas' ecological roles. Readers interested in a more comprehensive treatment are referred to the extensive body of review literature covering CRISPR-Cas biology, molecular mechanisms, ecological roles, evolutionary dynamics, and related topics (Barrangou et al. 2013; Bonsma-Fisher et al. 2023; Garrett 2021; Hille et al. 2018; Jackson et al. 2017; Koonin et al. 2017; McGinn et al. 2019; Mohanraju et al. 2016; Nussenzweig et al. 2020; Van Houte et al. 2016a; Wang et al. 2023; Wang et al. 2022; Westra et al. 2016).

Abundance of CRISPR-Cas systems. CRISPR-Cas systems are relatively abundant in bacteria, occurring in approximately 40–50% of species, and particularly common in thermophiles, where around 90% carry complete systems (Makarova et al. 2025; Weinberger et al. 2012a).

Archaea were previously thought to harbor CRISPR systems at very high frequencies, with estimates exceeding 80% (Hille et al. 2018; Makarova et al. 2015; Pourcel et al. 2019). More recent analyses by Makarova et al. (2025), which account for sampling bias, report a lower prevalence of approximately 52% in archaea. This reduction is largely attributed to increased sampling of mesophilic archaea. Indeed, CRISPR-Cas systems are less common in mesophiles and psychrophiles, occurring in fewer than 46% and 38% of genomes, respectively, whereas they remain highly prevalent in (hyper)thermophiles, where more than 85% of archaea carry complete CRISPR systems (Makarova et al. 2025).

The discrepancy between mesophilic and thermophilic organisms has been suggested to arise from larger population sizes and higher viral mutation rates in mesophiles. Population genetic models indicate that elevated mutation rates and increased genetic diversity can reduce the effectiveness of CRISPR immunity, potentially leading to the loss of CRISPR systems when their maintenance costs outweigh their benefits (Bonsma-Fisher et al. 2023; Iranzo et al. 2013; Weinberger et al. 2012a). *Pseudomonas aeruginosa*, which inhabits a broad temperature range, has been shown to exhibit increased CRISPR adaptation at lower temperatures, possibly due to higher phage pressure under these conditions (Høyland-Kroghsbo et al. 2018).

2.1 Overview and terminology

CRISPR-Cas systems are composed of two components in the genome (see Figure 1.1):

1. A *CRISPR array* or *spacer array*: a non-coding component that serves as the adaptive "memory" of the CRISPR system. It consists of repeats separated by *spacers*, which are derived from foreign genetic material. In most types, the array is flanked by a *leader sequence* that directs and facilitates the integration of new spacers and promotes expression. Some systems possess multiple associated CRISPR arrays.

2. A *Cas locus* or *Cas cassette*: they constitute the "toolbox" of CRISPR systems, comprising protein-coding Cas genes and, in some cases, additional associated genes and RNAs. The composition of the Cas locus largely determines the system's *Cas type* and functional capabilities (Figure 2.1). The encoded Cas proteins assemble into *effector complexes* or *cascades* that mediate the systems active function, i.e. spacer acquisition, crRNA maturation and interference.
   In the literature, CRISPR arrays are often considered part of the Cas locus due to their frequent genomic proximity to Cas genes. In this work, however, we often treat CRISPR arrays and Cas loci as distinct entities, while analyzing them jointly for Cas type inference.

2.1.1 CRISPR array

As the name implies, CRISPR (**C**lustered **R**egularly **I**nterspaced **S**hort **P**alindromic **R**epeats) arrays consist of short repeats (in Chapter 7: mean 32 bp, median 30 bp, range 21–56 bp) and short spacers (mean 34 bp, median 32 bp, range 18–55 bp, with occasional larger outliers). Within a given array, both repeat and spacer lengths are typically highly conserved. The number of repeat-spacer units per array is generally modest (mean 15) but varies substantially across Cas types and genomes, ranging up to several hundred; see Section 7.3.4 and Supplementary Tables S1–S4 in Fehrenbach et al. (2026).

Repeats within a CRISPR array often accumulate mutations and therefore differ slightly along the array (Mitrofanov et al. 2026). As a result, repeat sequences are typically represented by a single *consensus repeat* derived from all individual copies. Consensus repeat sequences vary substantially across CRISPR-Cas types, as discussed in Chapter 7. As suggested by the P in CRISPR, many, but not all (Kunin et al. 2007), arrays have repeats that are partly palindromic and can form secondary structures (Hille et al. 2018).

Most arrays (87% in archaea, 76% in bacteria) have a characteristic *leader sequence* located on one of its ends with a length of 60 to a few hundred base pair (Alkhnbashi et al. 2016; Buyukyoruk et al. 2025). Leaders are essential for CRISPR array function: they contain motifs that direct the polarized insertion of new spacers (Barrangou et al. 2007; Kim et al. 2019; McGinn et al. 2016; Pougach et al. 2010; Wei et al. 2015a; Wright et al. 2016; Xiao et al. 2017a; Yosef et al. 2012) and house promoters that drive transcription of the array into guide RNAs (Alkhnbashi et al. 2016; Liao et al. 2022; Martynov et al. 2017; Wei et al. 2015a; Weissman et al. 2018b).

Due to their relevance of adaptation and expression, the two ends of a CRISPR array are designated *leader-proximal* or *leader-distal* according to their position relative to the leader.

2.1.2 Cas locus and classification

Efforts to classify and group CRISPR-Cas systems began shortly after their discovery (Haft et al. 2005; Jansen et al. 2002; Makarova et al. 2006), and the first comprehensive classification scheme was introduced by Makarova et al. (2011). Since then, the group around Kira Makarova and Eugene Koonin has substantially expanded and refined this framework (Makarova et al. 2015; Makarova et al. 2020; Makarova et al. 2025; Shmakov et al. 2015), while preserving its fundamental structure.

Classifying CRISPR-Cas systems is challenging because they lack universal genetic markers and both Cas loci and CRISPR arrays evolve quickly. As a result, they are primarily grouped into two *classes* and *Cas types* (or *CRISPR types*) based on the Cas gene content of the Cas locus (Figure 2.1). Additional phylogenetic and comparative criteria are used to define *subtypes* (Makarova et al. 2011; Makarova et al. 2015; Makarova et al. 2020; Makarova et al. 2025; Shmakov et al. 2015).

In practice, the CRISPR array(s) closest to a Cas locus, if one is present, are assigned the Cas type inferred from the associated Cas genes. Both the CRISPR array and the Cas genes are required for a functional CRISPR system, and therefore typically genetically linked and co-inherited as a unit. In our datasets, following community consensus, we consider a distance of less than 10 kb between an array and a Cas locus sufficient for confident Cas type assignment. In some cases, additional evidence, such as repeat sequence similarity, genomic context, phylogenetic relationships, and experimental data is used to link a CRISPR array to a specific Cas type and locus.

Currently, CRISPR-Cas systems are separated into two *classes* with 7 types and a total of 46 subtypes and even more variants (Makarova et al. 2025). Figure 2.1 shows a breakdown of characteristic Cas genes across types and their respective functions. The classes are primarily separated by what type of interference complexes they employ. Class 1 systems (containing Cas types I, III, IV, and VII) employ Cas complexes for interference that are composed of multiple Cas proteins. Class 2 systems (containing Cas types II, V, and VI), however, employ a large single effector protein for interference (and crRNA expression) (Makarova et al. 2011; Makarova et al. 2015; Makarova et al. 2020; Makarova et al. 2025).

Cas types are defined based on type-specific Cas genes and the composition of their effector complexes (Figure 2.1). Each type is further divided into subtypes, e.g. type I includes I-A, I-B, . . . , I-F. Subtype classification is less formal and is based on Cas locus architecture and gene composition, as well as sequence similarity and phylogenetic relations of conserved Cas genes (especially Cas1, Cas2), together with the presence and location of CRISPR arrays. These gene- and sequence-based assignments are further supported by experimental evidence on system function where available (Makarova et al. 2015; Makarova et al. 2020; Makarova et al. 2025).

Note that subtype and variant classification is not always robust and would benefit from clearer criteria (Makarova et al. 2020). For instance, Makarova et al. (2020) point out that in class 2 systems, new subtypes are defined based on statistically significant sequence divergence in effector complexes relative to previously classified systems, which introduces a degree of arbitrariness.

2.1.3 Detecting CRISPR systems

Identifying and accurately classifying CRISPR systems is a complex task. In most cases, the process begins with detecting the CRISPR array itself, which is comparatively straight-

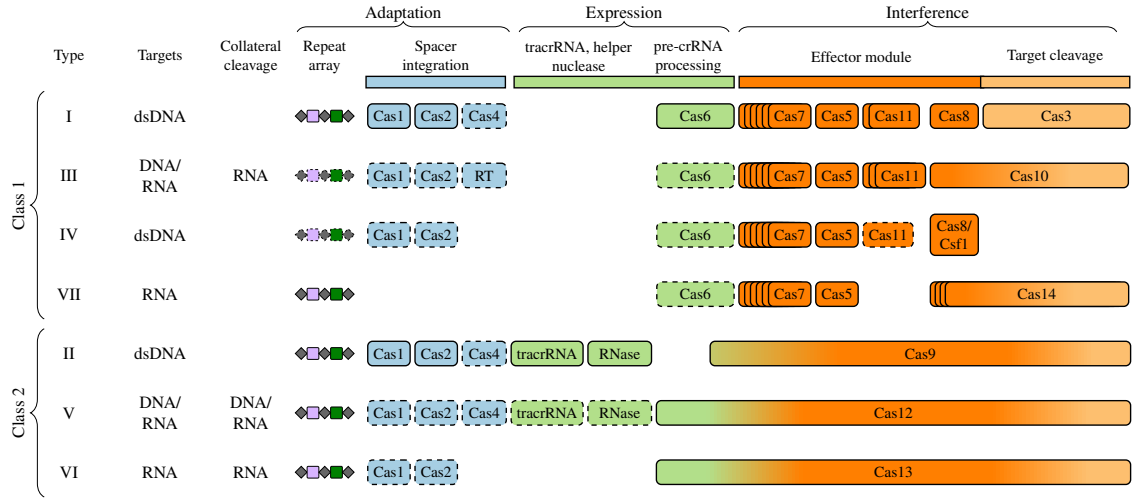


Figure 2.1: Targets and modular organization of CRISPR systems by type. Overview of the targets and typical genetic, structural, and functional organization of CRISPR-Cas systems across Cas types (adapted from Fig. 1 in Makarova et al. (2025)). Some systems exclusively target double-stranded DNA (dsDNA), whereas others target combinations of dsDNA, single-stranded DNA, and RNA (particularly variable within type V). Types III, V, and VI can additionally induce non-specific collateral cleavage. Colors indicate functional roles of Cas proteins. Dashed outlines of CRISPR arrays and Cas genes denote components that are dispensable and/or subtype-specific. Multiplicity indicates that the effector complex contains multiple copies of the respective Cas protein. This overview is not exhaustive; see Makarova et al. (2025) for a comprehensive classification.

forward because of its repeated nature.

The two most reliable and widely used tools for detecting CRISPR arrays and Cas genes in long-read data and assembled genomes are *CRISPRCasFinder* (Couvin et al. 2018) and *CRISPRidentify* (Mitrofanov et al. 2021).

Both tools begin by identifying repeated elements in the input DNA within specified constraints on repeat length and inter-repeat distance. Candidate arrays are then assigned confidence levels based on repeat and spacer properties, sequence similarity, and other curated features. In CRISPRCasFinder, this classification is performed using a manually curated scoring function, based on selected features, whereas CRISPRidentify applies a machine learning model trained on curated datasets.

Cas genes are subsequently identified in both pipelines: both CRISPRCasFinder and the Cas detection module of CRISPRidentify, *CRISPRcasIdentifier* (Padilha et al. 2020), rely on *Prodigal* (Hyatt et al. 2010) to predict putative coding sequences. Based on current classification consensus (Makarova et al. 2020; Makarova et al. 2025), Cas types are then inferred from Cas gene content using representative Hidden Markov Model profiles in CRISPRCasFinder and machine learning models in CRISPRcasIdentifier.

A multitude of other (older) tools exist, sometimes partly embedded in the newer tools, these include: *CRISPRCasTyper* (Russel et al. 2020), *CRISPRDetect* (Biswas et al. 2016), *PILER-CR* (Edgar 2007), *MacSyFinder* (Abby et al. 2014), *CRISPRdisco* (Crawley et al. 2018), *CRT* (Bland et al. 2007) and, derived from CRT, *minced* (`https://github.com/ctSkennerton/minced`).

Recently, it has been proposed to use conserved sequences in the leader for detection and classification of CRISPR systems, particularly for short or degenerate arrays (Buyukyoruk et al. 2025). However, leaders are difficult to identify since they have few universal

features and sequence motifs, and thus vary substantially across repeats, Cas type and species (Alkhnbashi et al. 2016; Buyukyoruk et al. 2025).

Other tools are designed for metagenomic reads, such as *MCAAT* (Talibli et al. 2025) and *CRASS* (Skennerton et al. 2013), as well as approaches based on deep genomic foundation models (Schröder et al. 2026). However, detection becomes unreliable for very short reads, and reconstructing array order and orientation remains challenging due to the repetitive structure of CRISPR arrays.

Many CRISPR-Cas systems are collected in the comprehensive database *CRISPRCasdb* (Pourcel et al. 2019), which continues the earlier *CRISPRdb* (Grissa et al. 2007). They collected complete genomes from NCBI and analyzed them using CRISPRCasFinder to identify CRISPR arrays and Cas cassettes (Pourcel et al. 2019). The database has been regularly updated, most recently in April 2022.

All datasets of CRISPR arrays (and Cas loci) considered in this work were obtained by applying CRISPRidentify or CRISPRCasFinder on whole genomes from GenBank (Benson et al. 2012), or collecting CRISPRCasFinder results from CRISPRCasdb.

2.1.4 Abundance of Cas types

Class 1 systems are more abundant in bacteria and archaea than Class 2 systems. In particular, type I, the most prevalent Cas type, accounts for approximately half of all CRISPR systems and arrays in recent large-scale analyses (Fehrenbach et al. 2026; Makarova et al. 2025). Together with recent studies on interactions between coexisting CRISPR systems, this suggests that type I systems act as "workhorse" interference modules: they are widespread, typically adaptation-competent, and can also support systems of other types in trans, whereas other systems appear more specialized or function as fallback components; see Section 2.3.4.

The remaining Cas types occur at substantially lower and more comparable frequencies. Makarova et al. (2025) report type III as the second most abundant system, followed by type II, with types V, IV, VI, and VII being progressively rarer. The abundance of individual subtypes can differ considerably. For example, type I-F alone accounts for approximately a quarter of all Cas loci, followed by type VI-B with 15%, in the dataset analyzed in Chapter 7 (Supplementary Table S2 in Fehrenbach et al. (2026)). The most recently identified subtypes (and type VII) are extremely scarce, collectively accounting for only about 0.3% of the CRISPR-Cas repertoire (Makarova et al. 2025). Due to their recent discovery and low abundance, most of these rarer (sub)types are not included in our datasets.

In our datasets, we obtain broadly consistent results but also identify a substantial number of CRISPR arrays located more than 10 kb away from their associated Cas loci, in particular for type V and the abundant type VI-B systems; see Section 7.3, and Supplementary Table S1, S2 and S4 in Fehrenbach et al. (2026). Overall, approximately half of all high-confidence CRISPR arrays can be classified as *distant* relative to their Cas loci. Of these, around 40% are *orphan* arrays, i.e. arrays found in genomes that do not encode *any* Cas genes.

These distant and orphan arrays exhibit substantial heterogeneity in repeat sequences compared to proximal arrays and, for most Cas types, tend to be comparatively short; see Supplementary Fig. S15 (Fehrenbach et al. 2026). This inhomogeneity may reflect misclassification or non-functional remnants of lost CRISPR-Cas systems or duplicated arrays (Buyukyoruk et al. 2023; Zhang et al. 2017). Retaining such arrays may be beneficial, as they are likely inexpensive to maintain and could serve as potential backups

that compensate for the loss or degradation of other crucial functional systems within the genome.

Note that many genomes encode multiple CRISPR-Cas systems, often of different types. In addition, Cas loci are frequently associated with multiple distinct CRISPR arrays, and some subtypes appear to preferentially maintain at least two arrays. The co-occurrence of diverse CRISPR system types, as well as the presence and relative activity of multiple arrays per Cas locus, are central themes and key findings in Chapter 7 and discussed in Section 7.3.

The distribution of Cas types across bacteria and archaea, as well as across phyla, genera, and species, is highly non-uniform. Makarova et al. (2025) report that some subtypes are widely distributed across many phyla, including I-B, I-C, I-E, II-C, and III, whereas others show strong preferences for either bacteria or archaea, or are restricted to only a few phyla, such as I-A, I-D, II-A, and II-B. In our datasets, we observe similarly pronounced differences in abundance across genera and species; see Supplementary Figure S7 in Fehrenbach et al. (2024). However, these patterns are likely heavily influenced by sequencing and sampling biases.

2.2 Mechanisms of CRISPR-Cas immunity

We give a short overview over the (molecular) mechanisms underlying CRISPR defense and state some of the differences between Cas types. As summarized before, CRISPR-Cas defense can be condensed into three subsequent steps:

(I) *Adaptation* or *spacer acquisition*: the uptake of new spacers collected from intruding MGE and insertion into the CRISPR array by Cas1-Cas2 complexes.

(II) *Expression*: the processing of spacer DNA into crRNA that serve as guides for *interference complexes* or *cascades* composed of single or multiple Cas proteins.

(III) *Interference*: the matching of crRNA equipped interference complexes to fitting targets, i.e. protospacers, and subsequent cleaving of the invading DNA or RNA. In a process called *priming*, interference stimulates spacer acquisition and loops back into (I).

See Figure 1.1 for a graphical representation of the steps involved in CRISPR-Cas immunity (for a type I system).

2.2.1 (I) Spacer acquisition

Across all Cas types, Cas1-Cas2 effector complexes composed of (multiple) Cas1 and Cas2 proteins are responsible, and often the only required components (Datsenko et al. 2012; Yosef et al. 2012), for selecting new spacers and integrating them into the CRISPR array (He et al. 2018; Hille et al. 2018; McGinn et al. 2019; Nuñez et al. 2014; Wright et al. 2017; Xiao et al. 2017a). Consequently, Cas1 and Cas2 are highly conserved and present in the Cas loci of most systems, particularly types I and II.

In contrast, some subtypes of III, IV, V, and VI lack Cas1 and Cas2 (Makarova et al. 2025); see Figure 2.1. These subtypes are generally considered adaptation-deficient, yet many can still acquire spacers by recruiting Cas1 and Cas2 in trans from co-occurring CRISPR-Cas

systems (Deecker et al. 2020; Erdmann et al. 2012; Ganguly et al. 2024; Hoikkala et al. 2021; Nickel et al. 2013; Pinilla-Redondo et al. 2020; Silas et al. 2017).

Depending on the Cas type, additional Cas genes are required for (effective) spacer acquisition as indicated in Figure 2.1. One such gene is the relatively conserved Cas4 (Makarova et al. 2025). Although it is not indispensable for acquisition, it is likely critical for many systems ensuring the selection of functional spacers in the right length and orientation (Kieper et al. 2019; Shiimori et al. 2018).

In type II-A, both an ancillary gene Csn2 and the interference complex composed of Cas9 and tracrRNA have been found to be critical for acquisition systems in most cases (Barrangou et al. 2007; Heler et al. 2015; Makarova et al. 2015; Wei et al. 2015b; Wright et al. 2016). RNA targeting type III (González-Delgado et al. 2019; Schmidt et al. 2018; Silas et al. 2016; Zhang et al. 2022) and VI (Molina-Sánchez et al. 2024) systems use reverse transcriptase (fused with Cas1) to enable the integration of RNA-derived spacers into the array.

Despite substantial progress, many questions remain regarding the (molecular) mechanisms of spacer acquisition, particularly in adaptation-competent systems of types III (González-Delgado et al. 2019; Schmidt et al. 2018; Silas et al. 2016; Zhang et al. 2022), V (Wu et al. 2022) and VI (Molina-Sánchez et al. 2024), for which experimental evidence of the mechanisms is still limited (Nussenzweig et al. 2020). Thus, we focus primarily on spacer acquisition in Cas types I and II, the best-characterized systems, highlighting notable differences between types where relevant.

Spacer selection, protospacer adjacent motifs (PAM) and priming

CRISPR systems face key challenges that shape the selection of (proto)spacers in particular:

(i) Acquiring spacers and maintaining them in the CRISPR array carries an inherent risk of autoimmunity, as the crRNA guided interference complexes could mistakenly target and cleave the CRISPR array.

(ii) CRISPR systems can acquire self-targeting spacers from the host genome, which are useless or even harmful (Levy et al. 2015; Paez-Espino et al. 2015; Yosef et al. 2012), so spacer selection must be biased towards foreign genetic elements.

(iii) Highly specific targeting carries a drawback, as MGE can escape CRISPR-based interference by (point) mutation and recombination of the protospacers (Cady et al. 2012; Deveau et al. 2008; Rao et al. 2016; Semenova et al. 2011).

CRISPR systems have evolved multiple strategies to address these challenges during the process of spacer selection.

To prevent autoimmunity (i), DNA-targeting CRISPR–Cas systems (types I, II, IV, and V) discriminate self from non-self DNA by requiring short (1–5 nt) *protospacer adjacent motifs* (PAMs) positioned next to the protospacer. Recognition of a PAM enables crRNA binding and, if sufficient complementarity is present, licenses DNA cleavage (Brouns et al. 2008; Deveau et al. 2008; Hille et al. 2018; Leenay et al. 2017; Mojica et al. 2009; Semenova et al. 2011; Shah et al. 2013; Vink et al. 2021).

In some systems, such as type I-E, the Cas1-Cas2 adaptation complex selects protospacers containing a compatible PAM during spacer acquisition (Vink et al. 2021; Wang et al. 2015). In contrast, type II systems do not strictly select for a PAM at the acquisition stage; instead, the interference effector Cas9 defines the PAM requirement and thereby shapes spacer selection (Heler et al. 2015; Vink et al. 2021).

Consequently, the PAM patterns observed in a given CRISPR-Cas system may reflect selection during spacer acquisition, referred to as spacer acquisition motifs (SAMs), and/or requirements during target interference, known as target interference motifs (TIMs) (Leenay et al. 2017; Shah et al. 2013; Vink et al. 2021).

Other RNA-targeting systems (type III and VI) generally do not require PAMs for target recognition. Instead, they depend on protospacer flanking sequences adjacent to the protospacer in the target RNA. Cleavage is triggered only when mismatches occur between these flanking regions and the repeat-derived sequence, thereby enabling discrimination between self and non-self RNA and reducing the risk of autoimmunity (Deng et al. 2013; Elmore et al. 2016; Leenay et al. 2017; Marraffini et al. 2010; Meeske et al. 2018). However, type III-B systems have been reported to be PAM dependent (Elmore et al. 2016), potentially due to co-option of crRNA and/or Cas machinery of other PAM-dependent CRISPR systems (Silas et al. 2017; Smith et al. 2025).

CRISPR-Cas systems employ two primary strategies for acquiring new spacers: *naive* acquisition and *primed* (or *interference-driven*) acquisition. While both strategies help prevent (ii) self-targeting, primed acquisition also directly counters (iii) the problem of phage escape.

In *naive* acquisition, Cas1-Cas2 target free double-stranded DNA ends which can arise during phage entry into the cell (Modell et al. 2017) or at the origin of replication of plasmids and phages (Levy et al. 2015). Naturally, since the chromosome is circular and replicates infrequently compared to multicopy phages and plasmids, this leads to preferential acquisition of non-self spacers (Levy et al. 2015; McGinn et al. 2019; Nussenzweig et al. 2020). In addition, CRISPR systems have been shown to rely on the DNA repair machinery of the cell to generate suitable spacer substrates. RecBCD (Levy et al. 2015) or AddAB (Aviram et al. 2022; Modell et al. 2017) which generate DNA fragments during the repair of double-stranded DNA breaks, can serve as a source of protospacers. RecBCD is inhibited by crossover hotspot instigator (Chi) sites which are underrepresented in foreign DNA compared to genomic DNA and result in a bias towards non-self spacers (Levy et al. 2015; Modell et al. 2017).

In *primed* and *interference-driven* acquisition, pre-existing spacers with full or partial matches to a previously encountered MGE stimulate the uptake of additional spacers (Hille et al. 2018; McGinn et al. 2019; Nussenzweig et al. 2020). Experimental and bioinformatic evidence for priming is widespread and has been demonstrated across many CRISPR systems, spanning diverse types, species, and biological contexts (Fineran et al. 2014; Garrett et al. 2020; Hoikkala et al. 2021; Li et al. 2014; Margolis et al. 2025; Nicholson et al. 2019; Nussenzweig et al. 2020; Rao et al. 2017; Richter et al. 2014; Semenova et al. 2011; Semenova et al. 2016; Smith et al. 2025; Staals et al. 2016; Wiedenheft et al. 2011).

However, the specifics of the molecular mechanisms underlying primed acquisition are not completely understood. In type I systems, target recognition by an interference cascade appears to recruit Cas3, which may either degrade the DNA to generate spacer substrates for Cas1-Cas2, or translocate with Cas1-Cas2 along the DNA to locate new protospacers (Hille et al. 2018; McGinn et al. 2019; Nussenzweig et al. 2020; Redding et al. 2015; Semenova et al. 2016). In type II systems, Cas9-mediated cleavage may create DNA ends that are subsequently recognized by Cas1-Cas2(-Csn2) for spacer acquisition (Nussenzweig et al. 2020; Nussenzweig et al. 2019).

Notably, only partial spacer-protospacer complementarity is required for priming, and even old, leader-distal spacers can efficiently trigger this process (Deecker et al. 2020). Thus, priming mitigates phage escape (iii) by promoting spacer diversification (Datsenko et al.

2012; Künne et al. 2016; Staals et al. 2016; Swarts et al. 2012).

In any case, both mechanisms for selecting new spacers are inherently biased towards non-self DNA. Indeed, in recent analysis of large CRISPR datasets, 80 to 90% of spacers were mapped to match phage genomes, indicating a significant bias towards phage DNA. However, only 10% of all investigated spacers could be mapped to known origins, which indicates that these results are heavily biased by the limited sequencing depth and classification of the mobilome (Dion et al. 2021; McGinn et al. 2019; Shmakov et al. 2017).

Spacer integration

After acquiring a spacer (and trimming the 3'-ends Nussenzweig et al. (2020)), Cas1-Cas2 complexes integrate the new spacer at the leader-proximal end and thus create the chronological order of spacers (Barrangou et al. 2007; McGinn et al. 2016; Pourcel et al. 2005; Yosef et al. 2012).

The molecular details of spacer integration are complex, and Figure 4 in Nussenzweig et al. (2020) provides a particularly useful schematic overview. The Cas1-Cas2 complex recognizes and attaches to specific motifs in the leader and the leader-proximal repeat (Kim et al. 2019; McGinn et al. 2016; Wei et al. 2015a; Wright et al. 2016; Xiao et al. 2017a). An integration host factor (in type I-E) or Cas1-Cas2 (in type II-A) bind and bend the leader DNA which is hypothesized to improve spacer integration (Nuñez et al. 2016; Nuñez et al. 2015; Wright et al. 2017; Yoganand et al. 2017). Once the Cas1-Cas2 complex binds the leader-repeat region, the DNA is first cleaved at the leader-repeat boundary. The 3'-end of the new spacer is then ligated to the 5'-end of the first repeat. Subsequently, the connection between the first spacer and repeat is cleaved, allowing the 3'-end of the complementary strand of the new spacer to attach to the opposite end of the repeat. Finally, the missing repeat strands flanking the new spacer are repaired using the complementary strand as a template (Goren et al. 2016; Kieper et al. 2019; Kim et al. 2019; McGinn et al. 2019; Nuñez et al. 2016; Nuñez et al. 2015; Nussenzweig et al. 2020; Wright et al. 2016; Wright et al. 2017; Xiao et al. 2017a; Yosef et al. 2012).

Although *ectopic* insertions, spacer insertions occurring within the interior of the array, have been reported for some Cas types, they are rare and generally occur only when the leader is nonfunctional, likely as a fallback mechanism (Achigar et al. 2017; Achigar et al. 2021; Erdmann et al. 2012; Garrett et al. 2015; McGinn et al. 2016; Mosterd et al. 2020; Nussenzweig et al. 2020). The rarity of ectopic insertions is further supported by our findings in Chapter 5.

2.2.2 (II) Guide creation: crRNA maturation

One of the unique properties of CRISPR is the specific targeting of invading DNA or RNA made possible by the guide crRNAs. Here we give a short description of the synthesis of crRNAs from the spacer content in CRISPR arrays.

crRNA production begins with transcription of a long *precursor-crRNA* (pre-crRNA) composed of multiple (to all) repeat-spacer units. Transcription is typically initiated from a promoter located in the leader sequence upstream of the CRISPR array, resulting in higher expression of leader-proximal spacers. Accordingly, the increased interference efficiency of these spacers can be explained by their higher expression levels driven by promoter proximity (Alkhnbashi et al. 2016; Elmore et al. 2013; Liao et al. 2022; Martynov et al. 2017; McGinn et al. 2016; Wei et al. 2015a; Weissman et al. 2018b).

Richter et al. (2014) showed that spacers can contain promoters themselves, leading to

increased expression of the spacer itself and downstream spacers. In addition, type II-C systems have been reported to encode promoters within the repeats (in 5'–3' direction), while the leader (and acquisition site) is located at the 3'-end (Dugar et al. 2013; Dugar et al. 2018; Hoikkala et al. 2021; Hooton et al. 2015; Laanto et al. 2017; Zhang et al. 2013). Consequently, crRNA maturation in type II-C systems partially differs from other Cas types (Feussner et al. 2025; Zhang et al. 2013).

The processing of the pre-crRNA into mature crRNA is highly class- and type-specific. In class 1 systems, Cas6 or Cas5d typically serve as processing enzymes (Carte et al. 2010; Carte et al. 2008; Deng et al. 2013; Garside et al. 2012; Hatoum-Aslan et al. 2011; Nam et al. 2012; Plagens et al. 2014; Prakash et al. 2022; Richter et al. 2012; Sefcikova et al. 2017; Shao et al. 2013; Shao et al. 2016), whereas in class 2 systems pre-crRNA processing is often associated with the interference machinery (Hille et al. 2018; Liao et al. 2022; Nussenzweig et al. 2020).

In most type I and some type III systems, palindromic sequences within the pre-crRNA repeats form hairpin structures that are recognized by Cas6 or Cas5d (Carte et al. 2010; Deng et al. 2013; Hille et al. 2018; Nussenzweig et al. 2020). These secondary structures are essential for processing. In I-A and I-B systems, which lack palindromic repeats, Cas6 remodels the repeats to generate the required secondary structures (Sefcikova et al. 2017; Shao et al. 2013; Shao et al. 2016).

Upon recognition of the hairpin, Cas6 cleaves the pre-crRNA into individual spacer-repeat units (a spacer flanked by partial repeats) (Hille et al. 2018; Nam et al. 2012; Nussenzweig et al. 2020). Cas6 remains bound to the mature crRNA, serving as a scaffold for the assembly of the multiprotein interference complex. However, in type I-A and I-B systems and some type III subtypes, Cas6 releases the crRNA and is not part of the mature interference complex (Charpentier et al. 2015; Nussenzweig et al. 2020).

In type V-A and VI systems, pre-crRNA processing is performed by the interference complexes themselves, which recognize hairpin structures and cleave upstream in a manner functionally similar to Cas5/Cas6 activity (Nussenzweig et al. 2020).

In contrast, type II and V-B instead rely on tracrRNA which binds to the repeat regions of pre-crRNA (Deltcheva et al. 2011; Hille et al. 2018; Shmakov et al. 2015; Shmakov et al. 2017; Zhang et al. 2013). This interaction enables the recruitment of Cas9 and Cas12, respectively. The resulting pre-crRNA-tracrRNA-Cas9 complexes are now further processed by RNases, yielding fully functional effector complexes (Deltcheva et al. 2011; Hille et al. 2018; Jinek et al. 2012; Nussenzweig et al. 2020; Shmakov et al. 2015; Zhang et al. 2013).

2.2.3 (III) Interference

Once a interference complex with a mature guide crRNA has assembled, the complex traverses the cell searching for the presence of foreign invaders (Brouns et al. 2008). Their subsequent targeting is characterized by accurate and sequence-specific cleavage (Gasiunas et al. 2012; Jinek et al. 2012).

The molecular processes behind interference are highly Cas type specific, but can be broadly classified depending on whether the effector targets double-stranded DNA, single-stranded DNA or (single-stranded) RNA. For very helpful visual overviews of the interference mechanisms, we direct the reader to figures provided in Nussenzweig et al. (2020) and Hille et al. (2018).

DNA-activated targeting (type I, II, V). Type I, II, and V are known to predominantly target double-stranded DNA, but differ in the mechanics of interference.

In type I systems, the interference relies on the targeting cascade composed of multiple (subtype-specific) Cas genes and a bound crRNA guide, together with Cas3, which is recruited upon target recognition and is responsible for DNA degradation.

The cascade traverses the cell to survey for a (3 bp) PAM sequence which is recognized by Cas8 in the cascade (Sashital et al. 2012). On PAM-recognition the cascade triggers a separation of the DNA strands (Hayes et al. 2016; Sinkunas et al. 2013), the crRNA is matched and "zips" to the complementary DNA, stably locking the cascade onto the DNA (Blosser et al. 2015; Nussenzweig et al. 2020; Rutkauskas et al. 2015; Xiao et al. 2017b). Then, the complex recruits Cas3, which cleaves the non-complementary strand releasing the cascade from the target and begins translocating in 3'-5' direction. Ultimately, the recruitment of Cas3 leads to complete degradation of the double-stranded DNA (Nussenzweig et al. 2020; Redding et al. 2015; Sinkunas et al. 2013). The precise mechanism behind this degradation, however, is not fully understood (Nussenzweig et al. 2020).

In type II and V, interference is facilitated by the structurally similar Cas9 and Cas12 effectors, respectively (Zetsche et al. 2015). Similarly to type I, the effectors search for compatible PAMs, after which the crRNA progressively "zips" to one strand until the cleavage sites are engaged (Nussenzweig et al. 2020; Redding et al. 2015). For Cas9 a blunt break of both strands is initiated (Garneau et al. 2010; Jinek et al. 2012; Sternberg et al. 2015), whereas Cas12 generates a staggered break cleaving the strands in sequence (Swarts et al. 2019).

In contrast to the effectors of type I and II, Cas12 can remain active after target cleavage and subsequently exhibit non-specific cleavage of single-stranded DNA. Moreover, other Cas12 subtypes have been shown to target single-stranded DNA and RNA and to induce collateral cleavage of non-target single- and double-stranded DNA and RNA (Makarova et al. 2025; Nussenzweig et al. 2020).

The downstream processes following DNA cleavage and the fate of the effector complexes remain relatively poorly understood; however, successful targeting ultimately leads to destruction of the target (Nussenzweig et al. 2020).

Due to the mechanistic targeting differences, type I and II systems have large differences in PAMs they recognize and how partial homology of crRNA impacts priming (Hayes et al. 2016; Nussenzweig et al. 2020; Vink et al. 2021).

RNA-activated targeting (III, VI). Type III systems are defined by the signature gene *Cas10*, which forms part of the interference complex, and by a diverse repertoire of accessory genes (Hille et al. 2018; Makarova et al. 2020; Makarova et al. 2025). Type III systems are activated by RNA and primarily target RNA transcripts, although they are able to target both DNA and RNA (Deng et al. 2013; Goldberg et al. 2014; Makarova et al. 2011; Makarova et al. 2020; Samai et al. 2015; Staals et al. 2014).

DNA targeting arises indirectly through transcription-dependent activation. Following base pairing of the crRNA with a nascent RNA transcript, Cas10 is able to non-specifically bind to DNA and activate its non-specific single-stranded DNase activity (Kazlauskiene et al. 2017). This activity persists only while the target RNA is present and ceases upon its degradation (Deng et al. 2013; Kazlauskiene et al. 2017; Samai et al. 2015; Staals et al. 2014). Consequently, interference can occur only when the target DNA is actively transcribed, generating the RNA required for system activation (Goldberg et al. 2014;

Samai et al. 2015; Smith et al. 2025).

Additionally to its active interference activity, type III systems encode a built-in signalling pathway (Makarova et al. 2025; Stella et al. 2024). In response to target recognition, Cas10 induces a secondary immune response that leads to non-specific RNA degradation of both host and foreign transcripts which ultimately leads to growth arrest of the cell (Kazlauskiene et al. 2017; Niewoehner et al. 2017; Rostøl et al. 2019). This systems is believed to be essential, if the initial response by type III is delayed or inhibited by spacer-protospacer mismatches (Jiang et al. 2016; Nussenzweig et al. 2020; Rostøl et al. 2019).

Type VI systems are defined by their large Cas13 interference nuclease and exclusively target RNA (East-Seletsky et al. 2017; Makarova et al. 2020; Shmakov et al. 2015). Upon target recognition, Cas13 facilitates both target RNA and non-specific RNA degradation (East-Seletsky et al. 2017).

Studies by Meeske et al. (2019) and Vialetto et al. (2022) demonstrated that Cas13-mediated non-specific RNA degradation is sufficient to arrest phage reproduction. However, unlike type III, where multiple regulatory mechanisms restrict the secondary non-specific immune response, Cas13 activity is never shut off. As a result, widespread transcript depletion leads to cell dormancy, effectively trapping the phages within their host, thereby protecting the host's sister cells (Meeske et al. 2019; Nussenzweig et al. 2020).

Importantly, type VI systems exhibit a threshold-dependent response: catastrophic transcript degradation and dormancy occur only when target transcript levels exceed a critical threshold, thereby reducing the risk of autoimmunity. Below this threshold, the system instead mediates transcript silencing (Vialetto et al. 2022).

In this regard, both type III and VI systems possess non-specific "fallback" mechanisms that act when other (CRISPR-based) defenses fail (Silas et al. 2017; Smith et al. 2025).

2.2.4 Recently characterized class 1 subtypes: IV and VII

Owing to their relatively recent discovery and low abundance, the functions and molecular mechanisms of type IV and VII systems are not well understood as of yet (Makarova et al. 2025). Type IV systems are rare in our datasets and type VII systems are completely absent, because they were not yet recognized as a distinct type at the time the databases were compiled and annotated.

Type IV systems are the second most abundant type encoded on plasmids, surpassed only by type I systems, and are known to target double-stranded DNA (Faure et al. 2019; Koonin et al. 2024; Makarova et al. 2015; Makarova et al. 2025; Pinilla-Redondo et al. 2020; Pinilla-Redondo et al. 2022). Although two subtypes interfere by inhibiting transcription instead (Makarova et al. 2025).

Their spacer content is biased towards plasmids, suggesting that they are heavily involved in inter-plasmid conflict, although a substantial fraction of spacers still target phages (Pinilla-Redondo et al. 2020; Pinilla-Redondo et al. 2022). Notably, a preference for plasmid-targeting spacers appears to be a general trend among plasmid-encoded systems regardless of type (Pinilla-Redondo et al. 2022).

Type IV systems show a high diversity of Cas locus architectures. Often they are minimalistic systems which lack adaptation genes and miss genes required for interference (Makarova et al. 2015; Makarova et al. 2020; Pinilla-Redondo et al. 2020). Since most type IV systems encode Cas6 and/or Cas5, their crRNA processing is likely similar to class 1 systems (Hille et al. 2018; Pinilla-Redondo et al. 2020). Type IV systems frequently co-occur with some type I subtypes (in the host), with which they frequently share PAMs, have high repeat,

Cas6 and leader similarity. All of which suggests that type IV systems co-opt or cooperate with type I systems for acquisition and interference (and vice versa) (Pinilla-Redondo et al. 2020).

Type VII systems are a recent discovery in archaeal genomes. They are uniquely characterized by their large effector protein Cas14 which is part of one of the largest effector complexes known among class 1 systems (Altae-Tran et al. 2023; Makarova et al. 2025; Yang et al. 2024). They lack adaptation modules and are believed to incorporate new spacers only infrequently (Makarova et al. 2025). They have been shown to target RNA via crRNA guided nuclease activity of Cas14 and analysis of the spacer content of the few available samples suggests that they target transposable elements specifically (Makarova et al. 2025).

2.3 Ecology of CRISPR-Cas systems

CRISPR-Cas systems interact closely with both their hosts and the surrounding microbial community. Through their effectiveness as antiviral defense mechanisms, they impose strong selective pressures on MGE, driving the evolution of diverse escape and anti-CRISPR strategies and thereby shaping highly dynamic host-phage interactions. Furthermore, theoretical, statistical, and experimental studies have revealed that the functions and consequences of CRISPR-Cas systems extend far beyond immunity. In this section, we review the ecological roles of CRISPR-Cas systems and discuss the diverse effects and functions they exert beyond their canonical role in defense.

2.3.1 Co-evolutionary dynamics of CRISPR-Cas and phages

There is an extensive catalog of theoretical, quantitative model- and simulation-based research on immune systems (of all lifeforms), their distribution and their interactions with their hosts and pathogens, highlighting that these interactions can give rise to complex phenomena, significant co-evolution constraining the pathogens (up to extinction) and immune responses (Bonsma-Fisher et al. 2023).

Following this tradition, a substantial body of theoretical and experimental work has investigated the ecological and evolutionary dynamics of bacteria, phages, and CRISPR-Cas systems at the population level.

Sophisticated mathematical, stochastic, and simulation-based models have been developed to study the complex consequences of CRISPR-phage interactions (Bonsma-Fisher et al. 2023; Bradde et al. 2019; Bradde et al. 2017; Chabas et al. 2018; Childs et al. 2014; Childs et al. 2012; He et al. 2010). These studies have examined, among other aspects, the role of stochasticity in host-phage dynamics (Bradde et al. 2019; Chabas et al. 2018), patterns of persistence, coexistence, and extinction (Bonsma-Fisher et al. 2023; Weinberger et al. 2012a), the influence of bacterial and phage diversity, spacer acquisition, and phage mutation rates on spacer diversity and immune effectiveness (Bonsma-Fisher et al. 2023; Bradde et al. 2017; Childs et al. 2014; Childs et al. 2012; Han et al. 2017; Han et al. 2013; He et al. 2010; Weinberger et al. 2012b), as well as the ecological and evolutionary costs associated with CRISPR-Cas immunity (Skanata et al. 2021).

Both experimental and theoretical work has provided several important insights, some of which are explored in greater detail below and in the following sections. Increased spacer diversity generally enhances immune effectiveness and raises the probability of phage extinction (Common et al. 2020; Guillemet et al. 2022; Iranzo et al. 2013; Van Houte et al.

2016b; Weinberger et al. 2012a). However, these advantages can be counteracted by high phage diversity and rapid phage evolution, allowing phages to escape CRISPR-mediated immunity through mutation (Bonsma-Fisher et al. 2023; Guerrero et al. 2021; Levin et al. 2013; Paez-Espino et al. 2015; Pyenson et al. 2020; Watson et al. 2019; Weinberger et al. 2012b). Nevertheless, over longer timescales, CRISPR-Cas systems are capable of maintaining sufficient spacer diversity and even increase bacterial defense capacity through continual spacer acquisition and turnover (Bonsma-Fisher et al. 2023; Laanto et al. 2017; Morley et al. 2017).

Escape from CRISPR-Cas immunity

The precision of CRISPR-Cas opens up two very effective ways for phages to escape CRISPR targeting. One possibility is the mutation of PAM sequences within the phage, i.e. reducing the frequency of the PAM and thus reducing the frequency of protospacer targets for spacer uptake and interference. As we indicate in Chapter 2, fitting PAMs are required for efficient interference and acquisition for most Cas types (Deveau et al. 2008; Semenova et al. 2011; Vink et al. 2021).

Stochastic modeling by Iranzo et al. (2013) indicates that CRISPR systems become more effective as the number of available protospacers per viral genome increases. Accordingly, PAMs appear to embody a trade-off between maximizing spacer diversity and minimizing the risk of autoimmunity. If the phage reduces PAM frequency, or PAMs become more specific (longer), the pool of usable protospacers shrinks, limiting spacer acquisition and ultimately reducing CRISPR effectiveness.

On the other hand, the precision of CRISPR-Cas allows phages to evade them by point mutations or recombination of protospacers corresponding to spacers in the host's CRISPR array. As a result, old spacers in the array are generally not able to retain identity with protospacers and become obsolete (Sun et al. 2016). Of course, this effect is (partially) mitigated by primed acquisition (Section 2.2.1).

Both evasion behaviors have been confirmed for phages targeted by many Cas types and subtypes, e.g. for type I (Cady et al. 2012; Fineran et al. 2014; Guerrero et al. 2021; Hargreaves et al. 2014; Ortega-Sanz et al. 2025; Semenova et al. 2011; Strotskaya et al. 2017; Watson et al. 2019; Westra et al. 2015) and for type II (Deveau et al. 2008; Martel et al. 2014; Paez-Espino et al. 2015; Pyenson et al. 2020; Sapranauskas et al. 2011; Sun et al. 2013).

In contrast, RNA-targeting systems such as type III and VI do not depend on PAM recognition and are therefore more resilient against this form of phage evasion (Deng et al. 2013; Elmore et al. 2016; Gudbergsdottir et al. 2011; Leenay et al. 2017; Manica et al. 2011; Marraffini et al. 2010; Meeske et al. 2018). Moreover, these systems can induce systemic collateral RNA degradation, a response that is inherently less dependent on precise target recognition (Section 2.2.3).

Alternatively, phages can escape CRISPR systems by actively inhibiting and inactivating the CRISPR systems via *anti-CRISPR proteins*, that directly disrupt the effector complexes, e.g. by inhibiting their assembly, blocking DNA binding, or preventing cleavage (Bhoobalan-Chitty et al. 2019; Bondy-Denomy et al. 2013; Dong et al. 2019; Hwang et al. 2019; Knott et al. 2019; Trasanidou et al. 2019). Notably, phages need to cooperate to get full mileage out of anti-CRISPRs since, often, the first phage delivering the anti-CRISPR is not fast enough in blocking the CRISPR system and still encounters active defense. However, subsequent phages invading a cell with inactive CRISPR are now more likely to succeed (Borges et al. 2018; Landsberger et al. 2018). Some phages are able to reduce Cas binding and

cleaving effectiveness by modifying their own DNA, similar to restriction modification systems (Strotskaya et al. 2017; Vlot et al. 2018).

Another strategy employed by phages is to encode their own (partial) CRISPR systems to directly target the host, e.g. Seed et al. (2013) found a phage encoded I-F system that targets anti-phage systems (not CRISPR). In another example, Hooton et al. (2015) hypothesize that phages of *Campylobacter jejuni* encode a Cas4 gene to induce the bacterial type II-C system to take up new spacers that target its own genome and thus induce autoimmunity.

There is growing evidence that CRISPR-Cas systems participate in "inter-MGE warfare". Many phages and plasmids encode CRISPR arrays and (partial) CRISPR-Cas systems, that carry spacers targeting competing MGEs (Faure et al. 2019; Hargreaves et al. 2014; Koonin et al. 2024; Pinilla-Redondo et al. 2020; Pinilla-Redondo et al. 2022; Al-Shayeb et al. 2020). By exploiting CRISPR-mediated interference, these elements can reduce competition and potentially manipulate host defense networks to their advantage. Because such systems are frequently missing essential Cas genes, particularly those involved in spacer acquisition, they likely depend on host-encoded Cas proteins for parts or all of their functionality (Faure et al. 2019; Koonin et al. 2024; Al-Shayeb et al. 2020).

Overall, the wide range of phage escape mechanisms and CRISPR countermeasures highlights that CRISPR immunity is not a static process, but continuously shaped by evolutionary dynamics. Through the interplay between CRISPR and anti-CRISPR strategies, PAMs, protospacers, CRISPR arrays, Cas proteins and anti-CRISPR genes undergo continual adaptation and counter-adaptation (Bonsma-Fisher et al. 2023; Shah et al. 2013). This process shapes the long-term dynamics of host-MGE interactions and helps explain the remarkable diversity of CRISPR-Cas systems found across prokaryotes.

2.3.2 Costs and fitness trade-offs of CRISPR systems

Although CRISPR systems can provide highly effective immunity, they are absent from roughly half of all bacteria and archaea (Makarova et al. 2025; Weinberger et al. 2012b). This variation in CRISPR prevalence may partly reflect ecological constraints, as phage-host dynamics models suggest that CRISPR-mediated immunity can be overwhelmed under conditions of high phage diversity or rapid phage evolution (Bonsma-Fisher et al. 2023; Iranzo et al. 2013; Weinberger et al. 2012a).

Alternatively, the absence of CRISPR systems may reflect the substantial fitness costs associated with their maintenance, such that the benefits of immunity do not universally outweigh the associated costs (Skanata et al. 2021; Zaayman et al. 2022). Theoretical models further suggest that these costs can promote host-phage coexistence by maintaining a balance between immune and susceptible bacterial subpopulations under sustained selective pressure (Iranzo et al. 2013; Levin et al. 2013; Skanata et al. 2021; Weissman et al. 2018a).

Autoimmunity likely represents a major source of fitness costs. The acquisition of self-targeting spacers can trigger deleterious interference, with potentially severe fitness consequences (Chabas et al. 2022; Chen et al. 2022; Nobrega et al. 2020; Stern et al. 2010; Vercoe et al. 2013; Weissman et al. 2018b; Wimmer et al. 2020). Such self-targeting spacers are relatively common; for example, in one study, approximately 20% of CRISPR-containing genomes, and up to 40% of type I-B and I-F systems, harbor self-targeting spacers (Nobrega et al. 2020). Strikingly, most of these genomes also contain a prophage, which is frequently the target of the spacers. Despite this, no degradation of the CRISPR systems was observed, suggesting that interference activity is suppressed or that additional mechanisms mitigate

autoimmunity (Nobrega et al. 2020).

In another study, elevated acquisition rates were reported to increase toxicity (Heler et al. 2015), suggesting that array expansion must be balanced between maintaining sufficient immune memory for effective protection and limiting potential risks such as autoimmunity (Bradde et al. 2020; Chen et al. 2022; Weissman et al. 2018b).

The array length itself appears to contribute only minimally to the transcriptional cost of CRISPR immunity. However, CRISPR systems can impose substantial energetic costs through the expression of Cas proteins, as demonstrated for type II-A systems (Skanata et al. 2021; Vale et al. 2015).

Additional costs arise from the intrinsic toxicity of CRISPR activity. In type III and VI systems, for example, collateral RNA degradation can induce dormancy or cell death, even under minor phage pressure (Vialetto et al. 2022). Although these systems may allow selective tolerance of beneficial invaders, if they repress transcription sufficiently (Goldberg et al. 2014; Nussenzweig et al. 2020; Vialetto et al. 2022).

Beyond intrinsic costs and autoimmunity, CRISPR systems impose indirect fitness costs by limiting horizontal gene transfer (HGT), as they target plasmids and other mobile genetic elements that mediate gene exchange (Bikard et al. 2012; Jiang et al. 2013; Liu et al. 2025; Marraffini et al. 2008). This can prevent the emergence of virulence and antibiotic resistance by acquisition of virulence-associated and resistance genes (Bikard et al. 2012; Jiang et al. 2013; Marraffini et al. 2008). Consistent with this, CRISPR-Cas systems and antibiotic resistance genes have been reported to be inversely correlated in *Enterococcus faecalis* (Palmer et al. 2010). Moreover, strong selective pressure favoring HGT can lead to inactivation of CRISPR systems (Jiang et al. 2013). A striking example is provided by orphan (lacking Cas genes) type I-F CRISPR arrays found in some *E. coli* that carry spacers targeting I-F Cas genes, thereby blocking reacquisition of a functional system (Almendros et al. 2016). The evolutionary or functional rationale for this apparent self-limiting configuration remains unclear, particularly given that the same strains harbor a functional type I-E system.

Despite this general inhibitory effect on HGT, its magnitude is highly variable (Liu et al. 2025) and may be confounded by the strong linkage between CRISPR systems and mobile genetic elements. In some contexts, CRISPR can even promote gene transfer via transduction (Watson et al. 2018). Overall, the impact of CRISPR systems on HGT over long evolutionary timescales remains unclear, with some studies reporting little to no effect (Gophna et al. 2015), while others suggest a measurable reduction in transfer rates (Liu et al. 2025).

2.3.3 Much more than defense

Shortly after their discovery, multiple observations suggested that CRISPR-Cas systems exhibit functions and behaviors extending beyond immunity and their later application in genome editing (reviewed in Hille et al. (2018) and Westra et al. (2014)).

For instance, one of the first characterized systems, type I-E in the laboratory strain *E. coli* K-12, is inactive under standard laboratory conditions due to transcriptional repression of its Cas genes (Mojica et al. 2010; Pougach et al. 2010; Pul et al. 2010; Westra et al. 2010). In addition, comparative analyses have shown that CRISPR spacer content can remain remarkably stable over long evolutionary periods, in some cases spanning thousands of years (Touchon et al. 2011; Touchon et al. 2010). Consistent with this, an intestinal microbiome sample from a 42,000-year-old mammoth calf contained *E. coli* carrying a

type I system, with approximately 20% of its spacers matching those found in modern strains (Savitskaya et al. 2017).

Such observations appear at odds with the expected rapid turnover of spacers under strong phage-driven selection. Nevertheless, the persistence of CRISPR-Cas systems over long evolutionary timescales suggests that they fulfill important biological roles beyond short-term adaptive immunity.

Beyond antiviral defense, CRISPR systems have been linked to a additional functions. For example:

(1) Gene regulation, including modulation of bacterial traits involved in host interaction and immune evasion (Li et al. 2016; Louwen et al. 2013; Mandin et al. 2007; Sampson et al. 2013; Toledo-Arana et al. 2009), as well as regulation of growth, and collective behaviors (Heussler et al. 2015; Viswanathan et al. 2007; Zegans et al. 2009).

(2) Genome maintenance and evolution, including contributions to DNA repair pathways (Babu et al. 2011), shaping genome evolution through self-targeting and selection pressures (Vercoe et al. 2013), and proposed roles in crRNA-guided transposition in certain minimal type I systems (Faure et al. 2019; Makarova et al. 2020).

(3) Co-option by MGE, which encode CRISPR-Cas components to target competing MGE or interfere with host defenses, often by exploiting host-encoded machinery (Faure et al. 2019; Hargreaves et al. 2014; Koonin et al. 2024; Pinilla-Redondo et al. 2020; Pinilla-Redondo et al. 2022; Seed et al. 2013; Al-Shayeb et al. 2020).

2.3.4 Cooperation of CRISPR-Cas systems

CRISPR-Cas systems of different types often coexist within the same genome and are frequently associated with multiple CRISPR arrays. Increasing evidence suggests that these systems can act cooperatively rather than independently. Such cooperation can occur across all stages of CRISPR immunity. In particular, type II, III and VI systems frequently co-occur with type I systems and appear to function in a coordinated manner.

Several studies indicate that CRISPR-Cas systems can cooperate across types during adaptation. Some adaptation-deficient type III subtypes show PAM preferences consistent with co-occurring type I systems, suggesting that they can recruit type I adaptation machinery in trans (Vink et al. 2021). Similarly, subtype VI-B systems have been shown to utilize type II-C adaptation components (Hoikkala et al. 2021; Margolis et al. 2025; Vialetto et al. 2022).

Cross-talk also occurs in primed spacer acquisition. Type I systems in particular can trigger spacer uptake in other types (I-B $\rightarrow$ II-C/VI-B (Margolis et al. 2025), II-C $\rightarrow$ VI-B (Hoikkala et al. 2021), and I-E/I-F $\rightarrow$ III-A (Smith et al. 2025)). In these cooperative settings, subtype-specific features such as spacer orientation and strand bias must be maintained to ensure that newly acquired spacers remain functional in both systems (Silas et al. 2017; Vink et al. 2021).

CRISPR arrays appear not only to share the burden of spacer acquisition, but also to contribute jointly to CRISPR expression and interference. For example, co-occurring I-F and III-B systems in Silas et al. (2017), as well as I-A, I-G, and III-B systems in Majumdar et al. (2015), were shown to share crRNAs across loci, effectively pooling CRISPR arrays and spacer repertoires. In these cases, crRNAs derived from one array were found to be incorporated into the interference complexes of another system. In other instances, co-occurring systems share components of the maturation machinery; for example, I-A and III-B systems can utilize a common Cas6 protein for crRNA processing (Deng et al. 2013).

As discussed in Section 2.2.3, type III systems initiate collateral interference only after a delay, which requires active transcription of viral DNA (Goldberg et al. 2014; Samai et al. 2015; Smith et al. 2025). Similarly, type VI systems trigger collateral RNA degradation only once target RNA levels exceed a defined threshold (Vialetto et al. 2022). Since type III and VI systems often co-occur with type I systems, these observations support a layered defense model in which type I systems serve as the primary line of defense, suppressing infections before they progress to stages that activate type III and VI responses. When early interference fails, for example due to spacer or PAM escape or high phage pressure, type III and VI systems can act as RNA-level backup defenses, triggering abortive infection and thereby protecting the bacterial population (Silas et al. 2017; Smith et al. 2025; Vialetto et al. 2022).

Clearly, the combined action of multiple CRISPR systems can provide a level of protection that exceeds that of any individual system alone (Smith et al. 2025). This functional synergy is reflected in the frequent co-occurrence of multiple systems and arrays within single genomes (up to seven arrays spanning three types in Majumdar et al. (2015)). These patterns highlight the complexity of CRISPR network organization and suggest that much remains to be understood about how different systems interact and coordinate (Margolis et al. 2025; Smith et al. 2025). We investigate these co-occurrence patterns in more detail using statistical analysis in Chapter 7.

3 CRISPR array evolution

The spacer content of a CRISPR array reflects both the current defensive repertoire of a system and its host, as well as a record of past exposure to mobile genetic elements (MGEs). This history is inherently time-structured due to the polarized acquisition of new spacers. At the level of individual strains, spacer content therefore provides valuable insight into interactions between a host and its environment.

Extending this perspective to groups of closely related strains within the same environment yields a richer information source, enabling inferences about phylogenetic relationships and broader ecological dynamics.

Numerous studies have explored the use of CRISPR array composition for typing and phylogenetic inference (Beauruelle et al. 2018; Beauruelle et al. 2021; Collins et al. 2023; Dion et al. 2021; Karah et al. 2015; Karimi et al. 2018; Lam et al. 2019a; Lier et al. 2015; Muzyukina et al. 2024; Shariat et al. 2014; Tomida et al. 2017). In parallel, several works have explicitly modeled spacer array evolution (Baumdicker et al. 2018; Chen et al. 2022; Collins et al. 2023; He et al. 2010; Kupczok et al. 2013; Kupczok et al. 2015; Weinberger et al. 2012b), and a range of tools have been developed to visualize and analyze CRISPR spacer content and its evolutionary dynamics (Collins et al. 2023; Dion et al. 2018; Lam et al. 2019b; Nethery et al. 2019; Sun et al. 2022).

Its utility increases in combination with metagenomic analysis of the environment, as the analysis of spacer content and discovering corresponding phage protospacers allows study of current and past phage-host interactions and MGE based selection pressures (Achigar et al. 2017; Camarillo-Guerrero et al. 2021; Dion et al. 2021; Han et al. 2013; Soto-Perez et al. 2019; Vink et al. 2021; Weinberger et al. 2012a). An illustrative (fictitious) example for the potential of CRISPR array analysis of microbial communities is shown in Figure 3.1.

CRISPR array spacer content evolves predominantly through two events: the (polarized) acquisition of new spacers and the deletion of (multiple) spacers (Deecker et al. 2020; Horvath et al. 2008; Lam et al. 2019b; Lillestøl et al. 2006; Lopez-Sanchez et al. 2012; Pourcel et al. 2005; Weinberger et al. 2012b). Mutations also accumulate within CRISPR arrays, leading to gradual changes in spacer content. However, this process is expected to be slower than spacer acquisition and deletion and can be accounted for relatively easily by clustering spacers appropriately (Section 5.2.1).

Although less frequent, more complex events such as duplications and rearrangements can have a significant impact on CRISPR array evolution (Achigar et al. 2017; Deveau et al. 2008; Held et al. 2010; Lillestøl et al. 2006; Pourcel et al. 2005). Due to their relative rarity and structural complexity, they are excluded from most modeling approaches, including

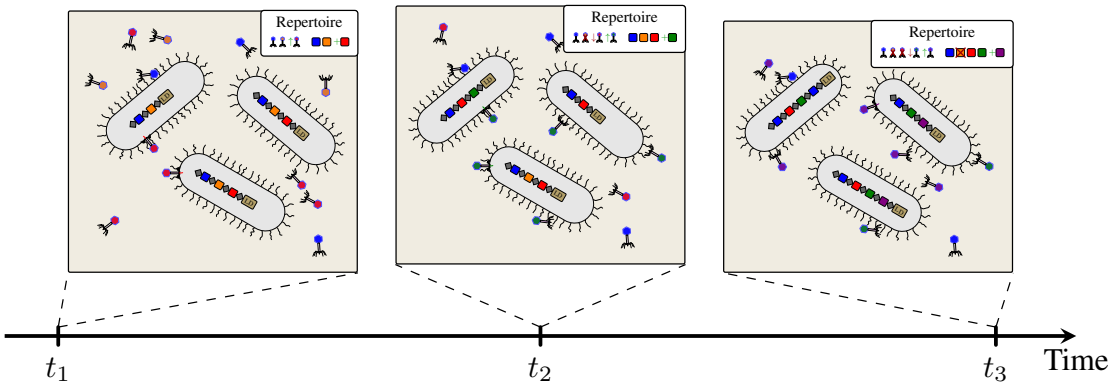


Figure 3.1: Illustration of CRISPR-phage co-evolution in a bacterial community over time. Each panel represents the community at time point t_i. The *repertoire* indicates the set of phages and the spacers present across all bacterial CRISPR arrays; colors identify phage variants and their corresponding spacers. Relative to the previous time point, phage abundance changes are shown by green/red up/down arrows, extinctions by red $\times$ and spacer additions/deletions by green/red $+/\times$. For simplicity, the number of bacteria is kept constant.
Phage invasion leads to spacer acquisition and subsequent decline or extinction of the corresponding phage, while obsolete spacers may be lost over time, generating diversity in spacer content. Some phages persist despite CRISPR pressure and remain endemic. Because spacer acquisition is polarized and delayed, past infections can be reconstructed from spacer order: invasions at earlier time points may still be detectable at t_3, even after phage extinction. However, not all events are retained (e.g., orange, where both phage and spacers are lost). This schematic illustrates general principles; real microbial dynamics are substantially more complex.

our own. In practice, however, many of these events can be detected and represented as combinations of acquisition and deletion events, as discussed in Section 5.3.2.

The biological mechanisms underlying spacer acquisition and deletion differ substantially, giving rise to specific constraints that must be accounted for in modeling. We now outline and examine these principles, which are essential for constructing realistic models of CRISPR array evolution. For a comprehensive review of spacer array evolution, see Garrett (2021).

3.1 Spacer acquisitions

The biological mechanisms underlying spacer selection and acquisition are detailed in Section 2.2.1 and will not be reiterated here. Instead, we briefly recall key properties and their consequences that are relevant for the construction of acquisition models.

1. Spacer acquisition depends fundamentally on Cas genes, particularly Cas1-Cas2. Because Cas1-Cas2 are highly conserved and essential for acquisition, the overall mechanics are generally consistent across CRISPR systems. However, there are significant type-specific differences in spacer selection, processing, and acquisition frequency.

2. Spacers are nearly always inserted at the leader-proximal end of the array. The polarized nature of acquisition leads to notable consequences:

   (i) Spacers located at the front of the array are more recently acquired and target current invasive MGEs. These elements are less likely to have evolved eva-

sion mechanisms and leader-proximal spacers tend to provide more effective defense (Deecker et al. 2020; McGinn et al. 2016; Rao et al. 2016).

(ii) Since the leader sequence encodes a promoter (and repeats and spacers generally do not), leader-proximal spacers generate higher levels of crRNAs (McGinn et al. 2016).

3. Interestingly, despite the generally greater protective value of recently acquired, leader-proximal spacers, comparative studies of spacer arrays have found that the trailer-end is generally well preserved (Lam et al. 2019b; Lopez-Sanchez et al. 2012; Weinberger et al. 2012b). It is believed that the greater protective value of recently acquired spacers is (partially) offset by priming (Section 2.2.1).

4. Newly acquired spacers are almost always unique, even when acquired via priming (Bolotin et al. 2005; Held et al. 2010; Lier et al. 2015; Lillestøl et al. 2006; Lopez-Sanchez et al. 2012; Stout et al. 2018). This uniqueness arises because spacers are short relative to the much larger MGE genomes, and PAM sequences occur frequently, creating a large pool of potential spacers (Iranzo et al. 2013).

5. Arrays acquire new spacers primarily during active invasions (Agari et al. 2010; Fusco et al. 2015; He et al. 2016; Perez-Rodriguez et al. 2011; Quax et al. 2013; Young et al. 2012), and several studies indicate that Cas gene expression is upregulated during these periods (Høyland-Kroghsbo et al. 2017; Høyland-Kroghsbo et al. 2018; Patterson et al. 2017; Patterson et al. 2016). Consequently, spacer acquisition rates are unlikely to be constant over time.

There are, however, exceptions to these general rules.

Rare deviations from the typical leader-proximal insertion pattern, known as *ectopic insertions*, are likely associated with non-functional leaders (Achigar et al. 2021; Erdmann et al. 2012; Garrett et al. 2015; McGinn et al. 2016; Mosterd et al. 2020; Nussenzweig et al. 2020). In addition, in some of the most minimal systems, such as type II-B and II-C, the orientation of spacer acquisition is less clear. These systems are thought to integrate new spacers at the opposite (3') end relative to transcriptional orientation (5' to 3'), making the definition of a canonical "leader end" less straightforward (Dugar et al. 2013; Dugar et al. 2018; Feussner et al. 2025; Hoikkala et al. 2021; Hooton et al. 2015; Laanto et al. 2017; Zhang et al. 2013).

Acquisition rates can only be meaningfully measured and interpreted relative to other evolutionary processes, typically the host mutation rate. Reported estimates vary widely and are difficult to compare directly. Some studies suggest relatively low rates over long timescales (Heler et al. 2015; McGinn et al. 2016), whereas others report rates comparable to or exceeding mutation rates (Paez-Espino et al. 2015; Peach et al. 2025; Somerville et al. 2022a; Somerville et al. 2022b; Zhang et al. 2025). In addition, mutant strains can exhibit substantially elevated acquisition rates (Heler et al. 2015), and strong phage pressure is known to trigger rapid bursts of spacer acquisition.

Acquisition rates are also likely to vary due to differences in Cas machinery, genomic context, and other factors (Weissman et al. 2018b). One such aspect is examined in detail in Chapter 7, where we investigate how spacer acquisition is distributed across multiple arrays within a single genome.

3.2 Spacer (block) deletions

Deletions act as the counterbalance to spacer acquisitions, preventing CRISPR arrays from accumulating an excessive number of spacers that could "dilute" interference effectiveness (Martynov et al. 2017; Rao et al. 2017), increase the risk of autoimmunity (Bradde et al. 2020; Chen et al. 2022; Weissman et al. 2018b) or impose additional maintenance costs (Vale et al. 2015).

They have been documented in numerous studies by comparing spacer content across closely related arrays in experimental or large-scale datasets (Achigar et al. 2017; Deveau et al. 2008; Gudbergsdottir et al. 2011; Held et al. 2010; Lillestøl et al. 2006; Lopez-Sanchez et al. 2012; Pourcel et al. 2005; Stout et al. 2018). When arrays are aligned by the leader-distal end, deletion events affecting the middle or distal region of the array are clearly visible, as new spacers are only added at the leader-proximal end. In contrast, deletions occurring near the acquisition end are more difficult to detect, as they may be misinterpreted as spacer acquisitions (Chapter 5). Another way to observe deletions more directly is through short-term experiments. These typically start with a clonal bacterial population carrying a plasmid with a specific antimicrobial resistance gene and a CRISPR system containing a spacer targeting that plasmid. Under antibiotic pressure, cells are strongly selected to delete the spacer (or otherwise inactivate their CRISPR system) in order to retain the plasmid and survive (Gudbergsdottir et al. 2011; Jiang et al. 2013; Lopez-Sanchez et al. 2012; Rao et al. 2017; Stout et al. 2018).

Both comparative and experimental studies indicate that deletions often occur in contiguous blocks of multiple spacers (Garrett 2021; Gudbergsdottir et al. 2011; Lopez-Sanchez et al. 2012; Stout et al. 2018).

Mechanistically, deletions differ substantially from acquisitions. However, to date, no deletion event has been directly observed, and although several hypotheses exist, the precise biological mechanisms remain unclear. Unlike acquisitions, there is no evidence that Cas proteins are involved in deletions; instead, they likely result from the repetitive structure of CRISPR arrays and the activity of general prokaryotic DNA repair processes.

One early theory, now considered unlikely, proposed that deletions occur simultaneously with duplications or acquisitions (Deveau et al. 2008). Later studies showed that the downstream repeat is typically retained after a deletion, leading some authors to suggest that homologous recombination between repeats could explain the observed patterns (Gudbergsdottir et al. 2011; Rao et al. 2017). Currently, two main homology-based mechanisms are discussed in the literature: (1) *RecA-dependent homologous recombination* and (2) *replication/repeat misalignment*, also called *strand slippage* (Bzymek et al. 2001; Garrett 2021; Gudbergsdottir et al. 2011; Treangen et al. 2009; Yosef et al. 2012).

In mechanism (1), deletions are thought to be actively facilitated by RecA proteins, which bind single-stranded DNA, pair it with a homologous region, and promote strand exchange. However, in at least one study on plasmid-borne CRISPR systems, Ding et al. (2020) found that deletion frequencies were not reduced in strains with nonfunctional RecA. Moreover, RecA is not required for CRISPR adaptation (Garrett 2021), and evidence for recombination between CRISPR arrays remains limited (Kupczok et al. 2015).

Consequently, RecA-independent replication misalignment appears to be the more plausible mechanism. Supporting this, type (2) deletions represent a substantial fraction of recombination events in prokaryotes in general, especially in contexts involving short repeats (< 200 bp), and are not specific to CRISPR arrays (Garrett 2021; Lovett 2004). This mechanism is thought to operate as follows: during replication, direct repeats can misalign,

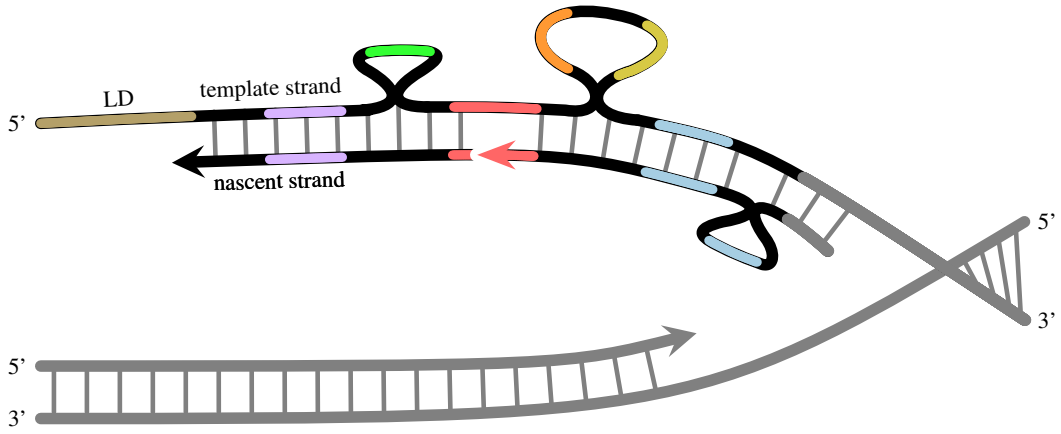


Figure 3.2: Illustration of deletion (and duplication) events caused by misalignment during replication. We depict a CRISPR array consisting of a leader (LD), repeats (black), and spacers (lavender, green, pink, orange, gold, light blue) during DNA replication. The nascent strand (its Okazaki fragments) may misalign with the template strand. Depending on the pairing of repeats and the resulting loop formation, such misalignments can lead to deletions (template strand; green, orange, gold) or duplications (nascent strand; light blue). It is plausible that loops span multiple spacers (orange, gold). Arrows indicate the direction of replication. Adapted from Figure 2 in Garrett (2021).

pairing with another repeat upstream or downstream instead of their correct partner. Subsequent replication then generates duplications or deletions (Lovett 2004). Deletion frequency increases with longer repeats and decreases as the distance between repeats grows (Lovett 2004).

See Figure 3.2 for an illustration of type (2) repeat misalignment-induced deletions and duplications in CRISPR arrays. During replication, the highly similar repeats on the template strand can misalign and pair, forming a loop that contains one or more spacers (and repeats). As the nascent strand is synthesized, this loop is bypassed, resulting in loss of the enclosed spacer(s). Conversely, if loop formation occurs on the nascent strand, spacers can be duplicated (Ding et al. 2020; Garrett 2021).

This mechanism also explains deletions spanning multiple spacers, as repeats could misalign across several repeat-spacer units, producing larger deletion blocks. However, misalignment with more distant repeats is expected to be less frequent. Indeed, results by Lopez-Sanchez et al. (2012) show a monotonic decrease in deletion frequency with increasing length (Section 5.2.3). The maximum deletion length may be constrained by the size of the replication fork, though this limit is likely soft, as forks are typically much longer than several repeat-spacer units (Garrett 2021).

The terminal, leader-distal spacer is often highly conserved, suggesting lower deletion rates (Held et al. 2010; Horvath et al. 2008; Lier et al. 2015; Lillestøl et al. 2006; Lopez-Sanchez et al. 2012; Pourcel et al. 2005; Rao et al. 2017). This pattern is consistent with the replication misalignment model, as the terminal repeat is frequently heavily mutated (Bolotin et al. 2005; Deecker et al. 2020; Horvath et al. 2008; Jansen et al. 2002; Lopez-Sanchez et al. 2012; Mitrofanov et al. 2026; Pourcel et al. 2005), which likely reduces the probability of misalignment (Treangen et al. 2009).

A potentially more important factor is that repeats near the array ends have fewer available misalignment partners (Garrett 2021). However, this effect is expected to be symmetric, since equivalent constraints also apply at the leader-proximal end. Surprisingly, such

symmetry is not commonly discussed in the literature, but in Chapter 5 we find evidence consistent with this prediction.

Under the replication misalignment model, deletion rates are expected to be roughly constant in time, as they result from a naturally occurring replication error. However, deletions are also shaped by selection pressures. For instance, removing younger spacers can reposition older, still effective spacers to the front of the array, enhancing their expression. Supporting this, experiments with type II-A arrays under strong selection pressure showed deletions of several leader-proximal spacers, moving the fifth spacer to the first position and roughly doubling its expression (McGinn et al. 2016). Variation in deletion rates has also been observed between types, with type I mini-arrays showing higher rates than type II arrays on plasmids in *E. coli* (Cañez et al. 2019).

Deletions may also result from other single-strand DNA repair mechanisms in bacteria, although direct evidence is currently lacking. There is, however, evidence that complete array collapse, deletion of all spacers, occurs in nature (Deecker et al. 2020), and that arrays can subsequently be repopulated (Deecker et al. 2020; Wei et al. 2015a; Yosef et al. 2012). Such full collapses appear unlikely for longer arrays under the misalignment model, suggesting that additional mechanisms may contribute.

In summary, we can now outline the key properties that a model must satisfy to be consistent with replication misalignment-driven deletions (Figure 3.2) and that are expected to be reflected in the data:

1. Deletions occur in contiguous blocks, with shorter deletions being more likely than longer ones due to the mechanics of repeat slippage.

2. Deletion rates are reduced at both ends of the array, as the boundary repeats lack partners for misalignment.

3. Deletion rates are roughly constant over time and appear largely independent of Cas type. However, they may vary between species or environments, likely reflecting differences in replication frequency and mutation rates.

4

Mathematical modeling of CRISPR array evolution

Chapter with co-authors
The following section is partly adapted from Fehrenbach et al. (2024) (Chapter 5) and its supplementary material. It incorporates substantial portions of Section 5.1, Supplementary Notes 4 and 5 and revisits selected aspects of Sections 5.2, 5.3.1, and 5.3.2.

We now introduce a series of models that progressively incorporate the evolutionary mechanisms discussed in Chapter 3, providing a set of evolutionary frameworks that allow different modeling assumptions to be compared systematically.

Our goal is to introduce the mathematical framework and models underlying the efficient ancestral reconstruction and analysis tool *SpacerPlacer*, introduced in Chapter 5, for groups of closely related CRISPR arrays that share spacer content. The chapter concludes with a brief overview of alternative modeling approaches proposed in the literature.

Readers may benefit from first familiarizing themselves with Chapter 5, in particular with *SpacerPlacer*, introduced in Section 5.3.2, and referring back to the present chapter whenever additional detail or a more in-depth explanation is desired or required.

CRISPR spacer array evolution challenges traditional phylogenetic reconstruction methods. Reconstructing bacterial evolution from CRISPR arrays faces the problem that the number of spacers in an array is small, and that polarized acquisitions (Section 3.1, 2.) and blockwise deletions (Section 3.2, 1.) generate stochastic dependencies between different evolutionary spacer events. This makes it difficult to use standard phylogenetic methods for DNA sequences, which neglect the specifics of CRISPR evolution, resulting in avoidable errors.

While a variety of visualization tools for CRISPR spacer arrays have been developed (Collins et al. 2023; Dion et al. 2018; Nethery et al. 2019; Sun et al. 2022), reconstruction methods that take into account the specifics of CRISPR array evolution are still rare. An exception is the CRISPR Comparison Toolkit (CCTK) introduced by Collins et al. (2023), which contains a maximum parsimony approach to reconstruct ancestral spacer arrays by first aligning the arrays using the Needleman-Wunsch algorithm, and then inferring ancestral arrays using empirically determined parsimony costs.

However, to go beyond ancestral reconstruction, it is crucial to employ explicit evolutionary models that we introduce here. These models incorporate interpretable parameters,

allowing for a more comprehensive analysis of the corresponding molecular mechanisms and evolutionary processes.

We begin by introducing suitable representations of CRISPR array spacer content, which form the basis for the efficient modeling and analysis methods developed in the remainder of this chapter.

4.1 Representation of spacer arrays

Representing data in a precise and useful manner, removing unnecessary details while retaining relevant information, is essential for developing mathematical evolutionary models and efficiently working with large datasets.

CRISPR arrays inherently contain a large amount of redundant information. Repeat sequences are highly conserved, differing only by a few mutations along the array. Since our focus is on analyzing spacer content, these minor repeat variations are generally uninformative; it is sufficient to distinguish arrays by their consensus repeat and to retain the spacers and their ordering. Accordingly, the individual repeats and leader sequences are discarded prior to analysis (Figure 4.1a).

For the mathematical analysis of spacer insertions and deletions, the specific nucleotide sequences of spacers are not relevant; it is sufficient to distinguish between distinct spacers. Therefore, each unique spacer in the considered group is assigned a unique identifier (ID), represented by a natural number (Figure 4.1a). In this way, each CRISPR array can be treated as a sequence of natural numbers and we obtain a simplified group of CRISPR arrays (Figure 4.1b).

Building on this simplification, we use a representation introduced in Section 5.2.1, in which the spacer arrays are aligned and encoded as a binary presence-absence matrix, termed a *multiple spacer array alignment* (MSAA). In this representation, each column corresponds to a (unique) spacer and each row to an individual array, while entries indicate whether a spacer is present (1) or absent (0) in a given array (Figure 4.1c). For details about the implementation see Supplementary Note 2 in Fehrenbach et al. (2024).

Note, that alignment of the arrays is essential for resolving more complex evolutionary events, such as spacer duplications and rearrangements, which are discussed in detail in Section 5.3.2. However, we disregard such events for now and assume that all spacers are unique throughout the remainder of this chapter.

Finally, *SpacerPlacer* depends on inferring the chronological order of spacer insertions – called the *partial spacer insertion order* (PSIO) – across a group of arrays. The PSIO represents the spacer content of all arrays in the group as a graph, where nodes correspond to spacers and edges encode their inferred order of acquisition (Figure 4.1d).

4.2 Models for spacer insertion and deletion

We do not review the (mathematical) foundations of (tree-indexed, time-homogeneous) Markov processes, standard models of DNA evolution, or phylogenetic inference here, and assume the reader is already familiar with these concepts. For background, we refer to standard texts such as Felsenstein (2004). Our notation and several of the models follow Baumdicker et al. (2018) closely and are based on the same general Markovian framework as the models introduced in Kupczok et al. (2013), although they differ in the specification of insertion and deletion processes and related modeling assumptions, as well as in their

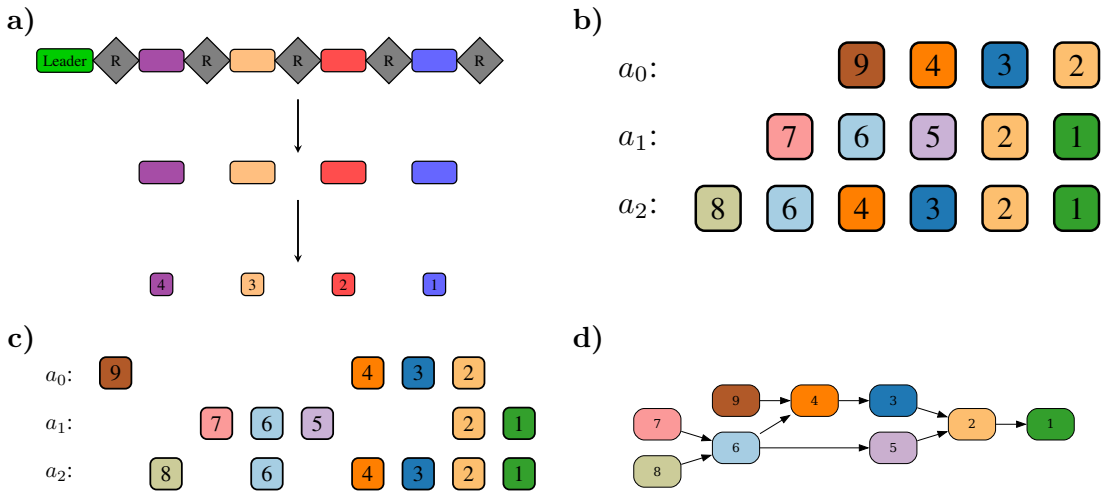


Figure 4.1: Preprocessing and representation of CRISPR spacer arrays. In **a)** we illustrate preprocessing steps in which non-essential components, such as leader and repeat sequences, are removed and each distinct spacer is assigned a unique numerical identifier. In **b)** we show a toy example of three CRISPR arrays (a_1, a_2, a_3) after this simplification. In **c)** we show the corresponding multiple sequence alignment of spacer arrays (MSAA). Finally, in **d)** we show the resulting partial spacer insertion order (PSIO) inferred from the group of arrays.

overall modeling objectives.

All of the models considered here are time-homogeneous Markov processes. While time-homogeneity contrasts with the observation that acquisition rates likely vary over time (Section 3.1, 5.), this simplification is necessary to make computations tractable and the analysis more manageable. Furthermore, our focus is primarily on deletions, which are expected to be approximately time-homogeneous over long timescales, especially when strong selection effects are disregarded (Section 3.2, 3.).

We focus on deletions because, thanks to polarized insertions, they can be reconstructed more accurately than acquisitions and the acquisition rates and deletion rates can be disentangled for parts of the array. As noted by Baumdicker et al. (2018), once the first shared spacer in a group of arrays is reached, the evolution of subsequent spacers is governed solely by deletions and the deletion rate. Or, more applicable to us, given an (assumed) accurate reconstruction of a group $\mathcal{G}$ along a tree $\mathcal{T}$, spacers already present at a branch can only evolve along the corresponding subtree via deletions (Figure 4.2).

We introduce our evolutionary models with two main goals: (1) to efficiently reconstruct the ancestral history of a group of CRISPR arrays while faithfully reflecting empirically observed principles of spacer evolution, and (2) to compute likelihood functions for these models, enabling estimation of rate parameters and the construction of likelihood ratio tests for comparing deletion models.

We first introduce the simpler, less accurate, independent deletion model (IDM) with unordered insertions, and then extend it to the more realistic block deletion model (BDM) with ordered insertions. As we will see, each of these models plays a role in *SpacerPlacer* and the analyses presented in Chapter 5. An illustration of these models is shown in Figure 5.1.

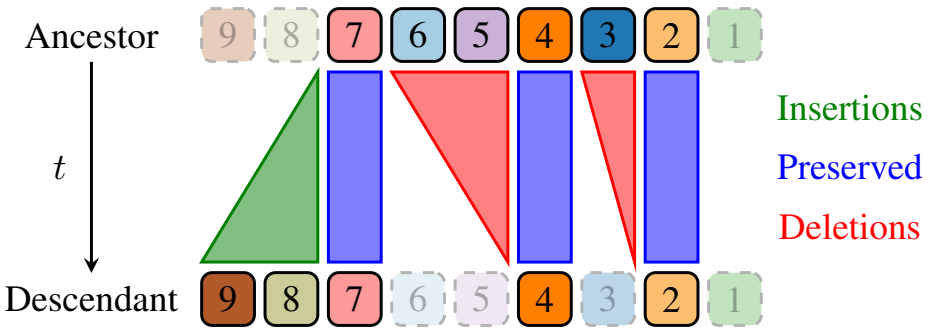


Figure 4.2: Separation of acquisitions and deletions by retained spacers. This figure illustrates an example of CRISPR array evolution along a branch (with length t), represented as MSAA presence-absence patterns. At branch resolution, acquisitions form a block at the leader-proximal end of the array, while deletions appear as contiguous blocks separated by preserved spacers. Thus, owing to polarized spacer acquisition, the evolution of spacers 7 through 2 is governed entirely by the deletion process and depends only on the deletion rate. Spacer 1 represents an already deleted (unobserved) spacer within this branch.

Independent deletion model (IDM) with unordered insertions. Mathematically, it is natural to define a spacer array as a possibly infinitely long sequence $S = (s_1, s_2, \ldots)$, where each $s_i \in [0, 1]$. Then, we can define a minimal model which ignores much of the known functionality of CRISPR arrays, polarized acquisition and block deletions, but is very useful for the implementation of *SpacerPlacer* and understanding the underlying mechanisms. In mathematical terms the model can be defined as follows, using notation introduced by Baumdicker et al. (2018).

Definition 1. Let $\mathcal{S} = (S_t)_{t \in \mathcal{T}}$ be a time-homogeneous Markov jump process with state space $[0, 1]^{\mathbb{N}}$ along a phylogenetic tree $\mathcal{T}$ with root $r \in \mathcal{T}$. The process is governed by the following dynamics: Let $S(t) = (s_1, s_2, \ldots)$ be the state of the array at time $t \in \mathcal{T}$, then S jumps to

(i) $(s_1, \ldots, s_i, U, s_{i+1}, \ldots)$ at rate θ for some independent, uniform $U \in [0, 1]$ and $i = 1, 2, \ldots$,

(ii) $(s_1, \ldots, s_{i-1}, s_{i+1}, s_{i+2} \ldots)$ at rate ρ_{I} for each $i = 1, 2, \ldots$

Note that in this model, newly inserted spacers (U) are by definition almost surely distinct from one another. Furthermore, because the process is time-homogeneous, the waiting times until insertion and deletion events at a given site s_i are exponentially distributed with rates θ and ρ_{I}, respectively. Consequently, the waiting time until the next event of any type is exponentially distributed with rate equal to the sum of the corresponding event rates.

Moreover, insertion and deletion events along the branches of $\mathcal{T}$ follow independent Poisson point processes with rates θ and ρ_{I}, respectively. As a result, the stationary distribution of the CRISPR array length is Poisson with parameter θ/ρ_{I}, implying an expected array length of the same value.

In practice, we use a simplified version of this model in *SpacerPlacer*, restricted to a finite-dimensional state space, i.e. we assume a finite spacer repertoire, where each site corresponds to one particular spacer, and the Markov process acts on a presence/absence profile as produced by an MSAA (of the considered group). We now define this model formally:

Definition 2. Let $\mathcal{S} = (S_t)_{t\in\mathcal{T}}$ be a time-homogeneous Markov jump process with state space $\{0,1\}^n$, $n \in \mathbb{N}$, along a phylogenetic tree $\mathcal{T}$ with root $r \in \mathcal{T}$. Then, let $S(t) = (s_1(t), \ldots, s_n(t))$, where $s_i \in \{0,1\}$ for all $i \in \mathbb{N}$ and $t \in \mathbb{R}$, be the state of the array at time $t \in \mathcal{T}$. Each s_i follows a Markov process with identical transition rates $\theta, \rho_{\mathrm{I}} \in \mathbb{R}$ and the processes $\{s_i\}_{i=1}^n$ are assumed to be independent. The dynamics are governed by two possible transition events:

(i) Insertion ($0 \to 1$): If s_i is in state 0, it transitions to state 1 at rate θ.

(ii) Deletion ($1 \to 0$): If s_i is in state 1, it transitions to state 0 at rate ρ_{I}.

By definition this model is a *General Time Reversible* (GTR) or *substitution* model and s_i and s_j are independent for each $i \neq j$. These substitution models (with varying alphabets) are frequently used for the modeling of mutations or the presence and absence of genes.

As the name implies, the model is *time-reversible*. Consequently, probabilities of array states can be computed efficiently both forward and backward in time along the tree $\mathcal{T}$. This property enables the use of the efficient dynamic programming algorithm of Pupko et al. (2000) for maximum likelihood-based ancestral reconstruction.

Independent deletion model (IDM) with ordered insertions. We now extend the model to incorporate polarized spacer acquisition. This model was first introduced by Baumdicker et al. (2018). Formally, it is obtained from Definition 1 by replacing the insertion transition process (i) with the following.

Let $S(t) = (s_1, s_2, \ldots)$ be the state of the array at time $t \in \mathcal{T}$, then S jumps to

(i) $(U, s_1, s_2, \ldots)$ at rate θ for some independent, uniform $U \in [0,1]$.

Note that, in practice, when applying this model to ancestral reconstruction on an MSAA (Definition 2), insertion events can only occur at sites corresponding to spacers younger than the most recently acquired spacer. As a consequence, the processes s_k are no longer independent and the model is no longer time reversible. Correct evaluation of the model requires explicit tracking of the youngest spacer that has been acquired. This is particularly problematic for backwards inference and likelihood computation, as a given state $S(t)$ can be reached through many distinct event sequences, depending on the position of the most recent acquisition.

Block deletion model (BDM) with unordered/ordered insertions. The replication misalignment model of deletions suggests that spacers are deleted in blocks (Section 3.2, 1.). There is a natural way to incorporate deletions of multiple spacers by retaining (i) and replacing (ii) in Definition 1. Then the jumps are defined as follows.

Let $S(t) = (s_1, s_2, \ldots)$ be the state of the array at time $t \in \mathcal{T}$. Then S jumps to

(i) an unordered or ordered insertion state as defined before,

(ii) $(s_1, \ldots s_{i-1}, s_{i+L}, s_{i+L+1}, \ldots)$ at rate ρ_{B} for each $i = 1, 2, \ldots$, where $L \sim \mathrm{Geom}(\alpha)$ is geometrically distributed with expected deletion length $\alpha \geq 1$.

We parameterize the geometric distribution by the expected deletion length α, this is equivalent to using the more frequently used success probability p, where $p = 1/\alpha$. Thus, if a deletion was initiated, the probability to delete $k \in \mathbb{N}$ spacers is given by

$$P(L = k) = \left(1 - \frac{1}{\alpha}\right)^{k-1} \frac{1}{\alpha}. \tag{4.1}$$

Note that if $\alpha = 1$, the model reduces to the independent deletion model.

Under this deletion model, the expected length of a CRISPR array is no longer exactly given by θ/ρ_{B}, but can be approximated by $\theta/(\rho_{\mathrm{B}} \cdot \alpha)$ (used in Chapter 5).

As described in Section 3.2, empirical evidence suggests that spacer deletions occur when repeats randomly misalign with other repeats in the array. This mechanism produces a deletion length distribution that favors shorter deletions, with probabilities decreasing monotonically as deletion length increases (Section 3.2, 1.). The geometric distribution provides a simple and natural model for capturing this behavior.

Clearly, this model introduces severe stochastic dependencies between the spacers, since deletion events at any position in the array can influence any other position. The site dependency introduced through block deletions makes this model *not* time-reversible and computations backwards in time challenging.

There are two additional immediate issues that need to be addressed with this model definition. First, we assume that deletions always extend in the same direction (to the right). This assumption is made without loss of generality, as allowing a deletion to extend to either the left or right with probability $1/2$ (equivalently, using extension rates $\rho_{\mathrm{B}}/2$ in each direction) yields the same distribution. Moreover, there is currently no evidence that deletions preferentially extend in one direction over the other.

Second, the model implicitly assumes an infinite array when drawing the deletion length L, since the geometric distribution has unbounded support. In practice, however, the length of a deletion is constrained by the number of spacers remaining in the array.

More precisely, let $|S|$ denote the length of the array immediately before the deletion event, and let i be the index of the spacer at which the deletion is initiated. Then the maximum possible deletion length is $l = |S| - i$. Consequently, L is distributed according to a geometric distribution with expected deletion length $\alpha \geq 1$, right-truncated at l. Then, given a geometrically distributed $X \sim \mathrm{Geom}(\alpha)$, the probability of deleting a block of length k is given by

$$P(L = k) = P(X = k \,|\, X \leq l) = \left(1 - \frac{1}{\alpha}\right)^{k-1} \frac{1}{\alpha} \Bigg/ \left(1 - \left(1 - \frac{1}{\alpha}\right)^{l}\right), \tag{4.2}$$

for $k = 1, \ldots, l$.

These boundary effects have important practical implications. In fact, a truncated distribution is more appropriate in the repeat misalignment model, since repeats near the array boundaries have fewer potential partners for misalignment (Section 3.2, 2.). We examine the implications of this boundary effect in detail using simulations and real data in Section 5.3.3, Figures 5.6 and 5.8.

These boundary effects can be substantial for the relatively short CRISPR arrays typically observed in nature and should therefore be taken into account when modeling array evolution. They do not affect the reconstruction procedures considered in Chapter 5, as the BDM is not used for reconstruction and can be straightforwardly incorporated into forward simulation models.

When deriving the likelihood functions, we adopt the simplifying assumption of infinite array length and neglect boundary effects. Of course, incorporating explicit boundary conditions into the likelihood calculations is possible, but it would substantially increase the complexity of the derivations presented in the remainder of this chapter. We find that our simplifications do not compromise the comparison between deletion models, parameter estimates and bias corrections, which remain sufficiently accurate for the analyses considered in this work.

4.3 Usage of evolutionary models in and around *SpacerPlacer*

We briefly summarize which models are used in which context in *SpacerPlacer* and Chapter 5. For a quick overview and comparison to the model usage of other works see Table 4.1.

For ancestral reconstruction, we always operate on MSAAs. MSAAs provide an efficient data format for computation and are also required for handling spacer duplications, as discussed in detail in Section 5.3.2.

Within *SpacerPlacer*, we employ an IDM with unordered insertions as the underlying model for maximum likelihood reconstruction. The reconstruction is implemented using an algorithm similar to that of Pupko et al. (2000). Clearly, this evolutionary model violates several assumptions required for a realistic model of CRISPR array evolution postulated in Chapter 3: (1) spacer insertions are not polarized, allowing insertions at any position in the array (Section 3.1, 2.); (2) the same spacer can be inserted and deleted multiple times (Section 3.1, 4.); and (3) deletions of multiple spacers in a single event are not permitted (Section 3.2, 1.).

SpacerPlacer addresses the limitations as follows. (1) During a refinement step, it enforces the observed PSIO, which is crucial for accurate reconstructions. (2) The insertion rate θ is assumed to be much smaller than the deletion rate ρ_{I}, making repeated acquisitions of the same spacer unlikely and encouraging reconstructions with single insertion events and multiple deletions per spacer site. (3) is unproblematic when deletion gaps are of reasonable length, as this mainly affects likelihoods rather than the observed gap patterns at branch resolution.

With these adjustments, the reconstruction procedure implemented in *SpacerPlacer* can be viewed as a hybrid between maximum likelihood and maximum parsimony. For a detailed description of the algorithm and validation of its reconstruction accuracy, we refer the reader to Section 5.2.2 and 5.3.

In principle, the three issues could be resolved using the more realistic block deletion model (BDM) with ordered insertions. However, because this model is *not* time-reversible and spacer sites in the MSAA are stochastically dependent, computations quickly become prohibitively complex, making it impractical for larger groups of arrays or datasets. However, the BDM with ordered insertions is used when estimating CRISPR-based phylogenies using distance-based methods such as UPGMA or NJ (Section 5.2.1).

With a fast method for accurate ancestral reconstruction of CRISPR arrays, we can analyze the evolution of large datasets in Chapter 5 efficiently. First, we aim to assess whether the IDM or BDM better explains the observed deletions within individual reconstructions and across the dataset. Second, we estimate parameters of the deletion models by maximum likelihood to quantify the evolutionary dynamics of spacer arrays.

These goals are achieved by deriving and evaluating likelihood functions for the IDM and BDM, estimating the the model parameters by maximum likelihood for each reconstruction, and comparing the resulting model fits using a likelihood ratio test.

Computing the likelihood function of the BDM is challenging because the stochastic dependencies between spacers make the underlying calculations highly complex. In particular, a contiguous block of spacers lost along a branch may result from multiple block deletion events of varying lengths, requiring the combined likelihood of several possible event histories to be considered. Moreover, practical estimation of BDM parameters is further complicated by the need to solve a two-dimensional optimization problem, jointly estimating ρ_{B} and α.

To address this challenge, we provide two approaches for computing the likelihood of each connected component. The first is a simplified approximation (Section 4.4.2) that ignores the exact timing and ordering of possible combinations of block deletion events along a branch. The second is a more precise approach (Section 4.4.3) that explicitly accounts for these temporal dependencies. This formulation gives rise to a system of ordinary differential equations that can be solved analytically, but at substantial computational cost.

In the following, after briefly discussing the IDM likelihood function, we introduce both analytical approaches for computing BDM likelihoods. Surprisingly, the simplified approach proves more practical in practice, yielding more accurate parameter estimates in simulations while also enabling additional bias corrections.

4.4 Derivation of the likelihood functions and the ratio test

To compute parameter estimates and compare our deletion models we compute the likelihood functions of both the IDM and the BDM. As already mentioned, the IDM is a special case of the BDM ($\alpha = 1$), nevertheless, we find it instructive to discuss the IDM separately to illustrate the differences between both models.

We introduce the following notation. Let $\mathcal{T}$ denote the underlying tree, which is composed of a finite collection of branches $b_i, i \in I$ for some set $I \subset \mathbb{N}$. We denote the length of a branch b as t_b. We denote the set of lengths of adjacent deletions (gaps) of branch b as $K_b := \{k_i\}_{i=1}^{m_b}$, where each k_i denotes the length of adjacent deletions that are separated by at least one retained spacer and $m_b \in \mathbb{N}$ is the total number of these sets along branch b. We write $|K_b| := \sum_{i=1}^{m_b} k_i$ for the total number of deleted spacers along branch b. The collection of all deletion sets along $\mathcal{T}$ is denoted by $\mathcal{K} := \{K_b \,|\, b \in \mathcal{T}\}$ Furthermore, we denote for each branch b the number of spacers present at the parent node N_b and the set of all these numbers $\mathcal{N} := \{N_b\}_{b\in\mathcal{T}}$ along $\mathcal{T}$. Often, we denote the probability to delete a (single) spacer along a branch as $P(1 \to 0)$, and the probability of retaining a spacer as $P(1 \to 1)$, referring to the representation of a spacer array as a presence/absence pattern.

4.4.1 Independent deletion model

In the independent deletion model (IDM) the likelihood of deletions is independent from their position in the array, and in particular, their adjacency.

The model is governed by one parameter, the deletion rate ρ_I at each site in the alignment. The likelihood along each branch b is solely determined by ρ_I, the branch length t_b, the number of deletions $|K_b|$ and the number of retained spacers, i.e. spacers present in the parent and not deleted in the child, which is given by $N_b - |K_b|$. Since the waiting times for deletions at each site are independent and exponentially distributed, the probability of retaining a spacer $P(1 \to 1)$ is given by

$$P(1 \to 1) = e^{-\rho t_b}, \tag{4.3}$$

and the probability of deleting a spacer $P(1 \to 0)$ by

$$P(1 \to 0) = 1 - P(1 \to 1) = 1 - e^{-\rho_\mathrm{I} t_b} \tag{4.4}$$

for each branch b.

Since all spacer sites are independent, the likelihood of losing $|K_b|$ spacers and retaining $N_b - |K_b|$ spacers along branch b with deletion rate ρ_I is given by

$$L_\mathrm{I}(b, N_b, K_b, \rho_\mathrm{I}) = e^{-\rho_\mathrm{I} t_b (N_b - |K_b|)} \left(1 - e^{-\rho_\mathrm{I} t_b}\right)^{|K_b|}. \tag{4.5}$$

And finally, since the events on different branches are independent, the likelihood of the collection of all deletions $\mathcal{K}$ occurring along $\mathcal{T}$ is

$$\mathcal{L}_{\mathrm{I}}(\mathcal{T}, \mathcal{N}, \mathcal{K}, \rho_{\mathrm{I}}) := \prod_{b \in \mathcal{T}} L_{\mathrm{I}}(b, N_b, K_b, \rho_{\mathrm{I}}). \tag{4.6}$$

4.4.2 Simplified likelihood function of the block deletion model

The computation of the likelihood function of the block deletion model (BDM) is non-trivial. For our use in *SpacerPlacer* (Chapter 5) we rely on the simplified version of the likelihood function of the BDM, due to its stronger performance in parameter estimations of simulations (compare Supplementary Fig. S3 and Supplementary Fig. S9 in Fehrenbach et al. (2024)) and relative ease of computation. For more information about the simplification and differences to the more precise computation see Section 4.4.3 below.

The BDM is governed by two parameters, the deletion initiation rate ρ_{B} and a parameter α determining the average deletion length. Each deletion event of any length still has a probability of $P(1 \to 0) = 1 - e^{-\rho_{\mathrm{B}} t_b}$ to occur. We assume the length of deletions to be geometrically distributed, i.e. $g_\alpha(k) := \mathrm{Geom}(\alpha^{-1}) = (1 - \alpha^{-1})^{k-1}\alpha^{-1}$ is the probability of a block deletion of length k.

We observe that deletions separated by a retained spacer are independent, see Figure 4.2. Thus we can consider each set of adjacent deletions separately. For each adjacent deletion of some length k, we have to consider all combinations of events that sum up to a total deletion length k. As an example consider a deletion of length 3 under the block deletion model. It can either be three independently occurring single deletion events of length 1, one deletion event of length 3, one of length 2 and one of length 1, or one of length 1 and one of length 2.

By extending the argument to the general case of deleting a joint set of length $k \geq 1$, we derive a recursive expression for the corresponding probability. The probability of deleting a joint set of length k in blocks is given by the recursive function

$$f(k, \rho_{\mathrm{B}}, \alpha) = (1 - e^{-\rho_{\mathrm{B}} t_b}) \left(\sum_{j=1}^{k} f(k - j, \rho_{\mathrm{B}}, \alpha) g_\alpha(j) \right), \tag{4.7}$$

$$f(0, \rho_{\mathrm{B}}, \alpha) = 1. \tag{4.8}$$

The intuition behind f is that we divide each deletion k into all possible subsets of deletions of lengths $1, \dots k$, adding up to k in total, and then consider these subsets as independent deletions of respective lengths k_i. Then, these subsets are divided into possible subsets of length $1, \dots k_i$, and so on, until no subdivision is possible (i.e. only a length 1 block remains). Of course, each single deletion event needs to be initiated, which produces the term $P(1 \to 0) = (1 - e^{-\rho_{\mathrm{B}} t_b})$.

For the likelihood of a branch b with adjacent deletions lengths $K_b := \{k_i\}_{i=1}^{m_b}$ we need to include the probabilities for retained spacers, as in the IDM case. Thus we have for the likelihood of branch b

$$L_{\mathrm{B}}(b, N_b, K_b, \rho_{\mathrm{B}}, \alpha) = e^{-\rho_{\mathrm{B}} t_b (N_b - |K_b|)} \prod_{i=1}^{m_b} f(k_i, \rho_{\mathrm{B}}, \alpha). \tag{4.9}$$

Again, since the events on different branches are independent, the likelihood of the tree

with deletion configuration $(\mathcal{T}, \mathcal{N}, \mathcal{K})$ is given by

$$\mathcal{L}_{\mathrm{B}}(\mathcal{T}, \mathcal{N}, \mathcal{K}, \rho_{\mathrm{B}}, \alpha) = \prod_{b \in \mathcal{T}} L_{\mathrm{B}}(b, N_b, K_b, \rho_{\mathrm{B}}, \alpha). \tag{4.10}$$

4.4.3 Precise likelihood of the block deletion model

As the name implies, the simplified block deletion model makes some strong assumptions to ease computations. While treating deletions separated by surviving spacers and branches as independent is correct, the function f, i.e. the probability of losing adjacent losses of length k, is much more difficult to compute in general.

In the *simplified* block deletion model we consider the events that compose a deletion as independent, each deletion event happens with rate ρ_{B} and contains at least one spacer and we only consider the sequencing of events. As an example see Figure 4.3a and b, there we only consider that the deletion of length 5 can be composed of events of different length and can be differently sequenced, e.g. $\{3, 2\}$ shown in a, $\{2, 3\}$ shown in b, $\{1, 1, 3\}, \{1, 3, 1\}, \ldots$.

We overlook two important facts that are illustrated in Figure 4.3c and d. Firstly, we already integrated over the *event times*. For accurate computation of the likelihood we have to keep in mind that it is a continuous process with random jumps where the sequencing of events *and* their event times are significant.

Secondly, we dismiss that *no other* deletion events are allowed to happen on the retained spacers until the deletion event occurs. Then, if a deletion occurs at time b_e, only the remaining spacers need to be considered for the likelihood computations along the remaining branch length $b - b_e$, since already deleted spacers do not contribute to the likelihood anymore.

For example, as illustrated in Figure 4.3c, if a deletion of length 3 occurs first at time t_1, then four spacers must remain unaffected by any deletion event until t_1 and, after t_1, one spacer must remain unaffected until the second event occurs at time t_2. If events are reversed, as illustrated in Figure 4.3d, then the likelihoods till the first event t_1 remains the same, *but* after the deletion 2 spacers need to be retained till t_2 hits. Thus, there is much more complexity in computing possible configurations and their likelihoods compared to the simplified approach.

Fortunately, very similar problems have been investigated previously (Karlin et al. 1975). In our case, we are confronted with a special case of a continuous time Markov chain known as *pure death process*.

A *pure death process* is a continuous time, discrete state, Markov process. More precisely, we have a family of random variables $\{X(t) \,|\, 0 \leq t < \infty\}$ dependent on time t where the possible values of $X(t)$ are positive integers, e.g. a population of individuals, in our case the length of a deletion block. Since we consider a Markov process, we have a transition probability function

$$P_{ij}(t) := P(X(t+u) = j | X(u) = i), \quad i, j = 0, 1, 2, \ldots$$

that is independent of u. This function describes the probability of getting from i individuals to j individuals in time t starting from time u. Since we consider a pure death process, we assume that $P_{ij} = 0$ for all $j > i$, i.e. no individuals are born and individuals can only die. By making some careful considerations, one can come up with a ordinary differential equation describing P_{ij} (Karlin et al. 1975).

In general, the pure death process is considered with the restriction that in each infinitesimal time-step only one individual can die, i.e. $P_{ij} \neq 0$ for i, j satisfying $|i - j| = 1$, and $P_{ij} = 0$

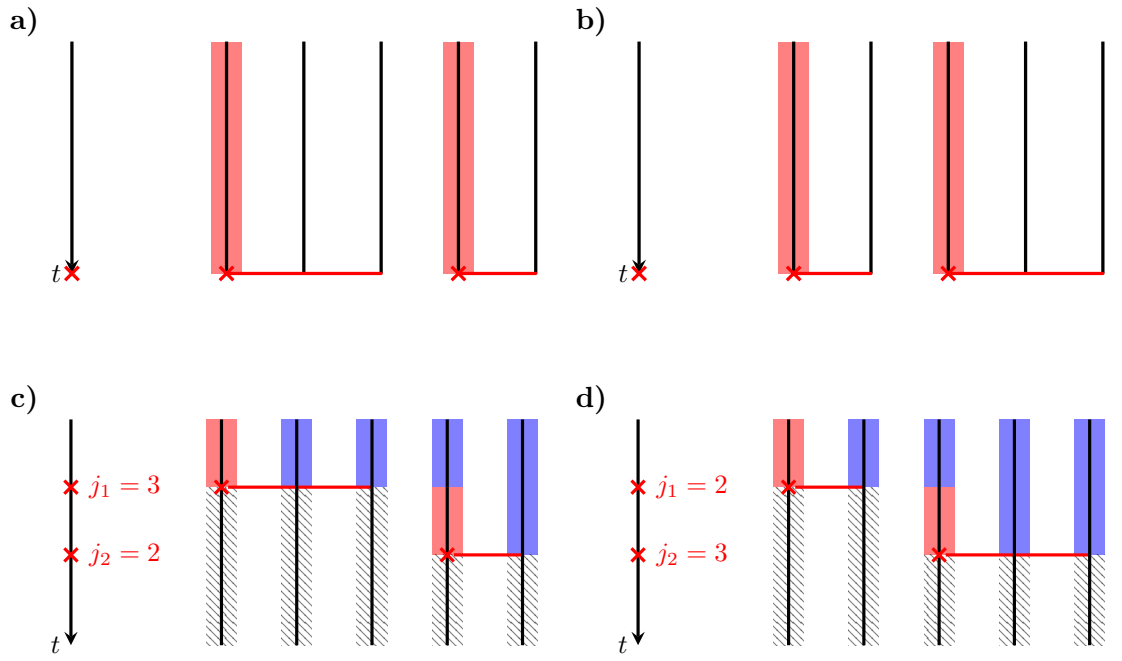


Figure 4.3: Key differences between the simplified and precise likelihood computations for block deletions. We consider a single block deletion of length 5 where each horizontal line represents one spacer position that is deleted in time t. Red shading indicates time intervals during which a deletion event may occur. Crosses mark deletion initiation events, and red lines indicate the extent of the resulting deletion. We show only two realizations of a deletion of length five in a), b) and c), d) respectively. To obtain the complete likelihood all possible configurations that result in a deletion of total length five need to be considered.
a) and **b)** illustrate the simplified likelihood computation. Only the two highlighted spacer positions are considered, and the deletion events affecting them are treated as independent. In this case, only the total time t, the number, order and length of deletions is relevant for the computation.
c) and **d)** illustrate the precise likelihood computation. In addition to deletion events (red), spacer positions that must remain preserved contribute to the likelihood (blue). Hatched regions indicate spacers that have already been deleted and therefore no longer contribute. Since event times j_1 and j_2 are exponentially distributed, both the exact time and ordering of deletion events affects the likelihood because it changes the number and duration of positions that must be preserved (blue area is different between c and d).

for i, j with $|i-j| > 1$. In our case, we consider a spacer block of length k and want to know the probability that at time t no spacer remains, i.e. the transitional probability $P_{k0}(t)$. But since we allow the deletion of multiple spacers described by a geometric distribution, we need to consider a more involved differential equation.

By adapting the considerations in Karlin et al. (1975), we construct a sequence of ordinary differential equations characterized by the following

$$P'_{i0}(t) = \rho_{\mathrm{B}} \sum_{j=0}^{i-1} g_\alpha(i-j)(j+1)\left(P_{j0}(t) - P_{i0}(t)\right), \tag{4.11}$$

$$P_{00}(t) = 1, \tag{4.12}$$

where P'_{ij} is the derivative of P_{ij} with respect to the time t, g_α is the geometric distribution and $\rho_{\mathrm{B}}, \alpha$ are the block deletion model parameters. We are only interested in the probability to delete all spacers $P_{k0}(t)$ in time t and thus do not state the equation in full generality for P_{ij}.

Let us examine this equation in more detail. We delete spacer blocks with constant rate ρ_{B}, in that sense we remain close to the (standard) pure death process, but the length of our losses is geometrically distributed leading to a more involved equation, since it is possible to jump from any value between k and 1 to 0. This implies, as can be seen in (4.11), that P'_{k0} depends on all P_{i0} where $i < k$. This means, to compute P_{k0} for some k requires us to solve all previous ordinary differential equations for $i < k$. Clearly, the initial condition $P_{00} = 1$ arises naturally for our model, as an empty set of spacers can not lose spacers.

Solving these equations analytically (and numerically) is possible, but, through the sequential dependency, gets more difficult for increasing k quickly. For Chapter 5 (Fehrenbach et al. 2024), we computed the likelihood function analytically up to a maximum joint deletion length of $k_{\max} = 68$. Since CRISPR arrays are generally short in length this suffices for most of the groups in our datasets.

In case of a larger set of adjacent deletions of size $k > k_{\max}$, we split the deletion into z fragments with length $k_{\max}$ and a residual of length k_{res} such that $k = z \cdot k_{\max} + k_{\mathrm{res}}$. We regard these $(z+1)$ fragments as independent and compute their likelihood accordingly.

Remarkably, as can be seen in Supplementary Fig. S3 compared to Supplementary Fig. S9 in Fehrenbach et al. (2024) the parameter estimates computed with the simplified function (and no bias corrections) yield better estimates then the ones computed with the precise likelihood function. We believe this is most likely due to inherent biases that are difficult or impossible to correct within the precise likelihood function, as well as limitations arising from finite precision and roundoff errors in the computation of the likelihoods. Moreover, the precise computation is more computationally expensive than the simplified version.

We provide the option to use this likelihood function in *SpacerPlacer*, but recommend using the simplified likelihood function (with bias corrections) for most applications. This is the approach adopted throughout Chapter 5.

4.4.4 Correcting biases of the likelihood functions

When we analyzed simulations in Chapter 5, we found that deletion rates and average deletion length are consistently underestimated in all deletion models. We identified two particularly significant biases:

(1) When we consider the MSAA of the group, we are naturally missing *unobserved* spacers, which were present at some point of the ancestral history in (some) of the arrays, but

deleted over time and thus are *extinct* and no longer observable in the considered samples. Accordingly, deletion events pertaining these spacers do not show up in the data and significantly decrease the estimated deletion rate ρ_{I} in the IDM. In the BDM unobserved spacers additionally influence the neighboring sites and lead to underestimation of both the deletion initiation rate ρ_{B} and the average deletion length α.

(2) The maximum likelihood estimator for the geometric model is known to be substantially biased toward lower values of α when sample sizes are small, i.e., in the regime of few observed deletions.

The significant effects of these biases can be clearly seen in simulation studies; see Supplementary Figure S8 in Fehrenbach et al. (2024).

Correcting the bias for unobserved extinct spacers. We have discussed that the estimates are biased due to unobserved spacers and therefore require correction. We use the IDM case to illustrate this correction. At the end of the section, we briefly describe how the bias correction is handled for the BDM. These considerations closely follow the approach of Felsenstein (1992), who developed likelihood calculations for restriction sites while accounting for unobserved sites.

To compute the corrected likelihood, we normalize the likelihood of observing the events characterized by $N_b, K_b, \rho_{\mathrm{I}}$ at branch b by the probability that all spacers acquired at branch b are observable, i.e., that they are retained throughout the subtree at b. This quantity is obtained via the complementary probability that a spacer is extinct and thus unobserved.

Accordingly, we first recursively compute the likelihood $\xi_{\mathrm{I}}^{\mathrm{unobs}}(b, \rho_{\mathrm{I}})$ that a spacer present before branch b is extinct/unobserved in the corresponding subtree.

Let C_b be the children of b, then

$$\xi_{\mathrm{I}}^{\mathrm{unobs}}(b, \rho_{\mathrm{I}}) = P(1 \to 0, b, \rho_{\mathrm{I}}) + P(1 \to 1, b, \rho_{\mathrm{I}}) \prod_{b_c \in C_b} \xi_{\mathrm{I}}^{\mathrm{unobs}}(b_c, \rho_{\mathrm{I}}), \tag{4.13}$$

where at terminal branches b_{ter} we have

$$\xi_{\mathrm{I}}^{\mathrm{unobs}}(b_{\mathrm{ter}}, \rho_{\mathrm{I}}) = P(1 \to 0, b_{\mathrm{ter}}, \rho_{\mathrm{I}}). \tag{4.14}$$

Since an insertion at branch b needs to be deleted along all child branches (and can not be deleted at the insertion branch), the likelihood $L_{\mathrm{I}}^{\mathrm{unobs}}(b, \rho_{\mathrm{I}})$ for a single spacer inserted at branch b to remain unobserved is given by

$$L_{\mathrm{I}}^{\mathrm{unobs}}(b, \rho_{\mathrm{I}}) := \prod_{b_c \in C_b} \xi_{I}^{\mathrm{unobs}}(b_c, \rho_{\mathrm{I}}). \tag{4.15}$$

Then the corrected likelihood, conditioned on observing the g_b spacers acquired at each branch b in the sample, is given by

$$\frac{L_{\mathrm{I}}(b, N_b, K_b, \rho_{\mathrm{I}})}{\left(1 - L_{\mathrm{I}}^{\mathrm{unobs}}(b_c, \rho_{\mathrm{I}})\right)^{g_b}}, \tag{4.16}$$

since we assume the spacer positions to be independent, and we obtain the corrected likelihood for the whole tree

$$\frac{\mathcal{L}_{\mathrm{I}}(\mathcal{T}, \mathcal{N}, \mathcal{K}, \rho_{\mathrm{I}})}{\prod_{b \in \mathcal{T}} \left(1 - L_{\mathrm{I}}^{\mathrm{unobs}}(b, \rho_{\mathrm{I}})\right)^{g_b}}. \tag{4.17}$$

By optimizing this likelihood, we obtain a bias corrected maximum likelihood estimate of the deletion rate $\hat{\rho}_{\mathrm{I}}^{\mathrm{corr}}$ of the IDM.

Since in the BDM, deletions started at unobserved spacers introduce a dependency that is very hard to quantify (and computationally expensive), it is infeasible to adjust the BDM estimate precisely. Thus, we approximate the bias correction of $\hat{\rho}_{\mathrm{B}}^{\mathrm{corr}}$ by using the bias correction of the IDM. In formulas, we compute $\Delta\hat{\rho}_{\mathrm{I}} = \hat{\rho}_{\mathrm{I}}^{\mathrm{corr}} - \hat{\rho}_{\mathrm{I}}$ to obtain a corrected estimate $\hat{\rho}_{\mathrm{B}}^{\mathrm{corr}} = \hat{\rho}_{\mathrm{B}} + \Delta\hat{\rho}_{\mathrm{I}}$ of the BDM deletion rate.

Correcting the bias of the geometric distribution. Since the maximum likelihood estimator of the geometric distribution is known to be biased for small sample sizes, we apply a standard bias correction defined as follows. Let $n := |\mathcal{K}|$ be the number of deletions along the tree $\mathcal{T}$ and $\hat{p} = 1/\hat{\alpha}$, then we compute the bias correction

$$\hat{p}^{\mathrm{corr}} = -\frac{1}{2}\left(\sqrt{(n+1)^2 - 4n\hat{p}} - n + 2\hat{p} - 1\right), \tag{4.18}$$

and obtain the bias corrected estimator $\hat{\alpha}^{\mathrm{corr}} = 1/(\hat{p} + \hat{p}^{\mathrm{corr}})$.

Combining both bias corrections substantially improves performance in simulations, particularly in terms of the median (compare Figure 5.5 and Supplementary Figure S9 in Fehrenbach et al. (2024)).

However, accurate parameter estimation for individual datasets remains challenging, since the trade-off between ρ_{B} and α is difficult to disentangle, especially at high values. Because CRISPR arrays are relatively short and their evolution can be highly variable, the limited number of events constrains statistical power. Thus, while improved likelihood functions and bias corrections may offer theoretical gains, practical improvements are limited by computational cost and inherently low sample sizes.

4.4.5 Likelihood ratio test

To assess whether deletions occur in blocks, we introduce a likelihood ratio test comparing the IDM and BDM in Chapter 5. Here, we provide a detailed mathematical description of this test.

Note that the IDM and BDM are nested models, since the IDM is a special case of the BDM corresponding to $\alpha = 1$. We now define the corresponding hypotheses

$$H_0 : \quad (\rho, \alpha) \in \Theta_0, \tag{4.19}$$

$$H_1 : \quad (\rho, \alpha) \in \Theta \setminus \Theta_0, \tag{4.20}$$

where $\Theta_0 := [0, \infty) \times \{1\}$ and $\Theta := [0, \infty) \times [1, \infty)$. H_0 corresponds to the IDM and H_1 corresponds to the BDM. We can write down the likelihood ratio test statistic

$$\lambda_{\mathrm{LR}} := \frac{\sup\limits_{\rho_{\mathrm{I}} \in [0,\infty)} \mathcal{L}_{\mathrm{I}}(\mathcal{T}, \mathcal{N}, \mathcal{K}, \rho_{\mathrm{I}})}{\sup\limits_{(\rho_{\mathrm{B}},\alpha)\in\Theta} \mathcal{L}_{\mathrm{B}}(\mathcal{T}, \mathcal{N}, \mathcal{K}, \rho_{\mathrm{B}}, \alpha)} \tag{4.21}$$

$$= \frac{\sup\limits_{\rho_{\mathrm{I}} \in [0,\infty)} \prod_{b\in\mathcal{T}} L_{\mathrm{I}}(b, N_b, K_b, \rho_{\mathrm{I}})}{\sup\limits_{(\rho_{\mathrm{B}},\alpha)\in\Theta} \prod_{b\in\mathcal{T}} L_{\mathrm{B}}(b, N_b, K_b, \rho_{\mathrm{B}}, \alpha)} \tag{4.22}$$

$$= \frac{\sup\limits_{\rho_{\mathrm{I}} \in [0,\infty)} \prod_{b\in\mathcal{T}} L_{\mathrm{B}}(b, N_b, K_b, \rho_{\mathrm{I}}, 1)}{\sup\limits_{(\rho_{\mathrm{B}},\alpha)\in\Theta} \prod_{b\in\mathcal{T}} L_{\mathrm{B}}(b, N_b, K_b, \rho_{\mathrm{B}}, \alpha)}, \tag{4.23}$$

The maximization problem can be solved more efficiently by taking the logarithm of the likelihood ratio. Furthermore, since the models are nested, Wilks' theorem offers an asymptotic distribution of the log-likelihood ratio

$$-2\ln(\lambda_{\mathrm{LR}}) \sim \chi^2(1), \tag{4.24}$$

as sample size approaches infinity, where $\chi^2(n)$ is the Chi-squared distribution for n degrees of freedom (in our case $n = 1$, since α is the only degree of freedom).

Accordingly, we use $\chi^2(1)$ as the significance threshold of the test and reject H_0 in favor of H_1 when $-2\ln(\lambda_{\mathrm{LR}})$ exceeds $\chi^2(1)$. This significance threshold should be treated conservatively since our sample sizes are generally relatively small. In Chapter 5, most of the considered datasets far exceed this threshold and the results are likely reliable.

If there is no deletion at all ($\mathcal{K} = \emptyset$), we have $L_{\mathrm{I}}(b, N_b, 0, \rho_{\mathrm{I}}) = e^{-\rho_{\mathrm{I}} b N_b}$, $f(\rho_{\mathrm{B}}, \alpha, 0) = 1$, $L_{\mathrm{B}}(b, N_b, 0, \rho_{\mathrm{B}}, \alpha) = e^{-\rho_{\mathrm{B}} b N_b}$ for all $b \in \mathcal{T}$ and thus $\lambda_{\mathrm{LR}} = 1$ with maximum point $\hat{\rho}_{\mathrm{I}} = \hat{\rho}_{\mathrm{B}} = 0$ ($\hat{\alpha} \in [1, \infty)$).

Note that we apply the bias corrections only to the parameter estimates and not to the likelihood ratio test.

4.5 Comparison to other models of spacer array evolution

Several stochastic models have been developed specifically for CRISPR spacer array evolution (Baumdicker et al. 2018; He et al. 2010; Kupczok et al. 2013; Kupczok et al. 2015), while others focus on the co-evolutionary dynamics between phages and bacteria (Section 2.3.1). These population-level models typically incorporate simplified representations of CRISPR array evolution. Although such approaches may inform extensions of the models developed here and vice versa, we do not explicitly consider host-phage co-evolution in this work, but instead focus on the evolutionary dynamics of CRISPR arrays.

Additionally, spacer array length has been the focus of several modeling approaches (Kupczok et al. 2013; Martynov et al. 2017; Pavlova et al. 2021).In these models, the balance between spacer acquisition and deletion determines an equilibrium distribution of array lengths, which may itself be subject to selection (Martynov et al. 2017). In our setting, where spacers are assumed to have no fitness effect, the number of spacers per array (approximately) follows a Poisson distribution whose mean depends on the ratio of the overall acquisition rate to the per-spacer deletion rate.

Within the space of models specifically developed for CRISPR spacer array evolution, several approaches have been proposed. While some postulates, such as simultaneous acquisition and deletion (He et al. 2010), have been rejected, our framework using Markov processes is consistent with, and extends, the models presented in Baumdicker et al. (2018) and Kupczok et al. (2013).

Table 4.1 summarizes the different evolutionary models used in *SpacerPlacer* and in two closely related works (Collins et al. 2023; Kupczok et al. 2013). Kupczok et al. (2013) compare the IDM with ordered insertions, the IDM with unordered insertions, and a deletion model in which contiguous fragments of the spacer array are removed. In contrast to the block deletion model with geometrically distributed lengths considered here, their fragment deletion model assigns equal probability to all contiguous segments of a given array, independent of length. Rather than separating acquisition and deletion rates, their analysis focuses on the induced distribution of array lengths, which in the IDM depends on the ratio θ/ρ_{I} and takes a more complex form under the fragment deletion model (dependent on the same ratio). These parameters are estimated from available data at the

Task	*SpacerPlacer*	Kupczok et al. (2013)	Collins et al. (2023)
Phylogeny computation	external or blockwise UPGMA/NJ	rooted NJ	external or most parsimonious
Reconstruction: Guide	unordered acq. individual del.	—	custom acq. costs custom del. costs
Reconstruction: Refined	partially ordered acq. individual del.		
Parameter estimation	ordered acq.* individual del. blockwise del.	ordered acq. individual del. fragment del.	—
Deletion models for LRT	individual vs blockwise	—	—

Table 4.1: Comparison of spacer acquisition and deletion models used across publications. The table is adapted from Supplementary Table S1 of Fehrenbach et al. (2024). The table shows a comparison of the spacer deletion and insertion models used here and in related publications (Collins et al. 2023; Kupczok et al. 2013). Naturally, the most parsimony approach by Collins et al. (2023) does not include rates for different events but custom cost functions for ordered and ectopic acquisitions, duplications, deletions depending on their position and length. *The parameter estimation and the likelihood ratio testing in *SpacerPlacer* rely solely on reconstructed deletions, thus the acquisition rate can only be estimated indirectly.

time, and Kupczok et al. (2013) conclude that both independent deletion and fragment deletion models provide plausible descriptions of the observed dynamics. They extend this work in Kupczok et al. (2015), where they analyze the contribution of recombination events to array evolution. However, their analysis is based on comparisons of spacer order between arrays rather than an explicit evolutionary model.

The block deletion model introduced here has two main advantages compared to the fragment deletion model: (a) observed deletion patterns in CRISPRCasdb and experimental studies (Lopez-Sanchez et al. 2012) are consistent with block deletions of geometrically distributed length, and (b) the memoryless property of the geometric distribution enables efficient likelihood computation for both reconstruction and parameter estimation.

In Baumdicker et al. (2018), the independent deletion model is used to derive an estimator for the deletion rate based solely on variability at the trailer end of spacer arrays. This result, together with the theoretical framework developed therein, motivate our separate consideration of the deletion rate. However, our ancestral reconstruction framework allows us to incorporate additional information from the full reconstructed histories rather than relying only on trailer-end variation.

Collins et al. (2023) propose an alternative to stochastic evolutionary models by introducing a reconstruction tool based on maximum parsimony principles. To accommodate all possible types of events in CRISPR array evolution, they define empirically determined scores for acquisitions at the leader end, duplications, insertions within the array, deletions (of arbitrary length), and independent acquisitions.

In contrast, we rely on explicit evolutionary models that can offer insight beyond ancestral reconstruction through parameters that correspond to molecular mechanisms and evolutionary processes. We use a dual approach where the underlying molecular mechanisms of CRISPR arrays influence the modeling choices, while the reconstruction-based parameter estimates also enable us to investigate the evolutionary (co-)dynamics of CRISPR arrays. For example, our framework allows us to introduce the likelihood ratio test that identifies the mode of deletions.

With a solid understanding of CRISPR-Cas systems and our spacer evolution models, we now turn to the main (published) contributions (Chapter 5, 6 and 7) and introduce our

ancestral reconstruction algorithm *SpacerPlacer*.

5

SpacerPlacer: ancestral reconstruction of CRISPR arrays reveals the evolutionary dynamics of spacer deletions

Chapter with co-authors
This chapter is based on Fehrenbach et al. (2024), with minor modifications, and accordingly contains contributions by the co-authors. Note that references to supplementary notes and figures refer to the supplementary material of Fehrenbach et al. (2024).

Chapter summary (Abstract)

Bacteria employ CRISPR-Cas systems for defense by integrating invader-derived sequences, termed spacers, into the CRISPR array, which constitutes an immunity memory. While spacer deletions occur randomly across the array, newly acquired spacers are predominantly integrated at the leader end. Consequently, spacer arrays can be used to derive the chronology of spacer insertions.

Reconstruction of ancestral spacer acquisitions and deletions could help unravel the coevolution of phages and bacteria, the evolutionary dynamics in microbiomes, or track pathogens. However, standard reconstruction methods produce misleading results by overlooking insertion order and joint deletions of spacers.

Here, we present SpacerPlacer, a maximum likelihood-based ancestral reconstruction approach for CRISPR array evolution. We used SpacerPlacer to reconstruct and investigate ancestral deletion events of 4565 CRISPR arrays, revealing that spacer deletions occur 374 times more frequently than mutations and are regularly deleted jointly, with an average of 2.7 spacers. Surprisingly, we observed a decrease in the spacer deletion frequency towards *both* ends of the reconstructed arrays. While the resulting trailer-end conservation is commonly observed, a reduced deletion frequency is now also detectable towards the variable leader end. Finally, our results point to the hypothesis that frequent loss of recently acquired spacers may provide a selective advantage.

5.1 Introduction

As we detailed in Chapter 2, bacteria and archaea defend against MGE with CRISPR-Cas systems. They provide adaptive immunity through spacer acquisition and interference and exhibit substantial diversity in structure and function.

CRISPR arrays evolve frequently through polarized spacer acquisition and deletion, forming an inheritable, dynamic record of past infections that reflects host-phage co-evolution. Moreover, CRISPR arrays are relatively short in length and can be reliably identified due to their repeat structure (Barrangou et al. 2016; Lam et al. 2019a; Shariat et al. 2014; Tomida et al. 2017) (Section 2.1.3). Thus, CRISPR array dynamics could be particularly useful for investigating evolutionary relationships at high resolution, tracking and typing samples in epidemiological outbreaks, understanding the dynamic ecology of microbiomes, and analyzing biotechnological and pharmaceutical biocultures (Barrangou et al. 2016; Guerrero et al. 2021; Karimi et al. 2018; Lam et al. 2019b; Martínez Arbas et al. 2021; Muzyukina et al. 2024; Sun et al. 2016; Westra et al. 2016).

Furthermore, while the molecular mechanisms underlying spacer acquisition, such as the roles of Cas1, Cas2, and the leader sequence, have been studied extensively (Chapter 2, Section 3.1), the processes governing spacer deletion remain much less well understood (Section 3.2). By comparing observed patterns of spacer deletions to neutral evolutionary models (Section 3.2, 4), it becomes possible to investigate whether selection influences CRISPR array evolution. In particular, López-Beltrán et al. (2024) recently reported that phage abundance varies dependent on the position of the corresponding spacers within the array. This observation raises the possibility that selection pressures may vary with spacer age, which is reflected in their position within the array.

Both understanding the mechanisms and dynamics of spacer deletions and detecting position-dependent selection require accurate reconstruction of ancestral spacer acquisition and deletion events. Developing the corresponding reconstruction framework is therefore a central objective of this chapter.

SpacerPlacer **allows to investigate spacer array evolution**

Here, we introduce *SpacerPlacer* a new parametric likelihood-based approach tailored for the reconstruction of ancestral spacer insertions and deletions. *SpacerPlacer* reconstructs ancestral states while considering the unique features of CRISPR arrays and their polarized insertion order through an evolutionary model. The advantage of using a probabilistic approach is that we can compare different evolutionary models for CRISPR spacer arrays and investigate the corresponding molecular mechanisms of spacer acquisitions and deletions, as well as selection pressures, in more detail. One major benefit of this model-based approach is that it – in addition to the reconstruction – allows to estimate the underlying evolutionary parameters. This enables us to infer the distribution of the number of jointly lost adjacent spacers, detect potential differences between CRISPR-Cas types, and test model assumptions using likelihood ratio tests.

We applied *SpacerPlacer* on all suitable high-confidence CRISPR arrays from the CRISPR-Casdb database (Pourcel et al. 2019). Despite the variability of subtypes, our analysis did not find significant differences in deletion rates and lengths between the subtypes considered here, which points to a Cas-independent process for deletion (Section 3.2). We find that the median deletion rate per spacer is 374 times higher than the mutation rate per site. Since spacers and their repeats are fairly short (Chen et al. 2022; Garrett 2021), we conclude that the evolution of CRISPR arrays by acquisitions and deletions is much

faster than the mutation of the spacer sequences themselves.

We then investigate the reconstructed spacer deletions in more detail. To this end, we perform likelihood ratio tests and find that spacers are indeed regularly jointly deleted. Using our probabilistic framework, we estimate the mean length of these joint deletions as 2.7 adjacent spacers.

We also investigated how the distribution of the deletion frequency changes with respect to the position within the array. We find that deletion frequencies are decreased towards both boundaries of the array. This boundary effect is consistent with the conjecture that spacer deletions are mainly caused by repeat misalignment (Section 3.2).

Moreover, by comparing empirical and simulated deletion frequencies, we find indications that either selection increases the deletion frequency of the first spacer, or mutations in the last repeat reduce the deletion frequency of the last spacer, or both.

5.2 Material & Methods

5.2.1 Spacer array data preparation

To run *SpacerPlacer* we require a group G of completely sequenced related CRISPR arrays with consecutively ordered spacers and a respective phylogenetic tree $\mathcal{T}$. Ideally, this tree should be provided by the user, but *SpacerPlacer* provides a heuristic method to infer a suitable tree $\mathcal{T}$ from the spacer arrays itself.

Each unique spacer DNA sequence in G is labeled with an individual natural number. We omit an illustration here, but simple examples of a group G can be found in Figure 5.2 and Figure 5.3a). Obviously, the arrays in G should share some spacers but not all be identical. So, we restricted the analysis to such groups.

Database

CRISPRCasdb is a freely accessible database that contains the CRISPR array predictions of "CRISPRCasFinder" (Couvin et al. 2018) on whole genome assemblies collected from GenBank. For the construction of the spacer database, we extracted spacer sequences from CRISPRCasdb (Pourcel et al. 2019) in July 2022. We only consider predictions of the highest evidence level (4), resulting in a total of 7352 repeats and 32624 arrays.

Clustering spacers based on sequence homology

After extraction, we clustered the spacers into sets that we treat as identical, and have likely undergone only mutations. Each spacer in the CRISPR arrays was relabeled according to its respective cluster label. The relabeled CRISPR arrays serve as input for *SpacerPlacer*.

The clustering was performed individually for each identified consensus repeat. The procedure encompassed several stages: Initially, all CRISPR arrays containing the specific consensus repeat were selected. Subsequently, spacers associated with these arrays were aggregated. Hierarchical clustering was then applied to these spacers using the following algorithm:

Initially, each spacer (combined with its reverse complement) was treated as an independent cluster. In subsequent iterations, the clusters were progressively merged if they contained spacer pairs that differed by a single Levenshtein editing operation (substitution, insertion, deletion).

In theory this liberal clustering method could result in huge clusters. However, in practice, we find clusters to be small. For all 53965 repeats on CRISPRCasdb, more than 97% had solely identical spacer sequences in the associated clusters. For the remaining 1426 repeats, the largest spacer cluster has size 11, but most of them had size 2.

Identification of sets of closely related CRISPR arrays

We further subdivided the pool of closely related arrays with the highest confidence level according to their repeat sequence. For all repeats, we checked if there was an overlap between reverse complement repeats and reverse complemented spacer arrays.

CRISPR arrays were assigned their Cas type according to the closest Cas-cassette found within the genome. Arrays with no close ($<$ 10000 bp distance) Cas-cassette were removed unless only one Cas-cassette was found within the genome. Groups were further categorized by genus and Cas type. Finally, the groups were further subdivided by pairwise overlap of the spacers. Each spacer array has to have at least one spacer overlap with any of the other arrays in a group to be included. We excluded groups that had less than three arrays or lacked spacer diversity between arrays. Thus, within each group, each sample has the same repeat, genus, and Cas type. For further details and a breakdown of the number of groups and arrays within each step see Supplementary Note 1A.

The final dataset with core gene trees fed into *SpacerPlacer* contains 518 groups with 5934 arrays in total. Of these, 333 groups (4565 arrays) contain at least one reconstructed deletion. We refer to these groups as the "CRISPRCasdb dataset". A breakdown of the Cas types and genera found in the CRISPRCasdb dataset is shown in Supplementary Figure S6 and S7, respectively.

Multiple spacer array alignment

The order of spacers in different arrays, and sometimes even within a single spacer array, can be inconsistent and produce contradictions if we assume polarized insertions. For instance, if a spacer occurs twice in array A and only once in array B, it is unclear which of the duplicated spacers in array A corresponds to the spacer in array B. Another type of inconsistency can arise if spacer 1 follows spacer 2 in array A but spacer 2 follows spacer 1 in array B. This is clearly not realizable through a single acquisition.

To identify and group such candidate spacers, *SpacerPlacer* performs a Multiple Spacer Array Alignment (MSAA) on each group of spacer arrays. We then identify the most plausible matching spacers according to their positions in the MSAA.

We align spacer arrays solely based on the spacer labels. Note that we do *not* align the spacer DNA sequences. For the multiple spacer alignment we use the tool MAFFT (Katoh et al. 2019), since MAFFT can also align arbitrary characters (the spacer labels in our case) instead of nucleotide and protein sequences. *SpacerPlacer* tweaks the MAFFT parameters and uses low gap penalties and a very high mismatch penalty, to prevent spacer mismatches, favoring gaps instead. More details about this process can be found in Supplementary Note 2.

Tree estimation

The quality of the underlying phylogenetic tree seriously affects both the ancestral spacer reconstruction and the parameter estimates. In general, we suggest using whole genome trees,

if available, to reduce the dependency between tree estimation and array reconstruction, especially, if rate estimation is desired.

For each group of spacer arrays, we reconstructed a phylogenetic tree, independent of the arrays, based on the single-copy core genes of the corresponding sample. We used Bakta to annotate genomes (Schwengers et al. 2021), the pangenome analysis tool panX (Ding et al. 2018), and IQ-TREE (Nguyen et al. 2015) to reconstruct trees. Throughout this work, we used these core genome trees unless otherwise stated.

We did not find evidence indicating that our median estimates were affected by potential biases introduced by using core genome trees for CRISPR arrays where horizontal transfer might have occurred. Furthermore, we removed samples with multiple arrays from our analysis to avoid the inability of core genome-based trees to differentiate between them.

As an alternative for cases where no plausible independent tree is available, *SpacerPlacer* offers the option to use basic unweighted pair group method with arithmetic mean (UPGMA) or neighbor joining (NJ) approaches to infer trees directly from CRISPR arrays. For this, we developed a distance function for spacer arrays that takes into account the specifics of CRISPR array evolution, including blockwise deletions and acquisition order. The details of the distance function can be found in Supplementary Note 3.

(Re-)orientation of spacer arrays

We used the algorithm introduced in Chapter 6 (*CRISPR-evOr*) to determine the orientation of spacer arrays.

Accordingly, the orientation of spacer arrays was chosen according to a comparison between the likelihood of the "forward" reconstruction of the group, i.e. the orientation supplied by CRISPRCasdb, and the likelihood of the "reverse" reconstruction, where the MSAA of the group was reversed. If no clear distinction, i.e. a large margin between likelihoods was found, the orientation provided by CRISPRCasdb was used.

5.2.2 Simulations

The simulation of spacer acquisition and deletion events along a simulated tree was done similarly to previous approaches (Baumdicker et al. 2018; Kupczok et al. 2013).

We start at the root of the corresponding phylogeny, with spacer arrays with a random length that is distributed according to the stationary length distribution of the respective evolutionary model. The array is then evolved along the provided tree by sampling insertions and deletion events according to an independent or block deletion model with polarized insertions. The underlying trees were simulated according to the coalescent, if not indicated otherwise. More details can be found in Supplementary Note 1B.

5.2.3 Estimation of block deletion length in experimental data

Naturally, the actual process that determines the number of jointly lost spacers is more complex than a geometrically distributed random variable, but this simplification provides an efficient and precise computation of the likelihood function in *SpacerPlacer*. However, we found statistical evidence that the lengths of experimentally derived spacer deletion blocks are well approximated by a geometric distribution.

To test the approximation of geometrically distributed lengths of block deletions we performed a goodness of fit test (Kolmogorov-Smirnov) on the empirical distribution

inferred by Lopez-Sanchez et al. (2012). They selected against an individual spacer (number 258) and experimentally determined the neighboring spacers that were deleted jointly with this spacer (Figure 3 in Lopez-Sanchez et al. (2012)). Since the time span of the experiment was very short and only a few (2) concurrent spacer deletions occurred, we assume the deletions of and around spacer 258 to be the result of single events. Therefore, we conditioned on a guaranteed deletion of spacer 258 to estimate the expected length of deletions.

5.3 Results

5.3.1 A new probabilistic model for CRISPR array evolution

Underlying probabilistic models

We introduce different models, all Markov processes along a given (estimated) phylogenetic tree $\mathcal{T}$ (note that we adapt the naming conventions of previous studies (Baumdicker et al. 2018; Kupczok et al. 2013)). See Section 4.2 and 4.3 for a detailed presentation of the models, the underlying mathematical framework and model usage in *SpacerPlacer*.

The state space in our models corresponds to the alignment of all (observed) spacers. At each branch, the spacer array is described by a presence-absence pattern, where each spacer corresponds to a position in the input alignment.

Spacer insertion and deletion events occur along the branches of $\mathcal{T}$ according to a Poisson point process. Thus, each of the models is characterized by rate-of-occurrence parameters. The main differences between the models are the possible positions along the spacer array where new spacers can be inserted and the specific characteristics of deletion events, as described in the following.

Ordered or unordered insertions

Despite variations in the adaptation proteins Cas1 and Cas2 (Garrett 2021; Makarova et al. 2015; Makarova et al. 2020; Padilha et al. 2020), polarized insertion remains nearly universal across Cas types (Section 3.1), and we leverage this fact to improve the reconstruction in *SpacerPlacer*. Thus, we mainly consider the *ordered insertion model* corresponding to the predominant mode of insertion (Barrangou et al. 2007), where a spacer can be inserted only at the leader end.

While the ordered insertion model is clearly the more realistic model, we also introduce a simpler *unordered insertion model* to speed up and simplify parts of the guide reconstruction. In this model, new spacers can also be inserted between existing spacers. The different events of both models are illustrated for an example array in Figure 5.1.

For all models, we assume that all acquired spacers are unique. To guarantee this in the reconstruction algorithm, *SpacerPlacer* renames multiple occurrences of the same spacer and thus guarantees unique spacer IDs.

Independent or blockwise deletions

In contrast to insertions, so far no active molecular mechanism modulating spacer deletions has been identified. Replication misalignment or strand/replication slippage have been proposed as the main underlying molecular mechanisms that cause spacer deletions between repeats (Garrett 2021). Misalignment is believed to occur when two repeats erroneously

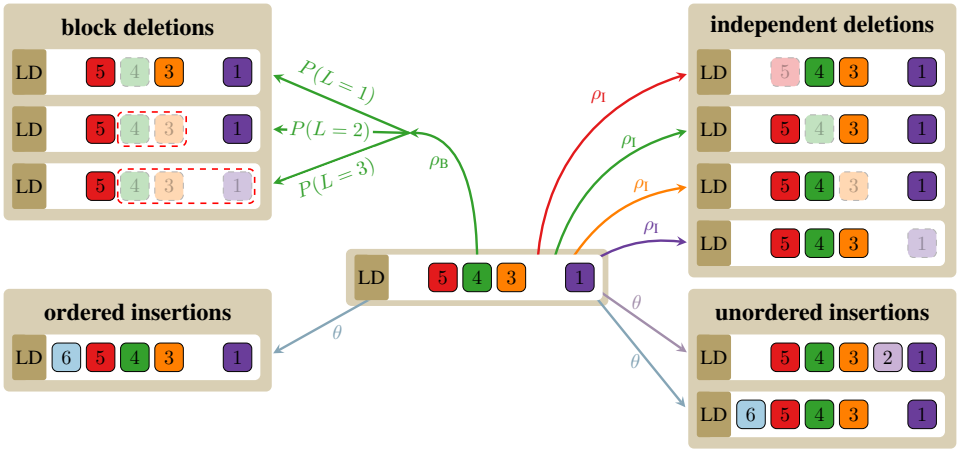


Figure 5.1: Different deletion and insertion models. The figure shows how aligned spacer arrays evolve in different models used within *SpacerPlacer*. Each colored and numbered box represents a different spacer. Unordered insertions can occur at any unoccupied position within the array (bottom right). Within the ordered insertion model (bottom left), spacers can only be acquired directly after the leader (indicated by LD) in front of the first existing spacer. In the independent deletion model (top right), each spacer can be deleted individually with rate ρ_I. Block deletions (top left) occur at each (occupied) position at rate ρ_B and $L \geq 1$ spacers are deleted jointly (including the spacer hit by the event), where L is geometrically distributed with mean α. For simplicity, we show only one block deletion event. The color of the arrows indicates at which spacer position an event was initiated. The most realistic model includes ordered insertions and block deletions (left side), while models with independent deletions and unordered insertions (right side) serve as a reference and speed up parts of the reconstruction pipeline.

align within a single replication fork (Lovett 2004). Consequently, this process is constrained to repeats that are relatively close in proximity, which aligns well with CRISPR arrays, where spacers between repeats tend to be rather short. Furthermore, this would also allow for deletions of several adjacent spacers, which we term block deletions. See Section 3.2 for more details about the deletion process.

Thus, to investigate how many spacers are typically affected by a single deletion event, we contrast two different deletion models. In the *independent deletion model (IDM)* each individual spacer s_i (of some array $s = (s_1, s_2, \ldots, s_n)$ of length n) can only be independently deleted with rate ρ_{I}. In the *block deletion model (BDM)*, deletion events also arise independently at any spacer s_i at rate ρ_{B}. However, a consecutive block of L spacers $s_i \ldots s_{i+L-1}$ is jointly deleted, where L is geometrically distributed with mean α. If L exceeds the boundary of the spacer array, the event is ignored. Note that, due to its monotonically decreasing probability mass function, longer deletions become increasingly unlikely and singular deletions remain the most likely. Further, note that setting $\alpha = 1$ yields the independent deletion model, which allows us to use the framework of likelihood ratio tests to distinguish between the IDM and the BDM. Both models are illustrated in Figure 5.1 and described in more detail in Section 4.2.

An alternative result of replication misalignment is spacer duplication or rearrangement (Bzymek et al. 2001; Lovett 2004; Treangen et al. 2009), which has also been observed for spacer-repeat units in CRISPR arrays (Garrett 2021). Here, we do not directly incorporate such more complex events into the evolutionary models. Instead, we detect and resolve them in a separate step of the reconstruction pipeline.

5.3.2 Ancestral CRISPR Spacer Array Reconstruction

Overview of the pipeline

SpacerPlacer is a parametric reconstruction and visualization tool for CRISPR spacer array evolution. An overview of the reconstruction pipeline is shown in Figure 5.2. Based on a set of overlapping spacer arrays and a provided phylogeny, *SpacerPlacer* proceeds in multiple steps to reconstruct ancestral spacer acquisitions, deletions, and more complex events, such as rearrangements.

(I) First, spacers are represented by the labels that group almost identical spacers and a multiple spacer array alignment (MSAA) is constructed based on these labels instead of aligning the DNA sequences. The spacer array alignment is described in more detail in Methods (Section 5.2) and Section 4.1. Aligning the spacer labels prevents misalignment of different spacers. The alignment of the spacer array reveals duplicated spacers as well as candidates for more complex events.

(II) The dependence between different spacer sites introduced by polarized insertion and block deletions makes computing the joint likelihoods for the reconstructions infeasible. Therefore, we first calculate preliminary ancestral states at the internal nodes of the provided phylogeny based on the less complex independent deletion model with unordered insertions.

(III) The guide reconstruction does not account for the relative acquisition-time information introduced by polarized spacer insertion, which is partially reflected in the alignment of the spacer arrays. We extract this information from the MSAA into a partial spacer insertion order (PSIO) which is explained in more detail below.

(IV) Finally, the guide reconstruction is refined so that the reconstructed insertions obey the partial order. This indirect inclusion of the insertion polarization is more efficient than calculations in a complex parametric model. Finally, the reconstructed events of putative duplicated spacers are categorized as duplications, rearrangements, reinsertions, or similar events. On the basis of such reconstructed insertion and deletion events, *SpacerPlacer* will also estimate the parameters of an IDM and the more complex BDM. Subsequently *SpacerPlacer* identifies the most likely model in a likelihood ratio test (Figure 5.4) for the dataset considered.

SpacerPlacer is available at `https://github.com/fbaumdicker/SpacerPlacer`. For additional information about the implementation, used software and packages see Supplementary Note 2. The analysis of a set of spacer arrays on a standard laptop takes usually just a few seconds. The analysis of all 518 groups from CRISPRCasdb took 515.13 seconds.

Guide reconstruction

As a first step of the ancestral spacer array reconstruction we calculate a preliminary *guide* reconstruction. As the unordered independent deletion model on spacer presence-absence profiles is a *general time reversible (GTR)* model we can use an effective dynamic programming algorithm (Pupko et al. 2000) for our ancestral guide reconstruction. As multiple independent acquisitions of the same spacer are rare (Section 3.2), we use parameters that encourage the model to prioritize multiple deletions over multiple insertions in the guide reconstruction.

The algorithm scales linearly with the number of arrays and linearly with the number of sites, in our case, the number of spacers in the MSAA (Pupko et al. 2000). Our implementation is based on *TreeTime* (Sagulenko et al. 2018) and outlined in more detail

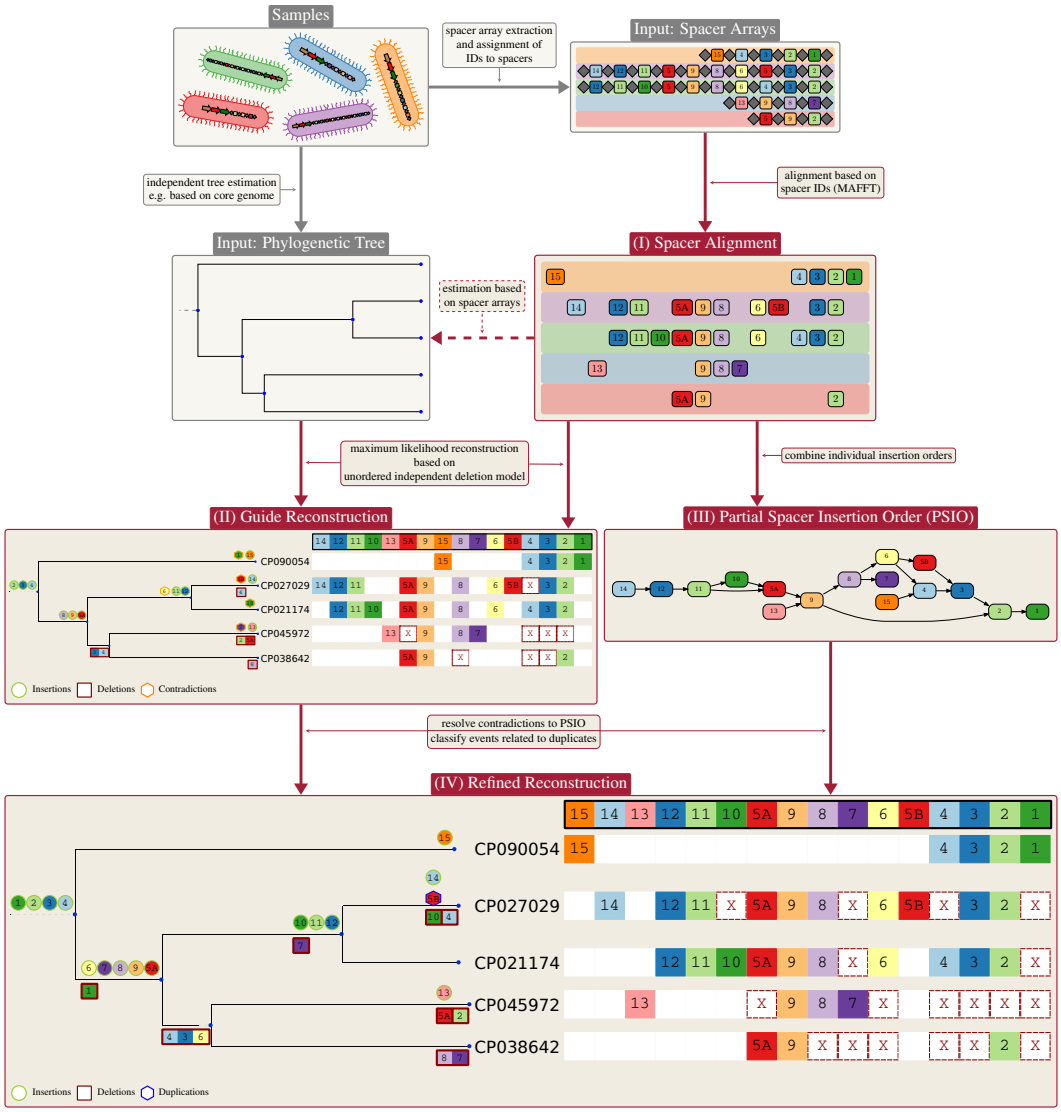


Figure 5.2: Workflow of *SpacerPlacer*: ancestral reconstruction. To prepare for the *SpacerPlacer* analysis the first step is to extract the CRISPR arrays from their sample genomes and reconstruct a phylogenetic tree, e.g. based on genome alignments of the samples. Then each unique spacer (cluster) gets assigned an ID. This step should generally be done by the user, but *SpacerPlacer* includes ID assignment for spacer arrays in a CRISPRCasdb/Finder format. As a first step, *SpacerPlacer* aligns the spacer arrays in their ID form. The spacer alignment optionally allows us to reconstruct an alternative phylogenetic tree. Then *SpacerPlacer* builds the partial spacer insertion order (PSIO) and performs a guide reconstruction along the provided tree. This is followed by refining the reconstruction using the PSIO, resulting in a finished reconstruction of ancestral acquisition, deletion and more complex events of the considered group. Preparation steps are shown in gray boxes and steps done by *SpacerPlacer* are shown in red boxes. The IDs can be chosen arbitrarily. Here, we use an ID assignment that makes the alignment more intuitively readable. The group portrayed here are CRISPR arrays of Cas type I-B from *Listeria monocytogenes* extracted from CRISPRCasdb. The tree shown is estimated based on the CRISPR arrays with *SpacerPlacer*.

in Supplementary Note 2.

The guide reconstruction cannot take advantage of all available information as it ignores the spacer insertion order. Consequently, it can still contradict polarized acquisition, as illustrated in Figure 5.3. In particular, the guide reconstruction will still underestimate the number of deletion events. However, it is a suitable starting point for the subsequent refinement steps that address these issues.

Partial Spacer Insertion Order graph construction

The partial order within the spacer arrays encodes the information that is generated from polarized insertions (Barrangou et al. 2007), one of the most consistent properties of CRISPR arrays. This has been empirically confirmed for many Cas types, with the notable exception of type II-A systems, in which ectopic spacer insertions have been observed, particularly when the leader sequence is non-functional (Achigar et al. 2017; Achigar et al. 2021; Mosterd et al. 2020).

Each spacer array constitutes a timeline of acquired spacers, i.e., an insertion order for the acquisitions of this array. Thus, by combining the order information of all arrays in a group, we can extract additional timeline information.

For example, when given two arrays, A (1-2-4-6) and B (2-3-5-6). Here, array A implies that spacer 1 was acquired after spacer 2 and array B implies spacer 2 after spacer 3. Together, we can conclude that spacer 1 was acquired after spacer 3. Note that when considering multiple spacer arrays, not all spacers need to have an order relation with each other. In the previous example, the arrays do not determine whether spacer 4 was inserted before or after spacer 5. Thus, combining the order information of multiple arrays leads only to a *partial* spacer insertion order.

The PSIO can be visualized as a graph (Figure 5.3b). In this graph, each arrow indicates a "younger than" or equivalently, "was acquired after" relation. We construct the PSIO by thinning out redundant edges from the graph based on all single-array insertion orders from the MSAA. The resulting graph is a directed graph without loops as the previous MSAA step already resolved potential conflicts. *SpacerPlacer* automatically generates a visualization of the PSIO graph with graphviz (Gansner et al. 2000).

Iterative PSIO-based refinement

The guide reconstruction does not need to be consistent with the PSIO and can contradict polarized acquisition. Therefore, we iteratively refine the guide reconstruction using three straightforward steps:

1. Identify any contradictions between the insertions estimated by the ancestral reconstruction and the PSIO. If no contradictions are found, the reconstruction is complete. Otherwise, continue with step 2.

2. For each reconstructed spacer acquisition event that is in conflict with the PSIO, determine the parent node in the tree. Regardless of the most likely state calculated at this node in the guide reconstruction, fix the spacer to be present at this node; see Figure 5.3c. This is done for all spacer positions in the MSAA and all PSIO-contradicting events along the tree.

3. Compute the maximum joint likelihood reconstruction with the added fixed ancestral states that help to resolve the contradictions. Note that the newly reconstructed

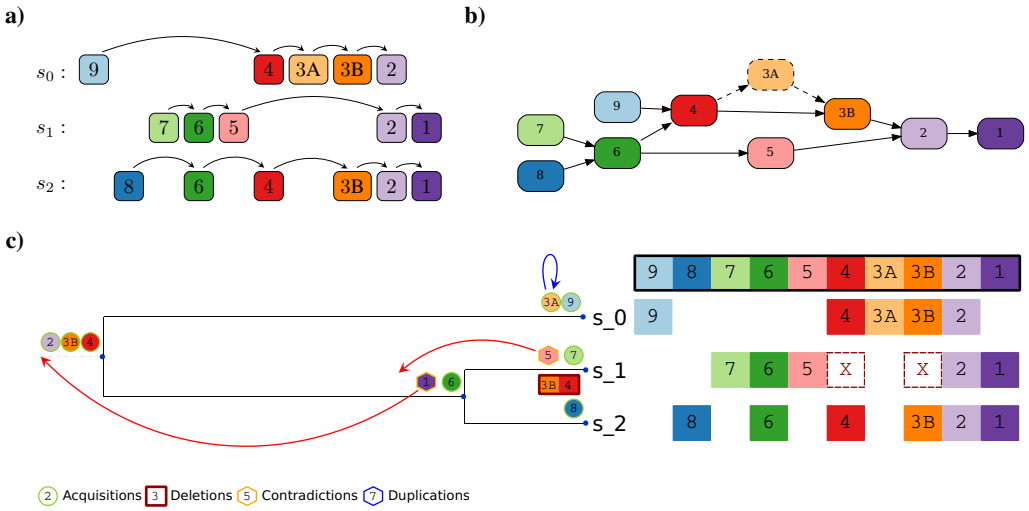


Figure 5.3: Partial Spacer Insertion Order based refinement of the guide reconstruction. **a)** Relabeled multiple spacer alignment of a group with 3 arrays s_0, s_1 and s_2, together with their induced orders. Only one step relationships are indicated. Spacers 3A and 3B are relabeled duplicates of each other. **b)** Partial Spacer Insertion Order (PSIO) as induced by the spacer array group in a). The spacer 3A is categorized as a potential duplication event occurring between 3A and 3B. Thus, it is removed from the PSIO for the refined reconstruction (dashed), and contradictions due to 3A are ignored. **c)** Illustration of the guide reconstruction and the PSIO-based refinement process. The acquisition event of 5 contradicts the acquisition of 6 in the parent branch. Similarly, the insertion of 1 is a contradiction to 2 (or 3, 4). Both contradictions are solved in the refinement process by fixing the spacer to be present at the parent node, as indicated by the red arrows. Spacer 3A is classified as a duplication event, since 3A is located in the middle of the array and spacer 3B is already present on this branch, indicated by the blue arrow.

insertion events pertaining the fixed spacers may not be assigned exactly to the fixed nodes, but can instead be inferred at nodes closer to the root. Then go to step 1. The added fixed ancestral states are retained throughout all subsequent iterations of this process.

For more details about this process see Supplementary Note 2.

Classification of spacer duplication candidates

The computed MSAA allows us to identify spacer candidates that are likely the result of spacer duplication, rearrangement, reinsertions, or independent acquisition events, since they are present in multiple columns of the alignment. Each identified candidate is relabeled such that each spacer/column in the MSAA is unique. For example, in Figure 5.3, spacer 3 has a set of two duplication candidates $\{3A, 3B\}$.

All of these event types share the characteristic of not conforming to the rules of polarized acquisition. As a result, they have the potential to cause misleading conflicts within the PSIO, e.g. when both versions of a duplicated spacer are included in the PSIO. To overcome this problem, we adopt the following approach:

Since each duplication candidate found in the MSAA is uniquely labeled, they are treated as independent spacers in the guide reconstruction. In the refinement step, only the candidate spacer that was acquired the earliest in the guide reconstruction remains in the PSIO and can produce conflicts with the PSIO in the guide reconstructions. Any conflicts to the

PSIO arising from other duplication candidates are ignored at this stage. In Figure 5.3, spacer 3B is found to be the "oldest" candidate and therefore remains in the PSIO, while 3A (dashed) is not considered to be part of the PSIO.

After refinement, candidates (excluding the "oldest" candidate), or rather their insertion events, are classified according to the reconstructed evolutionary events of all candidates. We distinguish between rearrangements, reacquisitions, duplications, independent acquisitions, and other more complex types of duplicate events. In Figure 5.3, the insertion of 3A on s_0 is classified as a duplication event, since 3B is already present at the root and 3A is in the middle of the array.

For a detailed description and illustration of the classification procedure and the different classes see Supplementary Note 2 and Figure S1. More than 96% of the spacers are consistent with standard polarized insertions and deletions (Table 5.2) and do not involve complex events, justifying the treatment of complex-event classification outside the scope of the model-based reconstructions.

Parameter estimation

We estimate the spacer deletion rate ρ_{I}, ρ_{B} and the mean block length of spacer deletions α by numerically maximizing the likelihood given by the independent deletion model or blockwise deletion model for the *reconstructed* spacer deletions.

For a group of spacer arrays, let $\mathcal{T}$ be the corresponding phylogenetic tree and N_b the number of spacers present at the parent node of branch $b \in \mathcal{T}$. Also, let K_b describe the set of reconstructed spacers that are deleted along branch b such that $|K_b|$ is the total number of spacers deleted at branch b. Moreover, let $\mathcal{K} := \{K_b \,|\, b \in \mathcal{T}\}$ and $\mathcal{N} := \{N_b \,|\, b \in \mathcal{T}\}$ be the collections of these sets for all branches in the tree.

We can compute the likelihood $L_{\mathrm{B}}(b, N_b, K_b, \rho, \alpha)$ of deletions K_b to occur along branch b, while retaining $N_b - |K_b|$ spacers, under the block deletion model with deletion rate ρ and mean block length α. Then the total likelihood of a group under the BDM is given by

$$\mathcal{L}_{\mathrm{B}}(\mathcal{T}, \mathcal{N}, \mathcal{K}, \rho, \alpha) := \prod_{b \in \mathcal{T}} L_{\mathrm{B}}(b, N_b, K_b, \rho, \alpha). \tag{5.1}$$

To compute the likelihood under the IDM we simply set $\alpha = 1$, i.e. $\mathcal{L}_{\mathrm{I}}(\mathcal{T}, \mathcal{N}, \mathcal{K}, \rho) := \mathcal{L}_{\mathrm{B}}(\mathcal{T}, \mathcal{N}, \mathcal{K}, \rho, 1)$.

The (analytic) computation of the likelihood function of the block deletion model L_{B} is not trivial. The main issue to solve is that a connected component of consecutive spacers that have been lost along a branch of the phylogeny can arise from a combination of multiple block deletion events of different lengths. The two approaches we use to calculate the combined likelihood are described and illustrated in Section 4.4 and Figure 4.3 (or Supplementary Note 4 and Figure S2). Furthermore, we show that one of the approaches performs clearly worse (Supplementary Note 4, Figure S3).

As the results are based on the branch lengths of the phylogenetic tree, the estimated deletion rate $\hat{\rho}_{\mathrm{B}}$ is proportional to the time scale of this tree. In the core genome-based trees used here, branch lengths are scaled by the expected number of substitutions per site. In this scaling, an estimated spacer deletion rate of 374 indicates that a spacer is deleted 374 times more frequently than a single site mutates.

Bias correction for unobserved extinct spacers

Spacers that existed at some time in the past can get deleted such that these spacers are not observed in any of the sampled spacer arrays. Consequently, deletion events related to such spacers cannot be reconstructed. This results in an underestimation of the deletion rate ρ in both the IDM and the BDM. Therefore, we correct the likelihood for such unobserved spacers by conditioning on maintaining the spacers in at least one of the samples that are descendants of the individual where the spacer has been acquired, similar to an approach introduced by Joseph Felsenstein in a different setting (Felsenstein 1992).

For the BDM the exact computation of this unobserved likelihood is hard due to the possibility of arbitrarily long deletions originating from unobserved spacers. Thus, we approximate the effect of the conditioned likelihood in the BDM using the correction factor from the IDM. For a description of the computation of the correction factor see Section 4.4.4 (or Supplementary Note 4D). The effects of unobserved spacers on parameter estimates are significant (Supplementary Figure S8), but our bias correction can effectively address this issue.

Moreover, the maximum likelihood estimator of a geometrically distributed random variable is known to be biased, especially for a low number of observations. Therefore, in addition to the IDM-based bias correction for unobserved spacers, we use a standard bias correction for the estimated mean spacer deletion block length α after the numerical estimation step; see Section 4.4.4 (or Supplementary Note 4D).

The improvement of parameter estimates by the combination of both bias corrections is evident when comparing Figure 5.5 and Supplementary Figure S9.

Likelihood ratio test

Based on the likelihoods computed in the previous section, we formulate a maximum likelihood ratio test to investigate within *SpacerPlacer* whether spacers are (likely) deleted in blocks or individually.

Here, a likelihood ratio test can be used, as the independent deletion model is nested in the block deletion model (setting $\alpha = 1$). We formulate the hypotheses which correspond to the independent and block deletion model

$$H_0 : \quad \rho_\mathrm{B} > 0, \alpha = 1 \quad (\text{i.e. } \rho_\mathrm{I} > 0), \quad (\text{IDM}) \tag{5.2}$$

$$H_1 : \quad \rho_\mathrm{B} > 0, \alpha \geq 1, \quad (\text{BDM}) \tag{5.3}$$

where the test statistic is given by

$$\lambda_\mathrm{LR} := \frac{\sup_{\rho_\mathrm{I}>0} \prod_{b\in\mathcal{T}} L_\mathrm{B}(b, N_b, K_b, \rho_\mathrm{I}, 1)}{\sup_{\rho_\mathrm{B}>0,\alpha>1} \prod_{b\in\mathcal{T}} L_\mathrm{B}(b, N_b, K_b, \rho_\mathrm{B}, \alpha)}. \tag{5.4}$$

If λ_LR is sufficiently small, we dismiss H_0 on $\mathcal{T}$ and accept H_1, i.e. the block deletion model fits the data better and is more likely. Otherwise, we cannot dismiss H_0, i.e. both models explain the data reasonably well. The test procedure is illustrated in Figure 5.4.

See Section 4.4 (or Supplementary Note 4) for a more detailed description of the likelihood calculation and the likelihood ratio test.

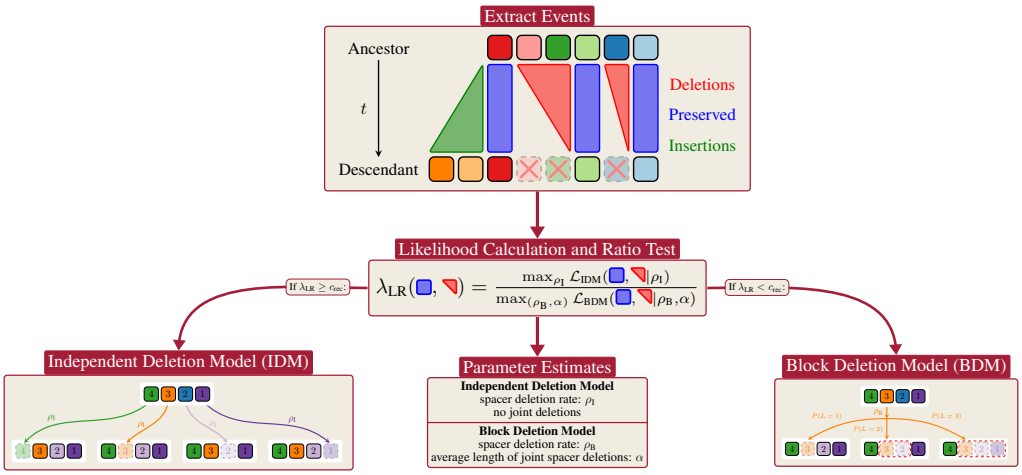


Figure 5.4: LRT workflow in *SpacerPlacer*. Using a parametric model allows *SpacerPlacer* to combine ancestral reconstruction with rate estimation and likelihood ratio tests. After reconstructing the ancestral states with *SpacerPlacer* we extract the deletion events and preserved spacers, as well as the lengths of the tree branches. These events are then used to generate maximum likelihood estimates for the parameters of both deletion models the IDM and the BDM. *SpacerPlacer* also uses a likelihood ratio test to identify whether the IDM can be rejected in favor of the BDM.

Parametric models enable accurate estimation of the rates and lengths of spacer deletions

We evaluated the performance of *SpacerPlacer* using simulated data (Figure 5.5). *SpacerPlacer* performs well in estimating the (median) deletion rates in simulations and counteracts the bias of likelihood models towards too few events by incorporating the information from the PSIO, indicating a well-performing reconstruction algorithm (Figure 5.5c, d, and e).

The likelihood ratio test reliably distinguishes independent individual spacer deletions compared to joint deletion of adjacent spacers, as soon as the average length of jointly deleted blocks of spacers exceeds 1.6 in the simulations (Figure 5.5a).

In simulations where spacer deletions occur in blocks, assuming an IDM results in an underestimation of the per-spacer deletion rate (Figure 5.5b). This highlights the importance of incorporating block deletions into the parametric model to accurately quantify both the frequency and length distribution of spacer deletions.

When considering the effective per spacer deletion rate, calculated as the product of ρ_B and α (Figure 5.5c), the IDM and BDM estimates become directly comparable. Under this comparison, the BDM clearly outperforms the IDM (compare Figure 5.5b and c).

Furthermore, using the BDM, *SpacerPlacer* accurately recovers the parameters ρ_B and α (Figure 5.5d and e). The estimation quality decreases only for very high α values, where deletion events that collapse (almost) the entire array become more frequent.

SpacerPlacer showed similar performance for simulations based on the parameter estimates and core-genome trees from the CRISPRCasdb dataset (Supplementary Figure S10).

The estimation of parameters in the BDM is non-canonical due to the dependencies created by jointly lost neighboring spacers (see Section 4.4 or Supplementary Note 4). In addition, we need to use bias corrections for both ρ_B, α. BDM-based estimates without bias corrections tend to underestimate ρ_B and α, as can be seen in Supplementary Figure S9.

Due to the typically short length of spacer arrays, the variance of estimated rates and block lengths is high for individual groups, suggesting that the estimated parameters are

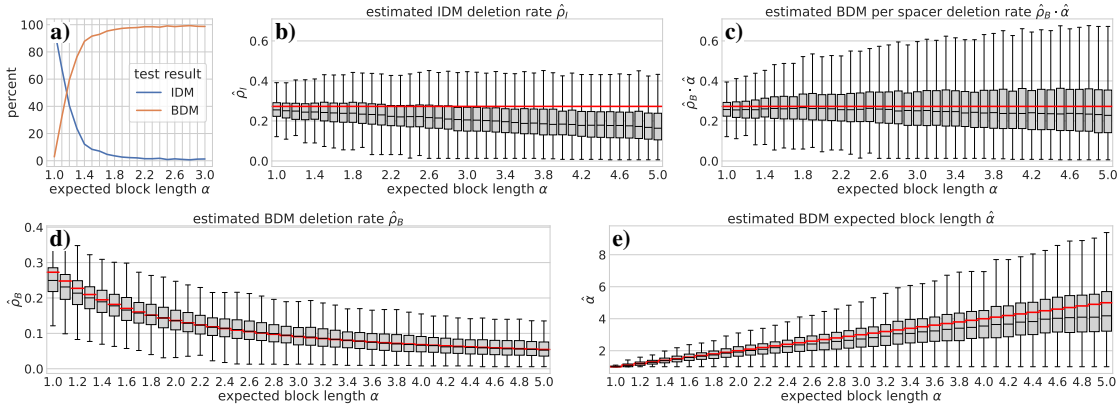


Figure 5.5: Parameter estimation and LRT-performance on simulated data. We evaluated *SpacerPlacer* on simulated spacer arrays. The simulations are based on random coalescent trees for different average lengths of blocks of jointly deleted spacers (α), where the deletion initiation rate ρ_B in the BDM was always chosen such that the per spacer deletion rate $\rho_B \cdot \alpha$ remains constant for different choices of α. True parameters are shown as red lines. *SpacerPlacer* was then used to reconstruct the ancestral spacer arrays and estimate the corresponding parameters (using the simulated coalescent trees and including both bias corrections), shown as boxplots here.
a) The percentages of likelihood ratio tests (LRTs) that indicate a BDM (orange line) or do not reject the IDM (blue line) are shown. For $\alpha > 1.6$ the likelihood ratio test in *SpacerPlacer* can correctly recognize the BDM used in the simulation (ground truth). **b)** Estimates of the spacer deletion rate ρ_I based on the IDM. One can clearly see that the accuracy of the estimated per spacer deletion rate (difference to red line) drops with the expected block length α used in the simulations. This indicates that the IDM is not sufficient to predict the accurate average deletion rate per spacer when deletions occur in larger blocks. **c)** Estimates based on the BDM for the per spacer deletion rate $\rho_B \cdot \alpha$. As expected, parameter estimation based on the BDM performs better than the estimation based on the IDM in part b). Furthermore, the performance is strong for all values of α, only decreasing for high values of α. Overall *SpacerPlacer* estimates the per spacer deletion rate $\rho_B \cdot \alpha$ well in the median. **d)+e)** Estimates of BDM deletion rate (bottom left) and BDM block lenght α (bottom right) separately. The result from part c) can be decomposed into the estimates of ρ_B (bottom left) and α (bottom right). As their product is fixed, the underlying ρ_B used in the simulations (red curve) decreases with growing α. This indicates that not only the per spacer deletion rate, but also the deletion initiation rate and the deletion length are well estimated in the median. There is a trade-off between ρ_B and α since both parameters are estimated jointly.

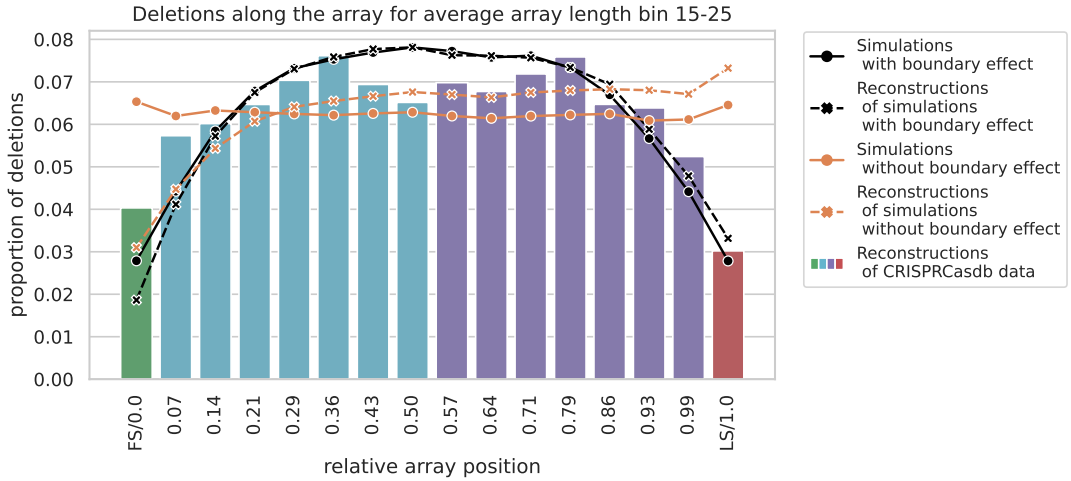


Figure 5.6: Proportions of deletions with respect to the position along the array. The solid orange line shows the proportion of deletions for simulations under an artificial block deletion model, where deletions of internal spacers and spacers at the ends of the array occur at the same frequency. In contrast, the solid black line shows the proportion of deletions for simulations under the more natural block deletion model where block deletions cannot exceed the array boundary as nearby repeats need to align. Here, deletions are less likely to occur towards both ends. The dashed lines show the corresponding values based on the events reconstructed by *SpacerPlacer* in colors corresponding to the respective models. In both cases *SpacerPlacer* is able to reconstruct the deletion distribution of the respective models, except near the leader end of the array. The underlying histogram shows the distribution of deletions in reconstructions based on CRISPRCasdb data which is similar to the deletion model with a boundary effect (solid and dashed black lines). But there are more reconstructed deletions at the first spacer position than at the last spacer position, contrasting the pattern we would expect based on the neutral simulations. For each category, we bin all deletion events according to their relative position compared to the reconstructed lengths of the array. Each point or bar corresponds to a bin, with the label on the x-axis indicating the upper limit of the bin. The y-axis shows the proportion of deletions falling within the respective bin. The first spacer (FS/0.0) and the last spacer (LS/1.0) are placed in their own bins (shown in green/red).

more reliable in well-curated, large groups and when averaging over multiple CRISPR array groups.

5.3.3 Analyzing the patterns of spacer array evolution across numerous bacterial genomes

The benefits of our modeling framework are best highlighted in the context of the dynamics of spacer deletions and the postulate that such deletions are caused by replication slippage. Deletions that arise from repeat misalignment introduce three effects that can be detected in the inferred parameters and reconstructions based on the BDM (Section 3.2). First, near both ends of the array, the observed deletion frequencies will be reduced because repeats at the boundary of the array have a lower number of nearby repeats with which they can align for a deletion event. We call this the "array boundary effect" on the deletion frequency. Second, misalignments spanning multiple repeat-spacer units can result in patterns where the simultaneous deletion of multiple adjacent spacers can be observed. In this case, the likelihood ratio test should favor the BDM and indicate how frequent and long such deletion blocks are. Third, deletions by repeat misalignment are, a priori, independent of the Cas genes, i.e. the Cas type.

Distribution of deletions along CRISPR arrays aligns with deletions caused by replication misalignments between repeats

To investigate the distribution of deletions according to their position in the spacer array, we compared *SpacerPlacer* reconstructions for two different simulation scenarios (Figure 5.6). We either assumed that spacer losses occur evenly distributed along the array without the need to pair different repeats or, in the second scenario, that spacers at the edges of the array are less often part of block deletions according to the lower number of neighbor spacers and repeats that could mismatch with them (Section 3.2). To avoid counter-intuitive binning effects, we only show groups with an average array length between 15 and 25 in Figure 5.6. We provide similar results for further array lengths, a differently scaled visualization, and additional details in Supplementary Figures S11, S12 and S13.

The PSIO is particularly advantageous for reconstructing older spacers, as younger spacers provide additional information about the relative timing of insertion events. In both simulation scenarios, the accuracy of ancestral reconstruction towards the leader end, where new spacers are acquired, is thus lower than towards the trailer end of the array (Figure 5.6). Therefore, we expect that reconstruction at the leader ends for CRISPRCasdb data are similarly impacted.

The frequency of reconstructed spacer deletions based on CRISPRCasdb data shows a fall-off towards both boundaries of the array with a roughly uniform distribution in the middle of the array (Figure 5.6). This is different from what we would expect from a distribution that is directly dependent on the length of the array, such as the fragment deletion model proposed by (Kupczok et al. 2013) or models that favor deletions on one side of the array (Chen et al. 2022; Collins et al. 2023). Interestingly, a uniform deletion rate along the array was also found to be a representative model for spacer deletions in a related simulation study (Han et al. 2013). The uniform distribution in the middle combined with the drop towards both ends of the array is consistent with a pairwise repeat-dependent deletion process (Section 3.2, Garrett (2021)) with geometrically distributed deletion lengths.

Blockwise deletions of neighboring spacers are frequent

Several theories and mechanisms have been proposed to explain the distribution of joint deletions (Garrett 2021; Kupczok et al. 2013; Kupczok et al. 2015; Stout et al. 2018). Until now, however, statistical tests have not been able to determine the most likely scenario, as the deletion patterns observed in different CRISPR array datasets are consistent with different models for spacer deletions. In particular, both the IDM and BDM contain losses of adjacent spacers that can arise from combinations of multiple overlapping deletions; only their frequency and distribution change.

To determine the mode of spacer deletions in CRISPR arrays, we applied the likelihood ratio test described in a previous section to the datasets extracted from CRISPRCasdb and reconstructed the ancestral CRISPR arrays to investigate the properties of spacer array evolution.

We found that in 76% of the CRISPR array groups, the likelihood ratio test rejected the null hypothesis, i.e. the IDM, in favor of the BDM. For a breakdown of the test results with respect to Cas type; see Supplementary Figure S14. Although the IDM cannot be rejected in the remaining cases, this indicates that, overall, the block deletion model describes the natural process of spacer deletions significantly better.

We inferred the distribution of the lengths of the joint deletions in our reconstructions based

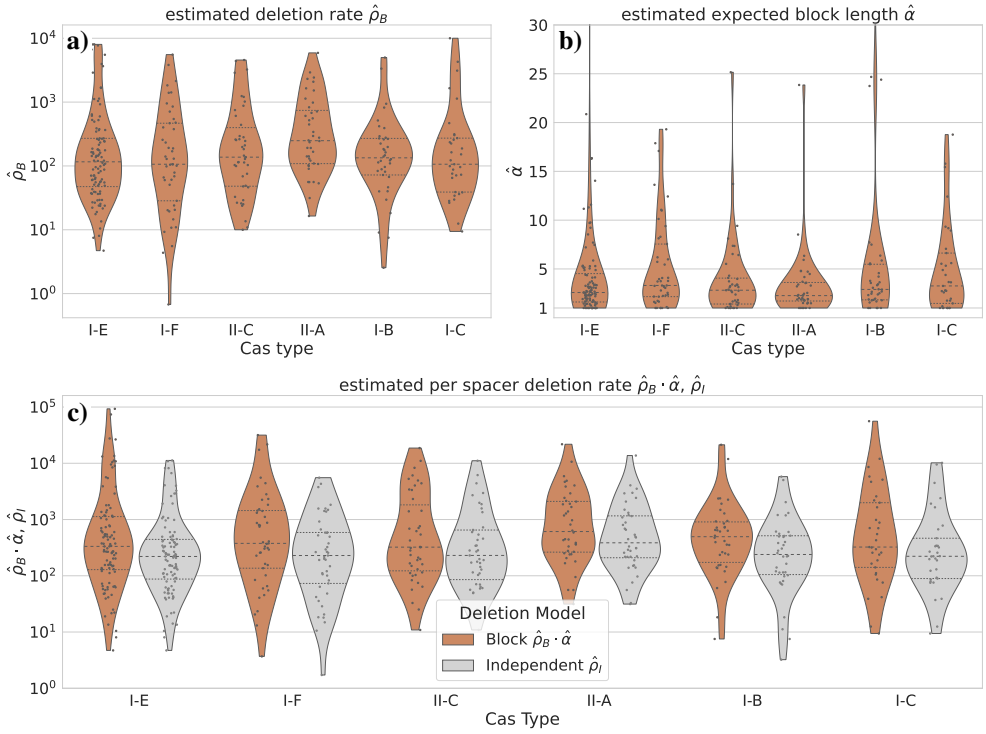


Figure 5.7: Parameter estimates on CRISPRCasdb data. The estimated parameters for **a)** the deletion initiation rate, and **b)** the expected length of deletion events in a block deletion model, as well as **c)** the resulting per spacer deletion rate for different Cas types are shown. There are no significant differences between the Cas types. The variation of individual estimates is high, but the median estimates of the per-spacer deletion rate are consistently similar. For comparison, we also show the estimates, if the underlying model is the less accurate IDM (light gray) in c). Although IDM-based estimates correlate with BDM-based results, the same bias as in the simulations can be observed. See Table 5.1 for the global mean and median parameters.

		mean	median
BDM deletion length	α	4.58	2.73
BDM deletion rate	ρ_{B}	601	137
IDM deletion rate	ρ_{I}	895	230
BDM per spacer deletion rate	$\alpha \cdot \rho_{\mathrm{B}}$	2602	374

Table 5.1: Mean and median parameter estimates in the independent and block deletion model. We estimated the parameters of the independent and block deletion model on CRISPRCasdb data across all groups with at least one reconstructed deletion; see also Figure 5.7. Top: Estimates for the BDM. In the BDM deletions are initiated at a rate ρ_{B} and cover a geometrically distributed number of spacers with mean α. Middle gray: Estimate of the deletion rate ρ_{I} of the IDM. In this model, all initiated deletions have length 1. Bottom: $\alpha \cdot \rho_{\mathrm{B}}$ is the approximate deletion rate per spacer in the BDM and roughly corresponds to ρ_{I} in the IDM.

on CRISPRCasdb data (Supplementary Figure S15). The inferred empirical distribution is consistent with a geometric distribution for smaller lengths but, notably, does have a heavier tail. Thus, we hypothesize that the deletion process may, to a lesser extent, involve additional mechanisms, such as fragment deletions (Kupczok et al. 2013) and the loss of entire CRISPR arrays.

For the expected block length parameter α the median of all estimated parameters based on the CRISPRCasdb datasets is $\alpha \approx 2.73$. This is consistent with our statistical analysis of the block deletion lengths observed by Lopez-Sanchez et al. (2012). The length of the spacer blocks jointly deleted in their experiment fits a geometric distribution (p-value $= 2.76 \cdot 10^{-5}$) and has a mean deletion length of $\alpha = 3.32$.

Therefore, in all subsequent analyzes, we will only present the results derived from the more accurate BDM, unless otherwise indicated.

Spacer deletions are frequent and Cas type-independent

The estimated spacer deletion rates and lengths are comparable for the different Cas types investigated here (Figure 5.7), indicating that, no Cas-dependent molecular mechanism influences the deletion of spacers. We find similar results with respect to genus (Supplementary Figure S16).

We can estimate the evolutionary turnover speed of CRISPR arrays because the core genome trees used for the estimates are time-scaled based on the average per site mutation rate. In the BDM, we estimate the deletion initiation rate ρ_B and the mean block length α of deletions. From these we can calculate the average deletion rate per spacer $\rho_{\mathrm{B}} \cdot \alpha$. Global median and mean estimates are shown in Table 5.1. The median of the spacer deletion rate per spacer in the BDM is 374, indicating that a spacer is deleted at a rate 374 times higher than the rate at which individual nucleotide mutations occur. In particular, since spacers (and their repeats) are generally smaller than 100 bp in length, the deletion of spacer sites occurs at a faster rate than nucleotide mutations within the array. The loss of immunity by spacer deletion is therefore the greater risk for the host compared to mutations within the spacer.

We can also give an ad hoc estimate of the median insertion rate $\theta \approx 6522$ by considering the following. As each individual spacer is lost at a mean rate of $\rho_B \cdot \alpha$, an overall spacer insertion rate of $n \cdot \rho_B \cdot \alpha$ will result in a balance that sustains an average array length

	Count	Proportion
Total number of spacers	24,090	
PSIO-compatible spacers	23,285	0.967
Duplication candidates	805	0.033

Breakdown of duplication candidates by event type

	Count	Prop. of candidates
Duplications	695	0.863
Rearrangements	43	0.053
Reacquisitions	8	0.010
Independent insertions	7	0.009
Other events	52	0.065

Table 5.2: Number and proportions of duplication events. The total number of spacers of the CRISPRCasdb dataset is divided into spacers that are compatible with polarised insertions and deletions only, and spacers that are duplication candidates according to the MSAA and the PSIO. Duplication candidates are further classified into reconstructed complex event types.

of $n \approx 17.4$. Note that the insertion rate represents the total rate of acquisition of new spacers and thus is naturally significantly higher than the per spacer deletion rate.

This indirect rough estimate of the average insertion rate should be considered with extreme caution, especially for single groups, where the variance of the estimates is large (Supplementary Figure S17). As the frequency of spacer acquisitions depends on environmental fluctuations such as co-evolution with phages, we would expect the insertions to occur in bursts with varying rates (Section 3.1), while the model assumes a constant rate. Thus, it remains unclear how an inferred average acquisition rate can be interpreted.

In contrast, spacer deletions can take place without the involvement of any foreign elements, which suggests that they can be better explained by a constant rate (Section 3.2). However, together the estimates indicate that spacer turnover is fast.

Spacer duplications and rearrangements are rare

Spacers that can be explained solely by insertions and deletions were by far the most common case in the CRISPRCasdb datasets, but we also identified roughly 3% of all spacers that form loops in the PSIO and are only explainable by more complex event types, as shown in Supplementary Note 2, Figure S1. Of these complex events, 86% are simple duplications, 5% are rearrangements, and only 1% new insertions of the same spacer (Table 5.2). This aligns with the finding of Kupczok et al. (2015) that recombination between CRISPR arrays is rare.

No evidence of age-dependent selection along the spacer array, except possibly at the first and last positions

If selection pressures would differ along the array, e.g. if more anciently acquired spacers would provide a higher or lower fitness benefit than more recently acquired spacers, one would expect that the observed deletion rate along the array correlates with this selective advantage. It is important to distinguish between internal spacers and spacers that are close to the first or last position. For the later, we expect a strong additional fitness-independent

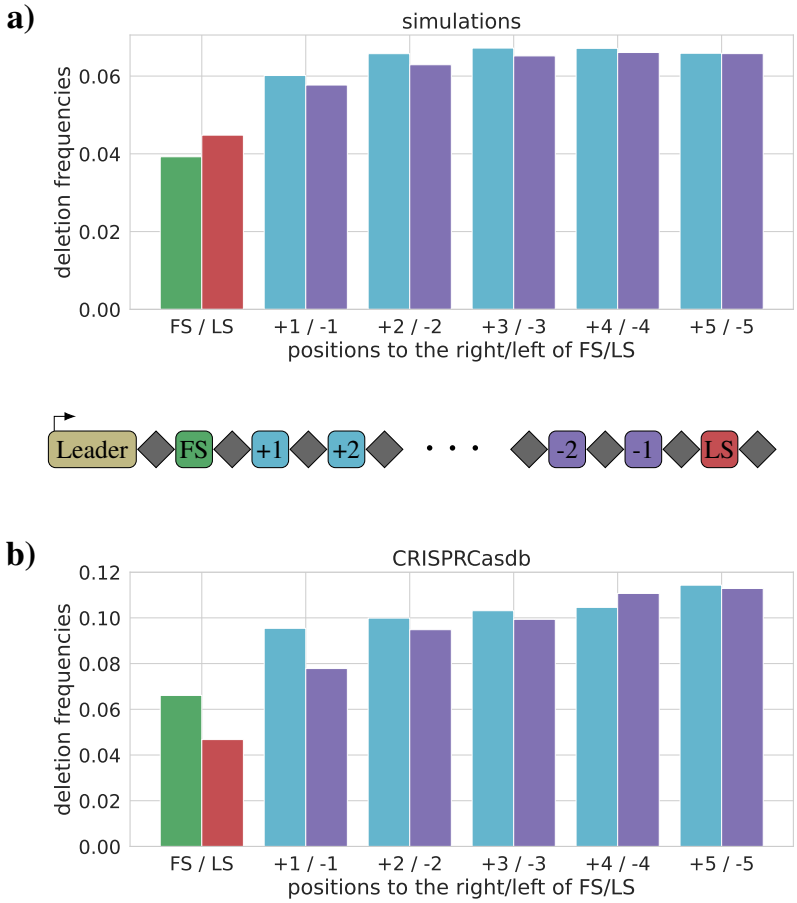


Figure 5.8: Deletion frequencies with respect to the proximity to the array boundary. We show the deletion frequencies for absolute positions next to the first and last spacer. **a)** shows the reconstructed deletion frequencies based on simulations with array boundary effect. The first spacer (FS) appears to be deleted less often than the last spacer (LS), although the simulation is perfectly symmetric. But, just as in Figure 5.6, the reconstruction is more likely to underestimate the frequency of FS deletions. **b)** shows the deletion frequencies for reconstructions based on CRISPRCasdb data. Here, the pattern is reversed, and the LS is less frequently deleted than the first spacer. Comparing the FS and LS deletion frequencies between a) and b) suggests that compared to a symmetric model, either the LS is deleted less frequently, the FS is deleted more frequently, or both.

influence of the underlying repeat-dependent blockwise deletion process which causes lower deletion rates towards the array boundary. When looking at the relative position of reconstructed deletions, the frequency of deletions are overall in line with the boundary effect in the BDM (Figure 5.6). At least for spacers that are not among the very first or last positions, the absence of a clear sign of a selective gradient indicates that the selective pressures to maintain a particular beneficial spacer and the fitness costs to keep a useless spacer are either constant over time or small.

For the first (and last) spacers, the task of inferring the selection strength is particularly challenging, as the bias created by the reconstruction is not symmetric and can obscure the analysis (Figure 5.6). To take this reconstruction effect into account, we compared the frequencies of reconstructed CRISPRCasdb deletions to reconstructions from simulated

spacer arrays. Of the perfectly symmetric simulated deletions *SpacerPlacer* reconstructs fewer deletions at the first position compared to the last position. In contrast, the CRISPRCasdb data show more reconstructed deletions of the first two spacers compared to the last two spacers (Figure 5.8). This suggests that the spacer deletion rate is higher towards the leader end. In the discussion, we describe potential selective benefits of losing recently acquired spacers and alternative explanations that could cause this difference.

5.4 Discussion

Comparison to other models of spacer array evolution

Multiple modeling approaches explored the coevolutionary dynamics between phages and bacteria (Chen et al. 2022; Han et al. 2013; Koonin et al. 2015) and the length of spacer arrays (Martynov et al. 2017; Pavlova et al. 2021). In addition, some stochastic models have been specifically designed for CRISPR spacer array evolution (Baumdicker et al. 2018; Kupczok et al. 2013). Kupczok et al. (2013) suggested a spacer deletion model in which any fragment of neighboring spacers of the array is equally likely to be deleted. Here we propose geometrically distributed block deletion lengths which fit well with observed deletion lengths in CRISPRCasdb and experiments (Lopez-Sanchez et al. 2012) and allow efficient likelihood computations and parameter estimations.

Collins et al. (2023) introduced a reconstruction tool based on maximum parsimony principles based on empirically determined scores for different events in CRISPR array evolution. In contrast, we rely on explicit evolutionary models that can offer insight beyond ancestral reconstruction through parameters that correspond to molecular mechanisms and evolutionary processes.

Additional discussion and an overview of the models used in related studies and different steps in our pipeline are provided in Section 4.3, 4.5 and Table 4.1 (or Supplementary Note 5 and Table S1).

Using the partial spacer insertion order is crucial to reconstruct ancestral spacer deletions

Inferring the dynamics of spacer array evolution is challenging, as arrays are typically short and provide very limited information. Furthermore, the acquisition of new spacers correlates with environmental conditions, such as the abundance of phages. In contrast, spacer deletions are less dependent on the environment, and thus are more likely to follow a molecular clock.

Fortunately, the order of spacers provides additional information and repeated acquisitions of the same spacer are extremely rare and thus can be safely ignored (Section 3.1). These two properties form the basis for the reconstruction approach. *SpacerPlacer* first reconstructs the most likely acquisition point(s) and the resulting deletions along the tree independently for each spacer. And afterwards refines these according to the PSIO.

Enforcing the chronological order of spacer insertion during the refinement step based on the PSIO counteracts the tendency of likelihood models to explain patterns by fewer (deletion) events. This allows *SpacerPlacer* to reconstruct ancestral spacer arrays with greater precision and improves rate estimates. In the CRISPRCasdb data, more than a quarter of the spacers (6,773 out of 24,090) are affected, i.e. their acquisition events are relocated during the refinement procedure. This results in a higher number of reconstructed

spacer deletions that are subsequently used by *SpacerPlacer* to estimate the maximum likelihood parameters in the more complex evolutionary model with blockwise spacer deletions.

Spacer deletion patterns are independent of the Cas type

Our results indicate that joint deletions are a common phenomenon in CRISPR array evolution and show no significant differences in their deletion rates or lengths between Cas types I-B, I-C, I-E, I-F, II-A, and II-C based on CRIPSRCasdb data. This suggests that spacer deletions are not controlled or regulated by CRISPR-specific molecular mechanisms, but rather occur in a random fashion. Quite contrary to this view, frequent spacer deletion and renewal of spacer composition is essential for the effectiveness of the system (Martynov et al. 2017) and could, in principle, have fostered the emergence of a dynamic deletion system similar to primed adaptation. Interestingly, deletion frequencies can be subject to selection indirectly when multiple CRISPR arrays with different lengths, and therefore, due to the boundary effect, different per spacer deletion rates coexist (Toms et al. 2017; Weissman et al. 2018b).

Although we did not find significant differences here, the parametric models behind *SpacerPlacer* will allow future studies on a larger CRISPR database to identify potential differences in deletion patterns.

The parameters estimated in groups of spacer arrays from CRISPRCasdb reveal two interesting characteristics of spacer array evolution. First, the deletion rate per spacer is 374 times higher than the mutation rate per site, highlighting that immunity loss via spacer deletion acts on a much faster time scale compared to mutations within spacers.

Second, in line with deletions that arise from repeat misalignments, blocks of jointly deleted spacers occur frequently, and their length distribution is consistent with a geometric distribution with mean 2.7 that is interrupted if the block length exceeds the spacer array. Nonetheless, single-spacer deletions remain the most common events (37% for $\alpha = 2.7$).

Reconstructed duplications shed light on the mechanisms of spacer deletions

Our analysis of CRISPRCasdb spacer evolution showed that complex events, including duplications, rearrangements, and reacquisitions, are rare and dominated by duplication events. Only 3% of the considered spacers have been reconstructed as duplicated, and 171 out of 333 CRISPR array groups do not include a potential duplication.

The number of observed duplications is much lower than we would expect from experiments in Lopez-Sanchez et al. (2012) where roughly 10% of the arrays had a duplication. There are two explanations: 1) Our model only provides a lower bound for such events, and 2) duplicated spacers provide an additional possibility for replication slippage since the two identical spacers can also misalign. It is conceivable that duplicated spacers are thus lost at a higher rate than nonduplicated spacers, leading also to a lower rate of detected duplications. This observation adds an interesting aspect to the postulate that most spacer deletions occur due to replication misalignment of the repeats (Garrett 2021; Lovett 2004).

Deciphering selection pressures from deletion patterns in spacer arrays

One of the main questions motivating this work was whether spacers at different positions in the spacer array are under different selection pressures. The parametric model-based approach allows us not only to investigate the likelihood of evolutionary forces and molecular

mechanisms at play, but also to reveal hints at selective pressures acting on the presence and absence of spacers at different positions in the array.

It is important to distinguish the position along the reconstructed ancestral array and the current position among the observed sample arrays. When looking at the deletions in the spacer arrays of a sample, these seem to appear predominantly at the distal end. However, deletions of older spacers are naturally observed more often in such alignments, since they existed for a longer time during which deletions can occur compared to more recently acquired spacers. At the time when the deletions occurred, their position was closer to the leader end. This observation bias has led to the perception within the scientific community that deletions occur more frequently among shared spacers at the end of the array, rendering them less beneficial than more recently acquired spacers, which seem to be less often deleted.

Our results indicate that this impression is a misinterpretation of the accumulation of observed deletions towards the distal end in alignments of current spacer arrays. Here, we correct this misleading impression by instead considering the reconstructed deletion positions in ancestral spacer arrays.

We observe no clear sign of selection acting differently on deletion frequencies along the reconstructed arrays, at least for spacers that are not among the first or last positions. While we by no means propose that CRISPR arrays evolve neutrally, our results hint at either overall weak selection pressures to delete specific spacers or an equally strong selection along the array to maintain spacers.

We found that the drop in deletion frequency towards the edge of the array is particularly pronounced for the last spacer compared to the first spacer. This difference could be caused by the often observed excessive mutation of the last repeat (Horvath et al. 2008), which renders them less likely to participate in deletion events based on misaligned repeats.

Vice versa, we can argue that the difference between the last and first spacers arises because of selective effects at the leader end. As we can only reconstruct deletions from individuals whose offspring survived, this could indicate that many recently acquired spacers are not beneficial in the long run and can be deleted again quickly. Another selective advantage of deleting spacers at the leader end could arise due to the expression levels along the spacer array. Modifying the composition of spacers at the first most expressed positions might be particularly beneficial and could lead to the observed increased deletion rates at the first spacer positions.

Interestingly, our findings are in line with recent observations by López-Beltrán et al. (2024). They showed that the environmental abundance of the phages and plasmids targeted by spacers varies along the spacer array. They found that at the first positions, the targeted phages are present in the environment, whereas the targets of the spacers directly following are less abundant. This indicates a lower selection pressure to maintain recently acquired spacers. Toward the trailer end they observed that the spacers correspond to more abundant but different phages and plasmids, which fits well with the reconstructed deletion frequencies here.

Dataset quality and accuracy of the underlying tree affect spacer array reconstructions

Provided there is sufficient overlap to form a PSIO, including incomplete arrays should not significantly affect the reconstruction. However, the impact of an incorrect tree on the reconstructed ancestral arrays could theoretically be large. Misplaced tips can cause

chain reactions during the refinement step, such that more and more spacer insertions are wrongly moved toward the root.

However, closely related sister arrays are rarely placed far apart and a minor misplacement will cause limited damage. In addition, such effects are apparent and likely to be observed by a knowledgeable user. Comparing the spacer-array based tree from *SpacerPlacer* with the core phylogeny can be a good starting point for this. Such a comparison can also help identify whether multiple spacers within a sample are artifacts of the CRISPR detection process, such as a mistakenly split spacer array, or related coexisting CRISPR arrays (Weissman et al. 2018b). The user can then resolve these problems by adjusting the tree, excluding arrays, removing specific spacers, or adjusting the parameters of *SpacerPlacer*.

Lastly, since the number of deletion events is typically small, parameter estimates based on a single dataset have a high variance. This is readily apparent in the estimates on CRISPRCasdb data (Figure 5.7) and simulations (Supplementary Figure S10). In general, we recommend trusting parameter estimates only in the median over a large number of arrays and groups using properly time-scaled trees.

Applications and current limitations

SpacerPlacer is a rapid and versatile tool for studying CRISPR arrays and systems, and we expect it to provide valuable insights in a variety of different application scenarios.

For example, we inferred that the deletion rate for CRISPR spacers, $\rho_B \cdot \alpha = 374$, is two orders of magnitude higher than the per-site mutation rate. This evolutionary speed and the robustness of complete spacers with regard to sequencing errors make CRISPR arrays a valuable fine-scale typing tool (Muzyukina et al. 2024), as has already been demonstrated for multiple species (Barrangou et al. 2016; Karah et al. 2015; Karimi et al. 2018; Rykachevsky et al. 2021).

When tracking the evolution of clinically relevant samples and investigating outbreaks, it can be beneficial to complement CRISPR typing with a reconstruction of their evolutionary relationships. While standard multilocus sequence typing (MLST) is often insufficient to reconstruct the evolutionary history of closely related samples, CRISPR arrays are rich in phylogenetic information due to the PSIO. This concept was also discussed by Tomida et al. (2017), who applied standard phylogenetic reconstruction tools on CRISPR arrays, which provided a higher resolution than the trees generated using MLST. The currently implemented block deletion model based distance in *SpacerPlacer* already solves some of the issues found in the CRISPR based trees of Tomida et al. (2017) (Supplementary Note 6A, Figure S4), and we anticipate that incorporating more advanced tree inference methods will further improve the results.

As a rough estimate of the discriminatory power of spacer arrays consider the following estimate. The number of deletions is proportional to the per spacer deltion rate times the length of the MSAA n (≈ 38 in the CRISPRCasdb dataset). Then to achieve a similar discriminatory power, a set of genes used for MLST needs to be of combined sequence length $n \cdot \rho_{\mathrm{B}}\alpha \approx 38 \cdot 374 \approx 14200$. Thus, spacer arrays have a roughly three to four times higher discriminatory power than MLST, where, typically around 3500 bp are considered (Enright et al. 2000). Clearly, core-genome MLST and whole-genome based discrimination exceed this threshold and are thus more precise. However, spacer arrays have a high discriminatory power compared to their relative length and, unlike point mutations, provide temporal information encoded in the PSIO. Moreover, spacer arrays are easy to find and less susceptible to sequencing errors because of their repeat structure.

Monitoring bacterial samples and their co-evolution with phages based on CRISPR arrays is also of importance for bacteria-driven biotechnological and pharmaceutical production processes, such as bioreactors (Martínez Arbas et al. 2021) or wastewater treatment plans (Guerrero et al. 2021) and other metagenomic time series (Guerrero et al. 2021; Martínez Arbas et al. 2021; Sun et al. 2016). However, CRISPR arrays derived from short-read sequencing and metagenome assembled genomes are often incomplete. Naturally, the rate estimations will be obscured when using partial spacer arrays, but *SpacerPlacer* will still provide a partial reconstruction of ancestral spacer arrays. When complete arrays are available through long-read sequencing on metagenomic data or have been carefully reconstructed from short reads (Rykachevsky et al. 2021), *SpacerPlacer* benefits from the complete PSIO increasing the precision of ancestral array reconstruction and rate estimation. For example Lam et al. (2019b) characterize CRISPR arrays and the corresponding PSIO graphs from a gut microbiome sequenced with long-read sequencing technologies. In this situation, *SpacerPlacer* can reliably infer the ancestral relationships for such datasets and reconstruct the corresponding spacer insertion and deletion events (Supplementary Note 6B, Figure S5).

In conclusion, *SpacerPlacer* is a new parametric ancestral reconstruction tool that enables in-depth investigations of CRISPR array evolution. Analyzing spacer arrays of closely related samples and the correspondence between spacers and invading elements within the reconstructions will allow us to track pathogens at a fine scale and shed light on the coevolution of phages and bacteria, the mechanisms of spacer acquisition and deletion itself, and the dynamic ecology of microbial communities and microbiomes.

6

An evolutionary approach to predict the orientation of CRISPR arrays

Chapter with co-authors
This chapter is based on Fehrenbach et al. (2025), with minor modifications, and accordingly contains contributions by the co-authors. Note that references to supplementary notes and figures refer to the supplementary material of Fehrenbach et al. (2025).

Chapter summary (Abstract)

CRISPR-Cas is a defense system of bacteria and archaea against phages. Parts of the foreign DNA, called spacers, are incorporated into the CRISPR array which constitutes the immune memory. The orientation of CRISPR arrays is crucial for analyzing and understanding the functionality of CRISPR systems and their targets. Several methods have been developed to identify the orientation of a CRISPR array. To predict the orientation, different methods use different features such as the repeat sequences between the spacers, the location of the leader sequence, the Cas genes, or PAMs. However, those features are often not sufficient to predict the orientation with certainty, or different methods disagree.

Remarkably, almost all CRISPR systems have been found to insert spacers in a polarized manner at the leader end of the array. We introduce *CRISPR-evOr*, a method that leverages the resulting patterns to predict the acquisition orientation for (a group of) CRISPR arrays by reconstructing and comparing the likelihood of their evolutionary history with respect to both possible acquisition orientations. The new method is independent of Cas type, leader existence and location, and transcription orientation. *CRISPR-evOr* is thus particularly useful for arrays that other CRISPR orientation tools cannot predict confidently and to verify or resolve conflicting predictions from existing tools.

CRISPR-evOr currently confidently predicts the orientation of 28.3% of the arrays in the considered subset of CRISPRCasdb, which other tools like CRISPRDirection and CRISPRstrand cannot reliably orient. As our tool leverages evolutionary information we expect this percentage to grow in the future when more closely related arrays will be available. Additionally, *CRISPR-evOr* provides confident decisions for rare subtypes

of CRISPR arrays, where knowledge about repeats and leaders and their orientation is limited.

6.1 Introduction

One of the defining features of CRISPR arrays is that new spacers are almost always inserted at the leader-proximal end. The leader is a non-coding sequence of variable length (up to several hundred base pairs), typically located at the 5' end of the array. Even though leader sequences exhibit limited sequence conservation and their structural motifs remain poorly characterized, they play a central role in regulating both spacer acquisition and transcription in (almost) all CRISPR arrays;(Alkhnbashi et al. 2016; Garrett 2021; Pougach et al. 2010; Yosef et al. 2012). We discuss the leader and its role in acquisition and expression in Chapter 2, and the importance of polarized acquisition for array evolution in Chapters 3, 4 and 5.

6.1.1 Importance of orientation for CRISPR systems

DNA and RNA are transcribed and synthesized in the 5'–3' direction. Consequently, the orientation of CRISPR arrays is critical for multiple aspects of their function, including spacer expression as pre-crRNA, processing into crRNA, and the mechanisms of interference and adaptation, and thus is of particular importance for understanding different Cas types (Chapter 2). In particular, correct orientation is essential for identifying leader regions, transcription start and termination sites, protospacer orientation, and PAM sequences (Alkhnbashi et al. 2016; Buyukyoruk et al. 2025; Vink et al. 2021). Furthermore, inserting foreign arrays into new organisms in the wrong orientation may cause them to be mischaracterized as non-functional in experiments.

Array orientation is also crucial for studying CRISPR ecology and evolution. Spacer acquisition is polarized in most systems (Mojica et al. 2005; Wang et al. 2023), meaning that an incorrect orientation reverses the inferred spacer order and can lead to erroneous evolutionary conclusions (Chapter 5).

6.1.2 Orientation concepts and tools for CRISPR-Cas systems

In the following, we introduce the concepts and tools used to predict CRISPR array orientation. Different tools may yield conflicting predictions, either because they rely on distinct orientation concepts or because they produce inconsistent results even when based on the same principle. A brief overview of these concepts and tools is provided in Table 6.1. The shown orientation tools are often part of comprehensive CRISPR system detection tools which we introduced in Section 2.1.3.

The *transcriptome* can be used to infer the orientation of arrays (Houenoussi et al. 2020), however, this information is typically not available. We thus focus on other available concepts and tools here.

In most known CRISPR arrays found in nature, there are Cas genes in close proximity. Thus, Milicevic et al. (2019) suggest to predict the CRISPR orientation based on the *Cas-orientation*. The Cas-orientation is determined by the transcription orientation of the Cas genes close to a respective CRISPR array (< 1000 bp distance) (Milicevic et al. 2019).

Orientation concept	Description and Tools/Papers using the concept
acquisition	Find the end of the array where new spacers are inserted by analyzing the evolution of groups of CRISPR arrays. **Tool:** *CRISPR-evOr* (new)
Cas	Determine the transcription orientation of Cas genes close to the array and identify their orientation with the array orientation. **Reference:** Milicevic et al. (2019)
PAM	Find PAMs of CRISPR systems and locate the PAMs on matched targets of spacers to determine the orientation. **Reference:** Vink et al. (2021)
repeat	Characterize the repeats according to available biological knowledge and their relation to transcription orientation. In particular, analyze the mutation pattern of the repeats along the length of the array. **Tools:** CRISPRstrand (Alkhnbashi et al. 2014), CRISPRidentify (Mitrofanov et al. 2021), CRISPRmap (Lange et al. 2013)
leader + repeat	Use empirically measured characteristics to analyze the repeats and locate the leader sequence by analyzing the regions adjacent to the CRISPR array. These characteristics are weighted and combined to generate a prediction. **Tools:** CRISPRDirection (Biswas et al. 2014), CRISPRCasFinder (Couvin et al. 2018), CRISPRDetect (Biswas et al. 2016)
transcriptome	Predict orientation by employing transcriptome sequencing to find the transcription orientation of arrays. **Tool:** TOP (Houenoussi et al. 2020)

Table 6.1: Orientation concepts overview. We show all orientation concepts in use for CRISPR arrays that are referenced within this work. We give them labels to simplify referencing the different orientation concepts. We also provide references for the concepts and tools that rely on the respective orientation concept.

Furthermore, the *leader-orientation* can be determined from the position of the characteristic *leader* sequence, which is typically located at the 5' end of the CRISPR array (Alkhnbashi et al. 2016).

A closely related orientation concept is based on the *repeats*. The *repeat-orientation* is characterized by mutation patterns along the array. Due to the molecular functionality of the (polarized!) insertion and deletion process of spacers, repeats degrade in 5'-to-3' direction by accumulating mutations (Biswas et al. 2014), moreover, the mutations are concentrated at the ends of repeats (Alkhnbashi et al. 2014).

Both the leader- and repeat-orientation are closely related concepts and one of the most commonly used prediction tools for CRISPR arrays, *CRISPRCasFinder* (Couvin et al. 2018), relies on both leader- and repeat based orientation methods. More precisely, *CRISPRCasFinder* employs a combination of leader- and repeat-orientation (Couvin et al. 2018) and incorporates *CRISPRDirection* (Biswas et al. 2014). The orientation is predicted by, first, trying to find the leader by searching for relative AT richness at both ends of the array and comparing the distance to the closest coding genes at both ends. Second, *CRISPRDirection* considers specific sequence motifs and mutations, the RNA secondary structure, AT content in the repeats, and degeneracy in the 3' end of the array (Biswas et al. 2016; Couvin et al. 2018).

CRISPRidentify (Mitrofanov et al. 2021) is another commonly used tool to find CRISPR systems that relies on repeat based orientation by incorporating *CRISPRstrand* (Alkhnbashi et al. 2014). CRISPRidentify does not use any additional information about the leader for the orientation prediction, but CRISPRleader within CRISPRidentify can annotate the leader by using the repeat-orientation predicted by CRISPRstrand (Alkhnbashi et al. 2016). CRISPRstrand uses domain expert knowledge about repeats to identify important features. It was observed that it is useful to partition the consensus repeat into different blocks as the mutation pattern along the consensus repeat strongly diverges. CRISPRstrand encodes the consensus repeats as graphs, which additionally, contain information about mutations and their position in the repeat. Then they train a graph kernel model on the encoded graphs of a curated dataset (Alkhnbashi et al. 2014).

Vink et al. (2021) identified the orientation of many CRISPR arrays by identifying the respective protospacers of the spacers in the array and *locating the protospacer adjacent motifs* (PAM). PAMs are (generally) 3 nucleotide long sequences at one side of the protospacer. They are used by the CRISPR-Cas system to correctly distinguish between spacers and protospacers to avoid autoimmunity (Section 2.2.1). These sequences are quite specific to the systems Cas-(sub)type.

While all of these concepts are viable ways to predict the orientation, they not necessarily agree and do not always provide a reliable basis for a confident prediction. For example, for some CRISPR arrays Cas genes will be reversed or no Cas genes are located closely to the array, a leader sequence does not exist (Alkhnbashi et al. 2016), no protospacers can be found, or the CRISPR arrays are too short to analyze the repeats. In particular, type II-C systems have been shown to be peculiar in multiple ways: they acquire spacers at the 3' end (opposite to the expected leader end) (Hoikkala et al. 2021; Hooton et al. 2015; Laanto et al. 2017; Zhang et al. 2013), carry promoters in the repeats (Dugar et al. 2018; Hoikkala et al. 2021; Zhang et al. 2013) and show peculiar repeat mutation patterns (Mitrofanov et al. 2026) all of which could possibly interfere with accurate orientation inference. Similarly, type II-B systems, which have been shown to aid bacteria in evading host immune detection, show evidence that spacer insertion may occur at the opposite end of the array (Milicevic et al. 2019; Sampson et al. 2013; Sampson et al. 2019).

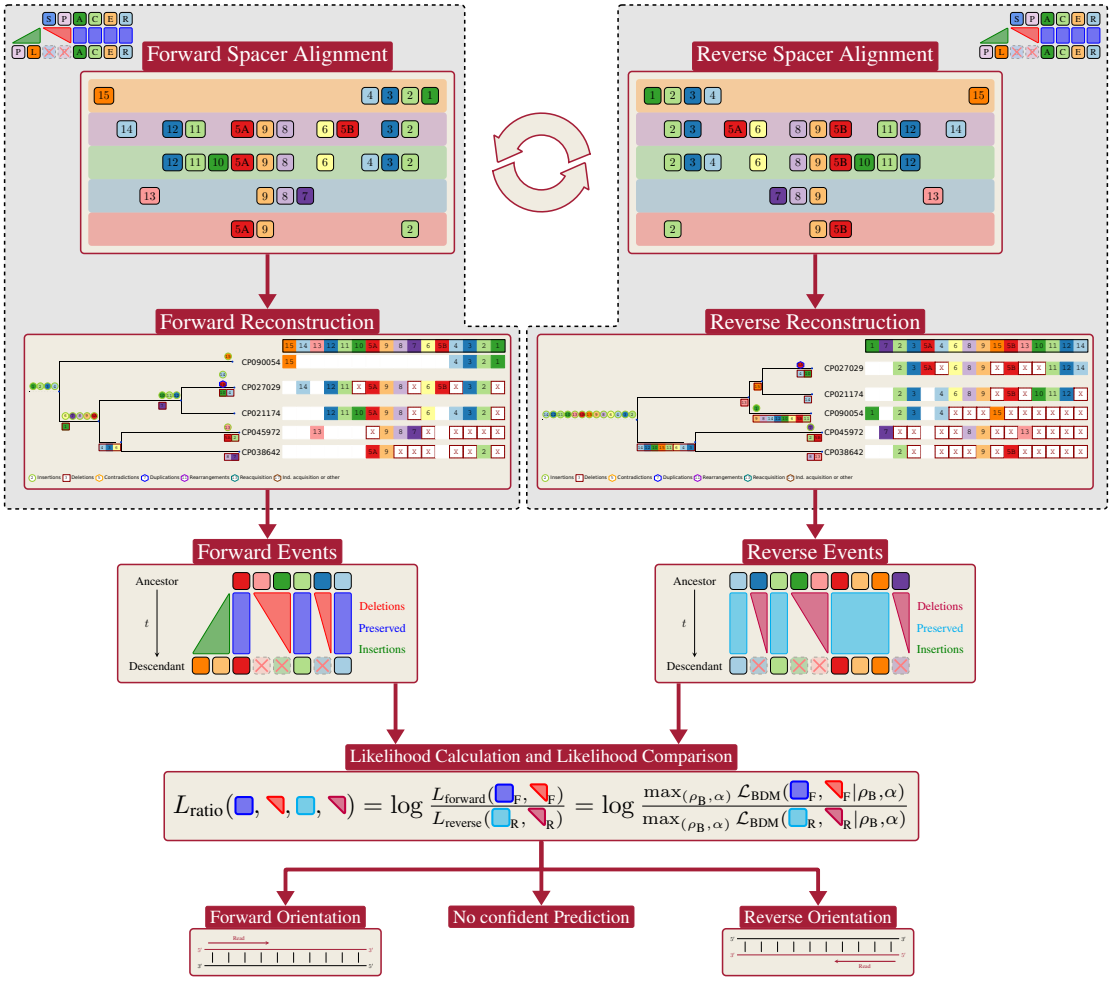


Figure 6.1: *CRISPR-evOr* workflow. We show the steps taken by *CRISPR-evOr* to arrive at a prediction for a given group of spacer alignments. The left and right column show the same steps of reconstructing the ancestral history of a spacer alignment with *SpacerPlacer*. They differ only by their orientation, i.e. the spacer alignments are turned around compared to each other. Note, that *SpacerPlacer* is heavily reliant on the partial spacer insertion order (PSIO) and thus, this difference can have major impact on the reconstruction. Furthermore, in this case here, no tree was provided to *SpacerPlacer* and thus, *SpacerPlacer* estimates a different tree based on the spacer arrays for each orientation. After both reconstructions are performed, *SpacerPlacer* obtains maximum likelihoods of the respective reconstructions based on the reconstructed events. Then by comparing the ratio of both likelihoods, *CRISPR-evOr* decides for one of the orientations or, if the likelihoods are too close, determines that no confident prediction can be made.

Consequently, novel concepts and methods for array orientation are necessary to expand the range of confidently orientable arrays.

6.1.3 Acquisition-orientation as a holistic evolutionary approach to predict array orientation

In contrast to features used in previous orientation-prediction approaches, we exploit the nearly universal property of CRISPR array evolution: the polarized acquisition of new spacers (Section 3.1). In most systems, spacer insertion occurs at the leader end, i.e. the 5' end (Barrangou et al. 2007; McGinn et al. 2019). This polarization generates characteristic evolutionary patterns in the reconstructed ancestral histories of closely related CRISPR arrays.

We propose *CRISPR-evOr* a method that identifies what we call the *acquisition-orientation*, the most likely end of acquisition. This orientation concept is closely related to the other orientation concepts, in particular, the leader- and repeat-orientation. However, instead of relying on markers in single CRISPR arrays or genomes, *CRISPR-evOr* uses a holistic approach that relies on reconstructions of the evolution of a group of CRISPR arrays. For the reconstructions, we rely on our tool *SpacerPlacer* (Chapter 5). *SpacerPlacer* reconstructs the ancestral history of spacer acquisitions and deletions of a group of arrays (with spacer overlap) along a phylogenetic tree by using maximum likelihood methods and leveraging the spacer insertion order (inferred from the arrays).

CRISPR-evOr is based on a simple heuristic. We apply *SpacerPlacer* (Chapter 5) twice. First, the evolution is reconstructed for the arrays in the given orientation. Then we reverse the CRISPR array spacer sequence and compute a second reconstruction. Finally, *CRISPR-evOr* compares the likelihoods of each reconstruction provided by *SpacerPlacer* to predict the most likely acquisition-orientation. In this way, we acquire an orientation prediction by consensus decision of all CRISPR arrays in the group. The workflow of *CRISPR-evOr* is illustrated in Figure 6.1. Figure 6.2 shows an example with a high confidence prediction where, clearly, the acquisition-orientation shown in A can be favored over the reverse as shown in B, simply due to the much smaller number of reconstructed evolutionary events.

We apply *CRISPR-evOr* on a large database of CRISPR arrays extracted from CRISPR-Casdb (Pourcel et al. 2019) and find that *CRISPR-evOr* confidently predicts orientations for arrays where the aforementioned orientation methods have low confidence. Quantitatively, *CRISPR-evOr* can predict the orientation of 28.3% of all arrays where CRISPRDirection and CRISPRstrand have low confidence. Thus, our tool increases the number of arrays where orientations can be reliably estimated.

Furthermore, on data where high confidence orientation predictions are already available, *CRISPR-evOr* helps to either reinforce the predictions, or resolve contradicting predictions of other tools, and thereby uncovers discrepancies among different orientation concepts dependent on the Cas type. We find that among the array subsets where multiple tools are confident, predictions by *CRISPR-evOr* almost always agree with other tools for two of the most abundant Cas types (I-E and I-F) in CRISPRCasdb. In particular for types I-B and II-A, *CRISPR-evOr* agrees more often with CRISPRstrand than CRISPRDirection. The differences indicate that CRISPRDirection is well-attuned only for arrays of some Cas types. Moreover, *CRISPR-evOr*, PAM-orientation, CRISPRDirection, and CRISPRstrand often disagree with each other for type II-C, which hints at differences between acquisition-orientation and leader-, repeat- and PAM-orientation for type II-C.

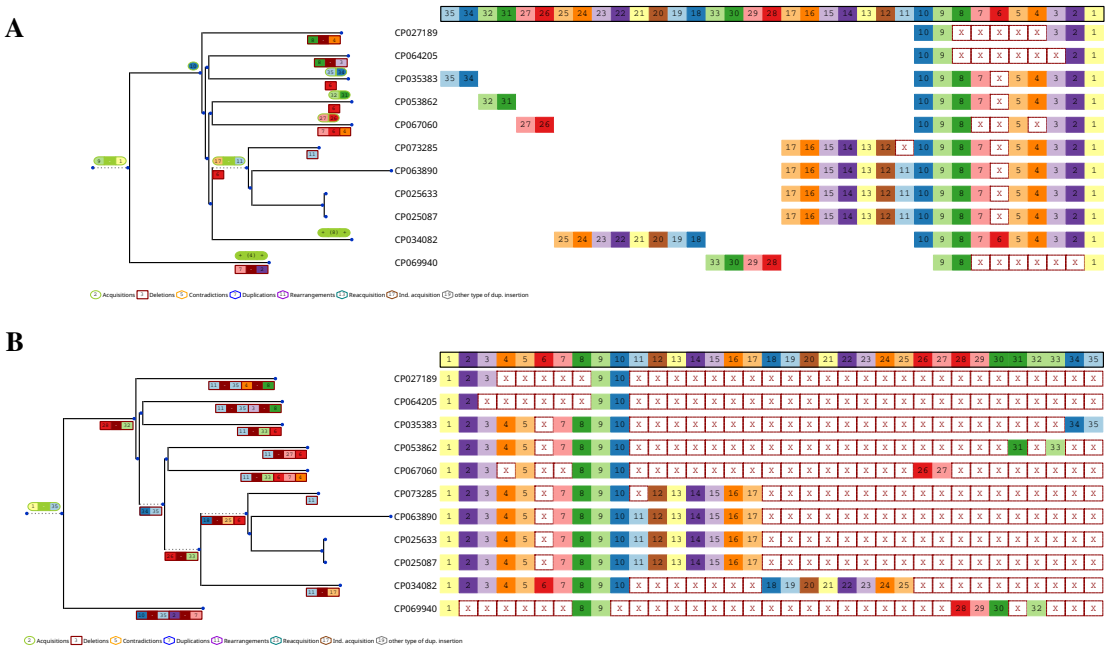


Figure 6.2: Example of a group of arrays with high confidence orientation prediction. **A** and **B** show reconstructions of the same alignment of spacer arrays. B has the orientation provided by CRISPRCasdb, while A is simply the reconstruction of the same arrays, but arranged in reverse order. Note, that in B all spacers are reconstructed to be inserted (green) at the root and thus the exceptional amount of deletion events (red) along the tree (also shown in red in the spacer alignment). The insertions are produced through the first spacer 1 (light yellow) which is present at the root, as it is present in all arrays, and "pulls up" all other spacer insertions through the spacer insertion order. In contrast, in A the insertions are distributed more evenly along the tree and thus the reconstruction avoids a lot of deletions. Comparing both reconstructions clearly indicates that the orientation of A is more likely under the assumption of polarized insertions. To make the figures more concise, acquisition and deletion events in the tree are consolidated, e.g. 9 - 1 in green/red indicates that spacers 9 through 1 were acquired/deleted at this branch. At the leafs "+ (x) +" indicates that x spacers were acquired. They are the leftmost spacers in the alignment for the respective leafs (which are not found in any other array). The shown example is a group of *Klebsiella pneumoniae* (Cas type I-E) from CRISPRCasdb with small adjustments for readability.

6.2 Methods

6.2.1 Database and available orientation predictions

We rely on the CRISPRCasdb dataset collected by us in Chapter 5 (Fehrenbach et al. 2024). It is based on high-confidence arrays extracted from CRISPRCasdb (Pourcel et al. 2019) in July 2022.

For each array the orientation prediction by CRISPRDirection was extracted from CRISPRCasdb and CRISPRstrand was applied to obtain the respective orientation predictions. These arrays were grouped successively by consensus repeat, spacer overlap, genus and Cas type to yield a large comprehensive dataset. For the grouping by spacer overlap, spacers were clustered as described in Section 5.2.1. Then groups were removed if they did not share any spacer overlap between arrays or if their arrays were completely identical. Thus, each group contains only arrays of the same consensus repeat, genus, Cas type and overlapping spacer content. Examples of such groups are shown in Figure 6.2 and Figures B, C in S1 Material.

The dataset contains a total of 518 groups with 5934 arrays. A breakdown of the Cas types in the dataset is shown in Figure A in S1 Material. Figure 6.4 shows the Cas type distribution for arrays with high confidence orientation predictions by *CRISPR-evOr*, CRISPRstrand and CRISPRDirection in this dataset.

For more details about the preparation of the groups and the dataset in general, see Section 5.2 and Supplementary Note S1A in Fehrenbach et al. (2024). Note that in Chapter 5 we investigated and estimated parameters of the deletion process on a subset (334 groups, 4565 arrays with at least one deletion in the reconstruction) of the larger dataset used here.

Additionally, we used an updated version of the dataset collected by Vink et al. (2021), who extracted their dataset from CRISPRCasdb in February 2020. The updated version uses the same dataset of spacers, but contains more confident orientation predictions. The old version is available in their supplementary material.

The database contains 4266 bacterial and archaeal genomes and a total of 221,591 spacers and their respective consensus repeat sequences. They subsequently matched these spacers against the NCBI nucleotide database and large metagenomic databases to obtain target matches. They then investigated the flanks of the matches to find the associated PAM sequences and predict the orientation of spacers based on the location of the respective PAM at the matching target's flanks.

We matched the repeats and respective spacers against our dataset to allow us to compare the orientation predictions. Thus, the dataset used for the evaluation of PAM-orientation versus *CRISPR-evOr* is the intersection between the dataset from Chapter 5 and Vink et al. (2021)'s dataset, both of which are part subsets of CRISPRCasdb. Furthermore, only 38.9% (2309) of the arrays and 77.2% (400) of the groups in the CRISPRCasdb dataset have PAM-orientation predictions. The Cas type diversity of this dataset is shown in Figure 6.3) and can be observed in Figures E, J, N, O in S1 Material.

6.2.2 Ancestral reconstruction with *SpacerPlacer*

The backbone of *CRISPR-evOr* is the ancestral reconstruction algorithm for groups of closely related CRISPR spacer arrays *SpacerPlacer* introduced in Chapter 5.

In short, *SpacerPlacer* first aligns labeled spacer arrays into a Multiple Spacer Array Alignment (MSAA) and derives a Partial Spacer Insertion Order (PSIO) from the inferred

chronological relationships between spacers. It then performs ancestral reconstruction in two steps: an initial maximum likelihood guide reconstruction under a straightforward IDM with unordered insertions, followed by iterative refinement to enforce PSIO constraints. The resulting refined reconstruction provides inferred acquisition and deletion events that can be visualized and used for downstream evolutionary analyses, such as the estimation of deletion model parameters. Figure 6.2 shows examples for such refined reconstructions.

6.2.3 Computation of likelihoods with *SpacerPlacer*

SpacerPlacer is able to compute the likelihoods of reconstructions according to their proposed block deletion model (BDM), which we found to be a suitable model for spacer deletions in statistical analysis on the CRISPRCasdb dataset (Section 5.3 and 5.4). We use the simplified likelihood function and default parameters computed on the CRISPRCasdb dataset, as is recommended (Fehrenbach et al. 2024).

6.2.4 Tree estimation

We estimated trees based on the single-copy core genes of the corresponding sample for each group as is stated in Section 5.2. We used Bakta to annotate genomes (Schwengers et al. 2021) and then the pangenome analysis tool panX (Ding et al. 2018) and IQ-TREE (Nguyen et al. 2015) to reconstruct trees. We used these core genome trees for all experiments and figures unless otherwise stated.

If no tree is provided by the user, *CRISPR-evOr* uses the included functionality of *SpacerPlacer* to estimate trees based on the CRISPR arrays. In that case, *CRISPR-evOr* estimates the tree for each orientation, i.e. reconstruction run, separately since the trees themselves depend on the acquisition orientation and might differ substantially. See Section 5.2.1 for a description of the tree inference procedure used in *SpacerPlacer*.

6.2.5 Simulations

Simulations were performed according to the methods described in Section 5.2. First, a spacer array with length distributed according to the (rough approximation) of the stationary distribution of the BDM is sampled. Subsequently, starting at the root of a given tree, the array is evolved along the tree branches by sampling acquisition and deletion events. The acquisitions are polarized and the deletion events follow the BDM.

To evaluate the performance of *CRISPR-evOr* we simulated two datasets:

Dataset A: Here, phylogenetic trees were sampled according to the coalescent and we simulated datasets with reasonable parameter values as described in Section 5.2.2. Dataset B: For a more realistic setting, we used the core-genome based trees estimated in Section 5.2 for the CRISPRCasdb dataset and simulated datasets using the respective estimated evolutionary parameters (Fehrenbach et al. 2024). Note that the parameters estimated in Chapter 5 were estimated for groups with at least one observed deletion events, and thus all groups in the simulations have non-zero acquisition and deletion rates.

More information and precise numbers can be found in Note A in S1 Material. Additionally, information about the simulation process can be found in Section 5.2.2 and Supplementary Note S1 of (Fehrenbach et al. 2024).

6.2.6 Simulation-based benchmarking

In simulations along coalescent trees and moderate spacer acquisition and deletion parameters (dataset A) we found that *CRISPR-evOr* can be completely relied upon with 100% of the decision being correct for decision threshold $c = 0$. Furthermore, almost all simulations exceed any reasonable choice of threshold c.

To investigate the reliability of *CRISPR-evOr* in a more realistic setting (dataset B), we tested *CRISPR-evOr* providing the original trees (inferred from core genes (Section 5.2) and upon which we simulated the dataset). However, since reliable trees are not always available, we also tested the performance based on trees inferred by *SpacerPlacer* based on the spacer arrays themselves.

When using the original tree and decision threshold $c = 0$, we found that 97.8% of the predictions were correct. Of the remaining 2.2%, only 0.2% give the wrong orientation prediction and 2% likelihood ratio exactly 0, i.e. no or symmetric events. When using the trees estimated based on the spacer arrays, we found that 95.5% of the predictions were correct and 1.5% have likelihood ratio exactly 0, similar to before. As the estimated trees based on the arrays can be of low resolution and diverge from the true tree, especially for groups with few observed evolutionary events, it comes as no surprise that *CRISPR-evOr* predicted the wrong orientation for 3.0% of the groups, i.e. a larger proportion of groups. However, imprecise tree inference does not change the underlying observed events and thus does not change results by a huge margin.

Naturally, when a decision threshold $c > 0$ is applied, the percentage of correct and wrong predictions decreases and some of the orientation predictions are labeled as uncertain instead. We report the results on the above simulation datasets with a conservative choice of $c = 5$ in Note B in S1 Material.

6.2.7 Confidence threshold c

We manually compared the forward and reverse reconstructions for CRISPRCasdb groups with low $|L_{\text{ratio}}|$ to find a conservative candidate for the confidence threshold with $c = 5$.

For additional evidence that the threshold is suitable, we considered the difference in the number of deleted spacers between forward and reverse reconstructions for each group in the CRISPRCasdb dataset. In general, we find that the number of differences increases rapidly with increasing $|L_{\text{ratio}}|$ (Figure D in S1 Material).

For $|L_{\text{ratio}}| = 0$, we typically do not have differences in the forward and backward reconstructions (see Figure B in S1 Material). For $0 < |L_{\text{ratio}}| \leq 5$ the average difference in the number of spacer deletions between forward and backward is 5.5 (median 2). *CRISPR-evOr* still provides an estimate for the correct orientation in these cases. However, since the number of underlying deletion events is low, we believe that these results should be treated with caution. For $|L_{\text{ratio}}| > 5$, the average number of differences is 140.3 (median 42) with almost all groups exceeding at least 5. This is a substantial event count considering that arrays in the dataset have a median length of approximately 15 spacers.

Combining both investigations leads us to believe that a conservative threshold of $c = 5$ allows a confident prediction of orientations.

6.3 Results

6.3.1 Orientation algorithm

We now describe the process with which *CRISPR-evOr* predicts the acquisition-orientation. Previous methods of estimating the strand orientation mostly rely on properties of the repeats and neighboring DNA sequence within a *single* genome. We, however, use spacer arrays of a *collection* of samples and leverage the polarized spacer acquisition process, to estimate the orientation. For a reliable acquisition-orientation prediction, overlap in spacers between samples in this group and enough spacer diversity are required. Before starting the reconstruction and prediction process, we identify and relabel unique spacers within the group with integers, as is done in *SpacerPlacer* (Chapter 5).

The tool *CRISPR-evOr* is provided within the *SpacerPlacer* framework at `https://github.com/fbaumdicker/SpacerPlacer`. We provide orientation predictions of all tools, accession numbers and some useful information for all groups considered here in S1 Table (csv) in S1 Material. The column headers of S1 Table are described in Note E in S1 Material.

The input into the algorithm is a group of spacer arrays. We show examples for such groups in the github repository and in Figures 6.1, 6.2 and Figures B, C in S1 Material. *SpacerPlacer* supports the input of array groups as fasta files of either the array DNA sequences of the spacers (repeats are removed) or arrays with already fixed spacer ids. The spacer DNA sequences can be extracted from CRISPRCasdb and CRISPRCasFinder or CRISPRidentify output. Scripts for the conversion and pre-processing for CRISPRCasFinder output are provided as options of *SpacerPlacer*.

If no tree was provided, the reconstruction is preceded by the estimation of a suitable phylogenetic tree based on the CRISPR arrays as described in Section 5.2.

Afterwards, *CRISPR-evOr* follows a three step strategy. First, we reconstruct the ancestral states of the provided spacer data, in the orientation it is provided in, with *SpacerPlacer*. The computed maximum likelihood L_{forward} is saved for later use.

In a second step, we *reverse* the spacer integer sequences (not the DNA sequence) of each sample. This is equivalent to reversing the orientation of *all* arrays in the dataset. We then estimate the ancestral reconstruction with *SpacerPlacer* in exactly the same way as in the first step. Again, we save the computed maximum likelihood L_{reverse}. Note that if the tree was estimated based on the pairwise distance approach from *SpacerPlacer*, the tree estimation is repeated as well, since the distances are highly dependent on the spacer order.

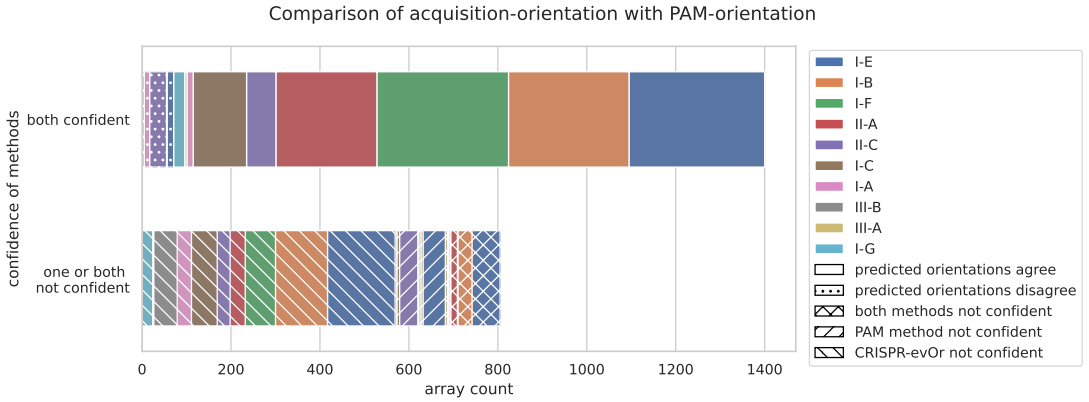
In the third step, we compare the maximum likelihoods computed for each orientation by taking the ratio of both likelihoods to arrive at an orientation prediction.

Likelihood comparison and measuring confidence

Since we use the same parameters for the reconstruction and either use a provided tree (which is independent of the spacer sequences) or, if no tree is provided, estimate the trees for each direction individually, we can compare the likelihoods in a meaningful likelihood ratio

$$L_{\text{ratio}} := \log L_{\text{forward}} - \log L_{\text{reverse}}.$$

To quantify the certainty of *CRISPR-evOr* we introduce a confidence threshold $c \geq 0$. However, since the likelihood is non-canonical, the asymptotic distribution of the likelihood



;

Figure 6.3: Comparison between acquisition- and PAM-orientation. We compare the predictions between the PAM-orientation and *CRISPR-evOr* broken down according to Cas type. Only subtypes with at least 5 groups with more than 50 arrays are shown. While there are many arrays were both methods can not make confident decisions, in cases where they are confident, they almost always agree. However, type II-C deviates from this behavior.

ratio is unknown. Therefore, we rely on empirical results in simulations and on available real data to determine a suitable confidence threshold c to make reliable decisions; see Methods (Section 6.2).

Computing the L_{ratio} results in one of two cases:

1. If $|L_{\text{ratio}}| > c$, one of the reconstructions can be confidently determined as the more likely according to the parametric model. Then, the orientation is determined by the sign of L_{ratio} and *CRISPR-evOr* returns "forward" or "reverse" accordingly.

2. If $|L_{\text{ratio}}| \leq c$, the likelihoods of the reconstructions are too close together and no confident prediction can be made. *CRISPR-evOr* returns "not determined".

A particularly common subgroup of uncertain orientation predictions is $|L_{\text{ratio}}| = 0$. Two such examples are shown in Figure B in S1 Material. In this case, the reconstructed events with forward orientation are exactly as likely as the reconstructed events with reverse orientation according to the parametric model. *SpacerPlacer* makes no difference between individual spacers (or arrays) but analyzes only the retention of spacers and the number and location of deletion events (Chapter 4 and 5). Thus, a symmetric group of arrays (not considering spacer labels) will result in the same likelihoods and return $L_{\text{ratio}} = 0$. Notably, this is necessarily the case if the first and last spacer in each sample are identical.

6.3.2 Evaluation of acquisition-orientation

In Methods (Section 6.2), we assess the performance of *CRISPR-evOr* in a simulation setting with three sets of simulated data generated as in Section 5.2.2. Note that since every discussed orientation tool relies on distinct data types, it is not feasible to compare and test the performance of all discussed tools in a fair simulation setting.

In all simulated settings, *CRISPR-evOr* achieves a strong performance with an accuracy greater than 95% (with a threshold of $c = 0$). Most of the wrong predictions can be avoided, instead being labeled as uncertain, by setting a suitable threshold (e.g. $c = 5$). However, due to the limitations of the simulations and the relative power of the algorithm,

the simulation setting cannot serve as a conservative estimator for a suitable choice of c and thus evaluation on empirical data is required.

In the CRISPRCasdb dataset, we found that many groups, e.g. the group shown in Figure 6.2, have high $|L_{\text{ratio}}|$ enabling *CRISPR-evOr* to make a confident decision for one orientation. However, for array groups where limited or erratic evolution results in lower likelihood ratios, decisions can be more difficult and not always obvious (Figure C in S1 Material).

Thus, we manually investigated the CRISPRCasdb data decisions with lower $|L_{\text{ratio}}|$ to determine a suitable conservative confidence threshold of $c = 5$, which indicates a difference of more than two orders of magnitude (e^5) between forward and reverse likelihoods. To further confirm the suitability of $c = 5$, we analyzed the difference in the number of deletion events between both reconstructions, as outlined in Methods (Section 6.2).

However, we recommend that users conduct their own experiments and manual investigations to ensure that the threshold is reliable for their particular use case and parameter settings.

We evaluate the performance of *CRISPR-evOr* by comparing our predictions to PAM-orientation predictions by Vink et al. (2021) and the leader- and repeat-orientation methods CRISPRDirection (Biswas et al. 2016) and CRISPRstrand (Alkhnbashi et al. 2014) which are part of CRISPRCasFinder (Couvin et al. 2018) and CRISPRidentify (Mitrofanov et al. 2021), respectively. The evaluation of PAM-orientation versus *CRISPR-evOr* and CRISPRDirection and CRISPRstrand versus *CRISPR-evOr* are performed on different datasets as reported in methods and in the respective figure captions.

Note that *CRISPR-evOr* considers groups of arrays, whereas the PAM-orientation and leader- and repeat-orientation methods consider single arrays. In all figures here, we show the orientation predictions for individual arrays. We show the results in terms of groups in Figures E and F in S1 Material. Moreover, we show pairwise comparisons between all tools in Figures J-O in S1 Material. For array based methods, we arrive at group orientation predictions by following a consensus decision approach, see Note C in S1 Material. In Figure P in S1 Material we show a comparison of the fraction of confident predictions within each group for all three array-based tools.

PAM- & acquisition-orientation are confident and agree for large parts of the dataset

The number of arrays with confident predictions far exceeds those with uncertain predictions. However, both tools have a significant number of arrays (36.5%) where at least one of the tools is not confident in its predictions (Figure 6.3). As can be seen, PAM-orientation is confident on more arrays (89.3%) compared to *CRISPR-evOr* (68.6%). This is primarily due to many array groups that exhibit minimal or no evolution (113 groups with 471 arrays, 68.1% of all uncertain array predictions by *CRISPR-evOr*), making it challenging to predict the acquisition-orientation.

Additionally, we show in Figure 6.3 that *CRISPR-evOr* and PAM-orientation are almost always in agreement for abundant types I-E, I-B, I-F and II-A. The only exception is type II-C, where PAM-orientation and *CRISPR-evOr* disagree in 21.3% of cases. However, the number of available groups and arrays of type II-C (178 arrays) is quite small. *CRISPR-evOr* is not confident for available type III-B groups as only one of them shows *any* evolution.

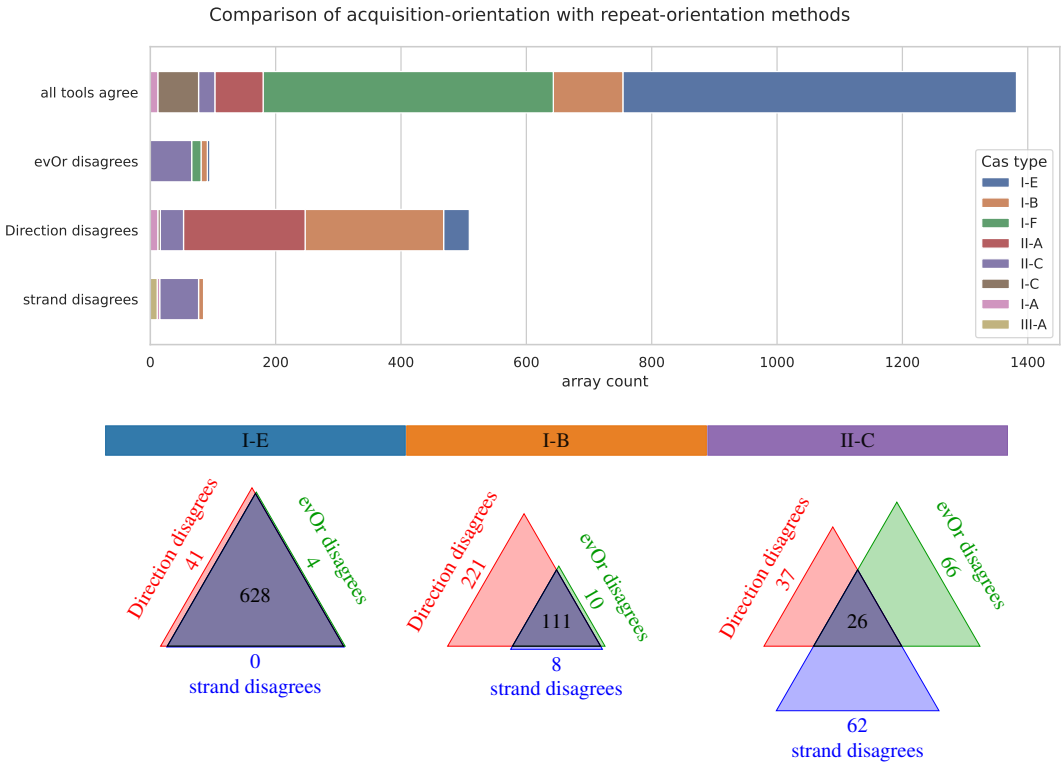


Figure 6.4: Comparison with repeat based orientation methods. We show the agreement of orientation predictions between CRISPRDirection, CRISPRstrand and *CRISPR-evOr* where all methods have high confidence broken down according to Cas type. Cas types were filtered as in Figure 6.3. At the bottom, we show Venn diagrams of the same data for three interesting subtypes with distinct behavior. Clearly, all methods agree quite often, especially for well-researched Cas types like I-E and I-F. However, there are substantial differences for predictions for type I-B, II-A and II-C. This indicates differences in the characteristics used for orientation prediction such as their repeats, leader and acquisition behavior. CRISPRDirection and CRISPRstrand were developed/trained using the more prevalent and well-studied Cas types. Consequently, they may struggle for specific uncommon Cas types. Venn diagramms for the remaining types can be found in Figure G in S1 Material.

Acquisition-orientation provides confident and reliable predictions compared to repeat based methods

When we applied *CRISPR-evOr* on the CRISPRCasdb dataset, we found that our method is confident for 63.7% of the arrays (55.4% of the groups), whereas CRISPRDirection and CRISPRstrand are confident on 65.3% and 83.5% of the arrays, respectively. While *CRISPR-evOr*, CRISPRDirection, and CRISPRstrand are confident on the same samples for 35.4% of the arrays (Figures H and I in S1 Material), there is a substantial amount of samples (28.3%) where either CRISPRDirection and/or CRISPRstrand are uncertain but *CRISPR-evOr* is confident. Thus, *CRISPR-evOr* allows for confident orientation predictions for many arrays where other methods are unreliable.

Among the groups where *CRISPR-evOr* is not confident (231 of 518 groups with 2152 of 5934 arrays), the number of groups/arrays with $L_{\text{ratio}} = 0$ far exceeds the number of groups/arrays with some amount of evolution but no confident prediction (48 groups, 671 arrays), i.e. $0 < |L_{\text{ratio}}| \leq c$. Across the CRISPRCasdb dataset, 35.3% (183) of the groups, consisting of 25.0% (1481) of the arrays, have a likelihood ratio of 0 as they carry little to no diversity in spacer content or exhibit strong symmetry in forward and reverse reconstructions. Currently, on the subset of groups with sufficient non-symmetric diversity to use *CRISPR-evOr*, i.e. all 4453 arrays (335 groups) with $|L_{\text{ratio}}| > 0$, *CRISPR-evOr* is able to predict the orientation of 85.0% (3782) of the arrays confidently (85.7% (287) of the groups).

To compare the contradictory predictions of the different tools, we performed a threefold comparison between *CRISPR-evOr* and CRISPRDirection and CRISPRstrand on the subset of our CRISPRCasdb dataset where *all* three tools have confident, but not necessarily identical, orientation predictions (Figure 6.4 and Figures F and G in S1 Material). This subset, amounts to 2071 arrays (excluding rare types), corresponds to the upper green column in Figures H and I in S1 Material.

We find that for common well-studied Cas types like I-E and I-F, all tools are typically in agreement. For other systems, especially for type I-B and II-A, CRISPRstrand's and CRISPRDirection's prediction are more frequently in conflict than in agreement. This is remarkable since CRISPRDirection and CRISPRstrand rely mostly on similar information, i.e. the repeat mutation structure, although CRISPRDirection uses leader-based criteria as well.

The independent evolutionary approach of *CRISPR-evOr* is able to resolve these conflicts. In general, our predictions are more aligned with CRISPRstrand's predictions than CRISPRDirection's predictions (Figure 6.4, Figures F and G in S1 Material).

The picture is different for Cas type II-C, where CRISPRDirection, and CRISPRstrand often disagree with each other, but *CRISPR-evOr* does not align consistently with either tool. Furthermore, *CRISPR-evOr* disagrees in many cases where CRISPRstrand and CRISPRDirection are in agreement.

6.4 Discussion

We demonstrate that the polarized insertion of spacers enables inference of the orientation of CRISPR arrays using evolutionary likelihood calculations. Compared to sequence-based methods, this is an independent approach, which not only allows to predict the orientation of previously unpredictable arrays with high confidence, but also to reject or confirm previous sequence-based CRISPR array orientation predictions.

Challenges in accurately predicting the array orientation. An advantage of our evolutionary approach in comparison with other approaches is that the detection of leader, repeat, and PAM sequences, which can be unreliable due to their inherent variability, does not influence our predictions. For example, some CRISPR-Cas systems do not have (a clearly identifiable) leader (Alkhnbashi et al. 2016). Orientation decisions based on repeats are often reliable, but also often lack confidence, particularly for short CRISPR arrays with few repeats (Couvin et al. 2018; Mitrofanov et al. 2021).

Similarly, the transcription orientation of the Cas genes and the array orientation frequently, but not always, align. Moreover, Cas genes are not always located near CRISPR arrays, and even when they are, CRISPR-Cas systems (e.g., types I-A and II-C2) may contain Cas genes transcribed in different directions (Makarova et al. 2020).

For PAM-based orientation predictions, efficiently and reliably identifying protospacers and PAMs remains a challenge and depends on the availability of appropriate metagenomic datasets. Nevertheless, in those cases where these challenges are manageable, we consider the orientation inferred from PAMs to be a reliable, particularly when the results are based on a multitude of spacers within an array. However, one has to consider that the location and properties of PAM sequences are subtype specific. For example, the PAM is known to be on the 5' end for type I, type IV and V, and on the 3' end for type II systems where the PAM is also strain- and not repeat-specific. Moreover, not all Cas types have (known) PAMs (Collias et al. 2021; Jackson et al. 2017; Vink et al. 2021).

An evolutionary approach for array orientation prediction. The acquisition-orientation considered here is a distinct concept independent of (but correlated with) the leader- and repeat-orientation concepts. Unlike all other tools discussed, *CRISPR-evOr* requires multiple CRISPR arrays with some amount of overlap and differences.

Naturally any evolutionary approach can not provide an orientation estimate for a single array or a set of identical arrays. In CRISPRCasdb, we found a substantial amount of closely related arrays which do not show observable evolution. We believe that besides CRISPR systems that lost their acquisition machinery, global sampling biases towards more closely related strains contribute to the current lack of diversity among these spacer arrays.

Ongoing sequencing and sampling efforts will extend the available datasets and generate larger more diverse groups of related arrays, and, in turn, allow *CRISPR-evOr* to make even more confident and reliable predictions in the future. However, *CRISPR-evOr* is already able to make confident predictions from groups containing only three arrays and limited (but sufficient) diversity.

Our approach to predict orientation based on the diversity of CRISPR arrays is not limited to the concrete likelihood calculation based on the evolutionary block deletion model in *SpacerPlacer*. In principle, any evolutionary model, such as the most parsimony approach implemented in the CRISPR Comparison Toolkit (Collins et al. 2023) could be used to reconstruct ancestral arrays for both orientations and identify the more parsimonious or likely direction.

Variations in the acquisition and deletion mechanisms between different Cas types, particularly those not captured by the model assumptions, could affect any evolutionary reconstruction approach. While the deletion mechanisms are rather universal across different subtypes (Section 3.2, Section 5.3), some Cas subtypes do not always strictly follow polarized insertions. In some studies ectopic insertions within the array have been observed for I-A (Erdmann et al. 2012; Garrett et al. 2015) and II-A (Achigar et al. 2017; Achigar

et al. 2021; McGinn et al. 2016; Mosterd et al. 2020). Nevertheless, polarized insertions appear to be the predominant mode of spacer acquisition (Section 3.1). In all observed cases of ectopic insertions, they occur less frequently than polarized insertions and only when specific regions of the leader are mutated or lost (Erdmann et al. 2012; Garrett et al. 2015; McGinn et al. 2016). *CRISPR-evOr* performs well on available II-A arrays, which supports results showing that type II-A acquisition is typically polarized (McGinn et al. 2016).

Indeed, *CRISPR-evOr* is more decisive on simulated data, based on a simple evolutionary model, than real data, suggesting that CRISPR arrays are subject to more complex evolutionary processes. This not only includes uncommon events like ectopic insertions, duplicate events, and array collapse, but also sequencing errors, selection pressures, and changes in the environment which can all lead to unexpected patterns. However, spacer duplications and other complex evolutionary events are classified as such by *SpacerPlacer* and thus do not directly impair the orientation prediction of *CRISPR-evOr*.

In conclusion, although there are various issues that can impede accurate prediction of acquisition orientation, there are three essential requirements for reliable predictions by *CRISPR-evOr* that can be summarized as: (i) a sufficiently diverse set of different, but related, CRISPR arrays with (ii) active evolution and (iii) polarized acquisition of new spacers. The strong performance in simulations in combination with the agreement with sequence based prediction methods for most CRISPR system types underscores the power of our evolutionary approach.

Comparison of orientation predictions across Cas types. We find that the orientation predictions of the abundant Cas types I-E and I-F show a strong consensus among all investigated orientation concepts. The consensus is to be expected as these abundant types are well-researched and served as blueprint for typical CRISPR behavior in terms of evolution, expression, transcription and repeat and leader characterization.

CRISPRDirection, CRISPRstrand and CRISPRleader rely on leader and repeat attributes that were defined based on these most abundant types, which are also a substantial part of the training and test datasets for CRISPRstrand and CRISPRdirection. Similarly, *CRISPR-evOr* is reliant on polarized insertion, which is a behavior that characterizes (almost) all types of CRISPR arrays but was identified and experimentally confirmed particularly for type I.

Interestingly, for types I-B and II-A, *CRISPR-evOr*, PAM-orientation and CRISPRstrand agree in their prediction for almost all arrays, while CRISPRDirection disagrees with them. Thus, for these types, CRISPRstrand is likely more reliable than CRISPRDirection. Since CRISPRDirection's predictions are contradicted by three independent sources of evidence, the PAM, repeat structure, and evolutionary likelihood, we conclude that the corresponding arrays are likely misclassified by CRISPRDirection. This is in line with the observation that CRISPRDirection is confident for fewer arrays than CRISPRstrand (Figures H and I in S1 Material). While wrongly predicted orientations by CRISPRDirection and consequently in CRISPRCasdb have already been noted for some strains and systems (Alkhnbashi et al. 2014; Prakash et al. 2022). Our results indicate that – at least among the considered dataset of arrays, with confident predictions by CRISPRDirection, CRISPRstrand and *CRISPR-evOr* – the orientations of 63.1% of type I-B and 71.6% of II-A arrays on CRISPRCasdb are incorrect.

Correctly inferred orientations are crucial for a multitude of subsequent analyses, e.g. regarding array transcription, or protospacer identification (Blombach et al. 2024; Shmakov

et al. 2017; Stringer et al. 2020; Zhang et al. 2022). Thus, for the afore mentioned and related Cas types we advise reassessing the corresponding data and experiments to verify if the conclusions remain valid when the orientation of many arrays has to be reversed.

Among the three tools, the greatest divergence in predicting the array orientation arises with type II-C (Figures 6.3 and 6.4, Figure G in S1 Material). Type II-C has been shown to be particularly challenging to orient due to its unusual features in spacer acquisition, expression, leader, repeat structure (Dugar et al. 2018; Hoikkala et al. 2021; Mitrofanov et al. 2026; Zhang et al. 2013), and PAM diversity (Vink et al. 2021). In particular, multiple studies reported that II-C arrays favor acquisition at the downstream 3'-end of the array (Hoikkala et al. 2021; Laanto et al. 2017; Zhang et al. 2013). Moreover, type II-C has not only leader promoters, but promoters in the repeats (Dugar et al. 2018; Zhang et al. 2013), which lowers the selective pressure to insert new spacers at the leader end and may cause deviations from patterns typically observed in leader and repeats of other CRISPR systems.

However, as long as the (majority of) acquisitions are polarized and the acquisition end is consistent *within individual groups*, the predictions of *CRISPR-evOr* likely remain accurate and consistent. Indeed, manual inspection of the reconstructions of the groups of II-C arrays with confident predictions showed normal (polarized) evolutionary behavior and thus sensible acquisition-orientation predictions. Thus, the inconsistency of acquisition-orientation with both methods to predict repeat-orientation suggests that type II-C arrays might be composed of subgroups with different acquisition ends and thus inconsistencies within leader and repeat structures. This could cause the complex discrepancies between acquisition-orientation and repeat-orientation.

Multiple subtypes of II-C have been proposed in Makarova et al. (2020) based on the composition of their Cas loci. Our results in Chapter 7 point into the same direction, where we found that repeat diversity clusters II-C in multiple groups and could be related to co-occurrence of other Cas types. In any case, the inability of the tools to determine consistent orientations for type II-C warrants further investigation of the diversity and functionality of II-C systems.

Conclusions and future perspectives. With *CRISPR-evOr* we provide a reliable tool to predict the acquisition-orientation of CRISPR arrays. Our results suggest to treat the orientations of CRISPR arrays from CRISPRCasdb, at least for some systems, with caution.

Among the different orientation concepts, acquisition-orientation is most directly related to the adaptive evolutionary dynamics of spacer arrays and thus carries an additional ecological importance. The acquisition-orientation is crucial to understand the interaction between the environment and the CRISPR-Cas system. This includes especially the acquisition and deletion mechanisms, and their defensive and/or regulatory function in bacteria and archaea. In particular, to assess the relevance and protective capabilities of specific spacers and the ecological importance of their targets, it is essential to know whether spacers have been acquired recently or in the more distant past (López-Beltrán et al. 2024).

As the availability of CRISPR arrays continues to grow, especially with the increasing number of long-read sequencing projects, *CRISPR-evOr* will allow to investigate CRISPR arrays for various CRISPR-Cas types in the future, including type III CRISPR arrays that have been found to be distinct in their spacer acquisition (Achigar et al. 2021; Zhang et al. 2022).

7

The complexity of multiple CRISPR arrays in strains with (co-occurring) CRISPR systems

Chapter with co-authors
This chapter is based on Fehrenbach et al. (2026), with minor modifications, and accordingly contains contributions by the co-authors. Note that references to supplementary notes and figures refer to the supplementary material of Fehrenbach et al. (2026).

Chapter summary (Abstract)
CRISPR and their associated Cas proteins provide adaptive immunity in prokaryotes, protecting against invading genetic elements. These systems are categorized into types and are highly diverse. Genomes often harbor multiple CRISPR arrays varying in length and distance from Cas loci. However, the ecological roles of multiple CRISPR arrays and their interactions with multiple Cas loci remain poorly understood.
We present a comprehensive analysis of CRISPR systems that uncovers variation between diverse Cas types regarding the occurrence of multiple arrays, the distribution of their lengths and positions relative to Cas loci, and the diversity of their repeat sequences. Some types tend to occur as the sole Cas locus present in the genome, but typically have two or more associated arrays, especially for types I-E and I-F. Multiple Cas types are also common, with some systems showing a preference for specific co-occurrence. Distinct array distributions and orientations around Cas loci indicate substantial differences in functionality and transcriptional behavior among Cas types.
Our analysis suggests that arrays with identical repeats in the same genome acquire new spacers at comparable rates, irrespective of their proximity to the Cas locus. Furthermore, repeat similarities indicate that arrays of systems that often co-occur with other systems tend to have more diverse repeats than those mostly appearing alongside solitary systems. Our results indicate that co-occurring Cas type pairs might not only collaborate in spacer acquisition but also maintain independent and complementary functions and that CRISPR systems distribute their defensive spacer repertoire equally across multiple CRISPR arrays.

7.1 Introduction

The distribution of CRISPR systems and, in particular, the variability of their CRISPR arrays offer insights into the defense strategies and evolutionary pressures exerted by phages and mobile genetic elements (Fehrenbach et al. 2024; López-Beltrán et al. 2024).

Prokaryotic genomes encode a remarkable diversity of CRISPR systems (Chapter 2), which differ in their molecular components, organization, and functions (Makarova et al. 2020; Makarova et al. 2025). These systems are widespread and can be located on both chromosomes and plasmids (Pinilla-Redondo et al. 2022), yet their distribution across bacterial genomes is patchy, suggesting associated fitness costs (Section 2.3.2). Multiple CRISPR system types can coexist within a single strain (Vink et al. 2021; Weissman et al. 2018b), enabling interactions between systems and their arrays, such as spacer acquisition in trans and the sharing of CRISPR arrays and crRNAs, as discussed in Section 2.3.4.

Beyond the diversity of Cas types and proteins, CRISPR arrays themselves exhibit substantial variability (Chapter 5). Their composition can provide insights into the evolutionary arms race between bacteria and phages, as spacer position and array length influence transcription frequency and immune effectiveness (Deecker et al. 2020; Liao et al. 2022; López-Beltrán et al. 2024; McGinn et al. 2016). Arrays differ in genomic location (near Cas loci or dispersed), length, and the uniformity of their leaders and repeats. Repeat sequences are typically type-specific (Alkhnbashi et al. 2016; Vink et al. 2021), while leader sequences play a key role in regulating spacer integration and expression (Chapter 2 and 3.1) and often contain both type- and taxonomy-specific motifs (Alkhnbashi et al. 2016; Buyukyoruk et al. 2025). Overall, this variability suggests that not only repeats and leaders, but also the structure and genomic positioning of CRISPR arrays, are important determinants of system functionality and adaptability.

When multiple CRISPR arrays are present within a bacterial genome, it remains unclear whether, and if so which, arrays are preferentially used for new spacer acquisitions. One key factor may be their genomic proximity to Cas genes, as closer arrays might be more actively transcribed, while distant arrays could serve as archival records. Additionally, the presence of multiple arrays raises the question of whether they serve distinct functional roles or provide redundancy. While previous studies and simulations have explored spacer acquisition dynamics and array length distributions (Martynov et al. 2017; Vink et al. 2021; Weissman et al. 2018b), a large-scale comparative analysis is still missing. Similarly, although the interaction between multiple CRISPR-Cas systems has been studied (Bernheim et al. 2019), it is not yet fully understood. If different systems coexist, do their arrays exhibit more similar repeats, potentially allowing for the cross-utilization of Cas proteins, are certain systems incompatible with each other, or do they function independently? Understanding these relationships can shed light on the evolutionary pressures shaping the diversity of CRISPR arrays and systems.

Here, we systematically explore a database with thousands of prokaryotic genomes, their CRISPR systems and CRISPR arrays. We utilize the relationship between mean spacer acquisition rates and array lengths to investigate the relative activity of multiple CRISPR arrays, the array position relative to Cas genes and the repeat sequence influences spacer acquisition. We also consider how multiple CRISPR systems within a genome interact, and find that all these patterns differ substantially between Cas types. We hope that the comprehensive analysis here provides the basis for a better understanding of these differences and can shed light on why certain array configurations are favored and how they impact bacterial immunity.

7.2 Methods

7.2.1 Dataset

We used a comprehensive dataset of CRISPR arrays obtained by extracting available whole genomes from GenBank (in November 2024) and running CRISPRidentify (Mitrofanov et al. 2021) to detect all high confidence (labeled by CRISPRidentify as "bona-fide" and "possible") CRISPR arrays within each genome.

Cas genes were identified and labeled by CRISPRCasidentifier (Padilha et al. 2020). Cas types are assigned according to the closest Cas locus, if no Cas was found within the genome they are assigned "no cas type" and called "orphans". These orphan arrays have been excluded for all major analyses and figures but appear in some supplemental figures and tables, as indicated. Arrays are labeled as "close" to their Cas, if they are less than 10,000 bp from their respective Cas locus. Array orientations were determined using CRISPRstrand (Alkhnbashi et al. 2014).

We obtained a total of 47789 genomes with 82981 CRISPR arrays, 44760 distinct Cas loci and 19946 unique consensus repeats; see Supplementary Tables S1–S4 and Supplementary Fig. S1–S5 for additional statistics.

7.2.2 Cas type co-occurrence analysis

We also considered the co-occurrence of Cas types relative to their total abundance. For each possible combination of Cas types, we compared the observed and expected numbers, where the expected number is given by the mean number of such genomes when the presence and absence of Cas types would be randomly and independently distributed across the dataset according to the observed frequency.

We did this (i) for the observed and expected number of times a system occurs as the only system within a genome, and (ii) the observed and expected number of times a pair of CRISPR systems co-occurs.

For example, if for a total of n genomes, one Cas type is present in j genomes and another type is present in k, not necessarily distinct, genomes, the expected number of genomes where both types co-occur would be jkn^{-2}.

The resulting ratios of observed and expected numbers are transformed into a fold-change and are depicted in Figure 7.1. The corresponding data and p-values are given in Supplementary Tables S5 and S6. A positive fold-change of x indicates that the observed number is x-times more frequent than expected, whereas a negative fold-change of $-x$ indicates that the observed number is x-fold smaller than expected.

7.2.3 Pairwise distances between consensus repeats

We compute any distances between repeats solely for the respective consensus repeats of the arrays. For any two consensus repeats we compute the Levenshtein distance between the two. These distances are normalized by the maximum length of consensus repeats within the genome for Supplementary Fig. S13 or globally for Figure 7.4.

7.2.4 Visualization of repeat distances with multidimensional scaling (MDS)

We computed the pairwise Levenshtein distances for all consensus repeats for all arrays in the dataset as described above. Then, we performed metric MDS, using the obtained distance matrix as dissimilarity measure, to obtain a projection into two-dimensional space using the MDS function of the python package scikit-learn (v. 1.3.2). This MDS function uses the metric SMACOF algorithm and minimizes the stress (residual sum of squares of the original distances and the distances in the 2-d MDS space).

The projection of each array's consensus repeat into the MDS space is shown in Figures 7.4 and 7.5. All subfigures show subsets of arrays projected using the *same* MDS projection for greater visual clarity.

In addition to the scatterplots of the repeats projected into the two-dimensional MDS space, Figure 7.5 shows kernel density estimates with Gaussian kernels of the marginal distributions for both MDS coordinates and the indicated (colored) subsets. To compute the kernel density estimates we use the kdeplot function of the python package seaborn (v. 0.12.2) with default parameters, which relies on SciPy (v. 1.9.1) for Gaussian kernel density estimation.

7.2.5 Estimation of comparative insertion rate weights

To compute the insertion rate weights (Figure 7.3) we assume an independent deletion model (IDM) as described in Chapter 4 which evolves by independent acquisitions and deletions of spacers with insertion rate θ and deletion rate ρ. In the IDM the array length is Poisson distributed with mean θ/ρ and thus the average length n is given by θ/ρ. In Chapter 5 we found that deletion rates are independent of Cas type and likely caused by processes, e.g. replication misalignment, which should be consistent for different arrays within a single genome (Section 3.2). Thus, we assume the deletion rate to be constant and the insertion rate is likely the main driver of differing array lengths of arrays within the same genome.

Therefore, for a genome with k arrays we can roughly estimate the insertion rate of array $i \in \{1, \ldots, k\}$ as $\theta_i = n_i\rho$, where n_i is the length of the array. Furthermore, we define the total insertion rate for all arrays in the genome as $\theta = \theta_1 + \ldots + \theta_k$ for arrays $i \in \{1, \ldots, k\}$ within one genome with k spacer arrays. To compare the acquisition behavior of CRISPR arrays within the same genome we compute what we call (relative) *insertion rate weights* given by $w_i = \theta_i/\theta$. Clearly, by definition of θ and θ_i, and due to the constant deletion rate, w_i is solely determined by the proportions of array lengths, i.e. we find $w_i = n_i/n$, where $n = n_1 + \ldots + n_k$ is the combined total length of all arrays.

To obtain the expected neutral distributions of insertion rate weights in Figure 7.3 we proceeded as follows. In a neutral setting, where all arrays are equally likely to acquire new spacers, for a genome with k arrays, any new spacer is inserted into any of the arrays with equal probability $(1/k)$. Thus, given a total of n spacers, the array lengths of the k arrays would be distributed according to a multinomial distribution with n number of trials, k distinct events and equal probabilities $p_i = 1/k$ for $i \in \{1, \ldots k\}$.

We therefore sampled using the following approach: for each datapoint, i.e. genome, with k arrays and the total number of spacers n we collected a single sample from the corresponding multinomial distribution divided by n to obtain the insertion rate weights. The resulting distribution thus has the same total number of samples as the real data analyzed.

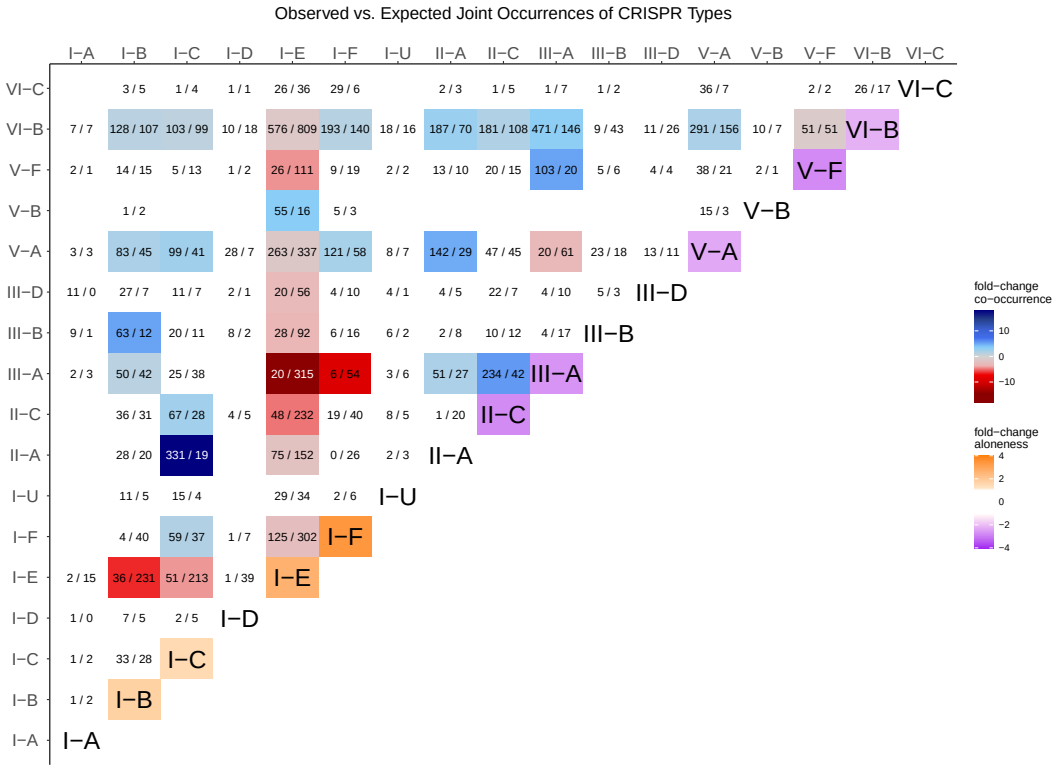


Figure 7.1: Single and co-occurring Cas types. For each pair of different Cas types we show the actually observed and the expected (if their occurrence would be reshuffled) number of joint occurrences (in the format observed/expected). For system-pairs with at least 50 observed or expected genomes, the color indicates the fold-change between the observed and the expected number. Specifically, a -5 fold-change is shown in red and indicates that the number of observed co-occurrences is 5-fold lower than expected if CRISPR systems were reshuffled, while blue colors indicate more frequent co-occurrence. On the diagonal the color indicates a different fold-change. Namely, the tendency for each type to occur as the only CRISPR system (positive fold-change, orange) or to occur in combination with at least one other system (negative fold-change, purple). For example types I-E and I-F are alone 2.6-fold and 3.4-fold more often than expected and II-C occurs alone 1.7-fold less often than expected. On the diagonal only systems with at least 50 expected or observed solo ocurrences and with a significant fold-change (bionomial test p-value $<$ 0.0001) are colored.

7.3 Results & Discussion

7.3.1 Diversity and co-occurrence of Cas types

We found that although the distribution of Cas types is dominated by a few abundant types such as I-E, I-F, III-A, V-A, VI-B and a substantial amount of orphan arrays (where no Cas locus was found in the genome), there is substantial diversity to explore (see Supplementary Tables S1–S3 and Supplementary Fig. S1, S2).

Moreover, although most genomes carry less than 7 distinct Cas loci with an associated CRISPR array (Supplementary Fig. S3) of less than 4 distinct types (Supplementary Fig. S4), a substantial number of genomes carry at least two different Cas types. Notably, certain combinations of Cas types are much more common than others, suggesting a potential interaction or synergy between some types (Wu et al. 2024). For example, based on the considered dataset this might be the case for II-A and I-C, as well as II-C and III-A,

and also III-A and V-F (see Figure 7.1) as those systems often occur together.

On the other hand, some systems, including I-E and I-F, tend to occur as the only CRISPR system more often than expected, which could indicate that these systems either provide already sufficient defense or that the presence of other systems interferes with their function.

In this context, it is also important to consider horizontal gene transfer (HGT) as one major vector in the propagation of CRISPR systems. Often, CRISPR systems are transferred via plasmids. CRISPR systems have been identified on (larger) plasmids, most commonly belonging to types I and IV. In particular, type IV systems, which are represented by only a few instances in our dataset, are predominantly plasmid-encoded (Pinilla-Redondo et al. 2020; Pinilla-Redondo et al. 2022). Thus, HGT, indirectly, plays a role in shaping the distribution of CRISPR systems and their co-occurrence. However, since we are considering a comprehensive, cross-sectional dataset, we expect our findings to be robust against noise introduced by HGT.

Moreover, it is believed that HGT is (often) inhibited by already existing CRISPR systems in the receiving genome. Bernheim et al. (2019) propose that systems that are incompatible with a functioning system in the cell would lead to the rapid deletion of one of the systems. For example, in *Escherichia coli* most strains contain a functional I-E system and at least one type I-F array without Cas genes where the majority of spacers within the I-F array target absent type I-F Cas genes (Almendros et al. 2016; Touchon et al. 2010). Bernheim et al. (2019) argue that this leads to immediate degration of incoming I-F Cas genes and thus to the inability of horizontal transfer of functional type I-F systems into *E. coli*. Plasmid-born arrays have been found to frequently carry spacers that target (other) plasmids (Pinilla-Redondo et al. 2022). Furthermore, a recent study (Liu et al. 2025) showed that indeed, the exchange of mobile genetic elements, and thus HGT, is significantly inhibited by CRISPR systems. However, they found this effect to be highly variable between species.

While other studies considered the co-occurrence of Cas types (Bernheim et al. 2019; Vink et al. 2021; Weissman et al. 2018b), here we also consider the number and length of the associated spacer arrays and the similarity of repeat sequences to obtain a more comprehensive perspective.

7.3.2 Cas loci often support multiple arrays

Clearly, the number of arrays tends to exceed the number of Cas loci, indicating that many Cas loci can and do serve multiple arrays (Supplementary Fig. S5). Although many Cas loci have only one array, we found that many Cas loci have multiple close CRISPR arrays, as there is an overall average of 1.48 arrays per Cas locus within the dataset (Supplementary Table S1 and Supplementary Fig. S6).

Larger amounts of around 5 arrays do occur quite often and larger numbers (up to around 10) are rare, but do appear (with a maximum of 23 for one V-A locus, see Supplementary Table S1). Types I-E, I-F and III-A are particular outliers since they have a median of two arrays which we will discuss further below.

On the other hand, Type II-A strongly favors only a single array (average 1.07). Similar observations have been made before based on the signature Cas genes for type I and II (Weissman et al. 2018b).

Our analysis shows that all Cas types have at least some Cas loci with multiple arrays. These arrays likely originate from duplication (of fragments of an array), multiple acquisitions of arrays (via HGT) and splitting of arrays into multiples due to mutations or insertions.

Naturally, if they remain functional, they will subsequently diverge in spacer content over time.

Duplications of arrays could be a major contributor for the frequency of same consensus repeat arrays. Recently duplicated arrays would have the same consensus repeat and likely share large subsets, or possibly all, of the spacers. Although CRISPR arrays diverge in their spacer composition relatively quickly, the last spacer at the leader distal end is the least likely to be deleted and would therefore be more likely shared by arrays with somewhat distant ancestry (Section 5.3, Figure 5.8).

To get a sense of the frequency of recent duplications, we compared the repeats and spacer content of all arrays and Cas loci and report the statistics in Supplementary Tables S1 and S2.

We found that similar consensus repeats are frequent. Of all arrays, 25.0% co-occur in a genome with at least one additional array with identical consensus repeat and 19.3% of the Cas loci have multiple associated arrays with the same consensus repeats. A substantial part of those arrays (11.9%) have at least some spacer overlap. However, only relatively few arrays (1.2%) have identical spacers. Thus, recent complete duplication or complete absence of spacer evolution is rare in our data set. A higher percentage (4.1%) have the same last spacer and thus indicate common ancestry. These values vary significantly between Cas types (Supplementary Table S2).

In general, the relatively high frequency of multiple CRISPR arrays suggests that, at least for some types, there is a beneficial fitness effect (Bernheim et al. 2019), such as robustness due to redundancy or specialization of arrays on specific targets (Bernheim et al. 2019; Weissman et al. 2018b). However, definitive empirical evidence for these theories is currently lacking.

7.3.3 CRISPR arrays are differently positioned and oriented for different Cas types

Most arrays tend to be close (< 10,000 bp) to their respective Cas locus (Supplementary Fig. S2 and Table S4). However, there are huge differences between Cas types. Types I, II, and III tend to favor closeness with few exceptions, but arrays of type V and VI are almost always located farther away (> 500,000 bp) from their Cas locus (Supplementary Fig. S7).

We analyze the genomic position of CRISPR arrays relative to their corresponding Cas locus by computing the distance between the closest ends of the array and Cas locus. Thereby we determine if they are positioned "before" or "after" the Cas. Moreover, we determined the orientation of the arrays. We show both the distance and orientation (for arrays with close (< 10,000 bp distance) Cas) for a subselection of interesting Cas types in Figure 7.2. Distributions for all types with sufficient data are shown in Supplementary Fig. S8.

Closeness to the Cas locus tends to be favored for all types, and most arrays are less than 500 bp distant from a Cas locus. However, CRISPR types also show very different distributions of distances and distributions of orientations. Moreover, there is a tendency for all types, that arrays close to Cas are forward oriented, whereas types further away from the Cas are reverse oriented.

We grouped the different subtypes, with respect to their distance and orientation distributions, into multiple groups. For each group we show particularly interesting and representative types in Figure 7.2; marked **bold** in the text.

1. Type I-C, **I-E**, II-A, and IV-A tend to have arrays *after* and close to the Cas locus

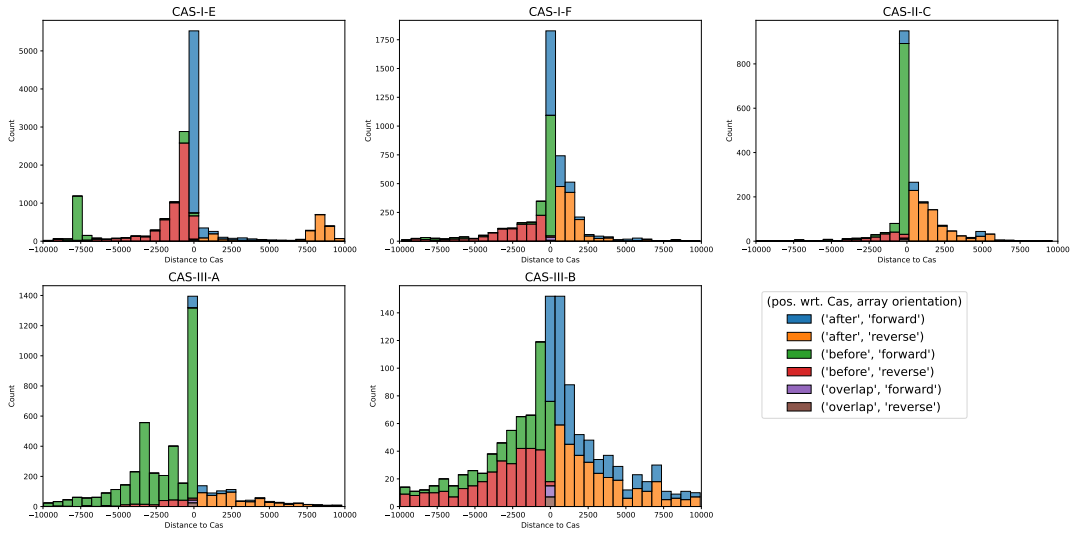


Figure 7.2: Distance to Cas and array orientation. We show the distance of CRISPR arrays to their respective Cas locus. The distances and labels 'before' and 'after' were obtained by comparing the positions of the arrays and Cas genes on the genome, i.e. arrays are positioned 'before'/'after' their Cas cassette and assigned negative/positive "Distance to Cas", respectively. A small number of arrays are contained within their respective Cas locus and were labeled as 'overlap' (with distance 0). Additionally, we indicate the array orientation as predicted by CRISPRstrand. As expected, all types favor close CRISPR arrays, however the different Cas types show large differences in the distribution of distances and orientations with particular biases. In general, arrays close to the Cas locus tend to be forward oriented and more distantly located arrays tend to be reverse oriented. We identify multiple classes of distributions and show representative types for these behaviors here. Distributions for all Cas types with sufficient data can be found in Supplementary Fig. S8. In particular, type I-E has a bias for close arrays to be forward oriented and to be positioned after the Cas locus. Type I-E places close arrays on both sides of the Cas locus. I-E and II-A tend to favor to place the array before the Cas locus. However, II-A shows a significantly wider distribution. Lastly, type III-B shows significantly higher variance and places less importance on closeness to the Cas genes.

(in forward orientation), and arrays *before* the Cas tend to be more distant (in reverse orientation).

2. Type I-A, I-B, I-D, and **I-F** prefer close arrays *both before and after* the Cas locus (in forward orientation). More remote arrays are typically found *after* the Cas (in reverse orientation).

3. Type II-B, **II-C**, **III-A** tend to favor arrays arrays *before* and close to the Cas (in forward orientation). The more distant arrays are usually *after* the Cas (in reversed orientation). For III-A there are a substantial number of arrays more than 2,500 bp away from the Cas locus.

4. Type **III-B** and III-D have a behavior similar to 2. However, they have distributions with much higher variance and less favoritism for specific orientations. This resembles a distribution one might expect in a setting with lower selection pressures that act on the orientation and the distance of CRISPR arrays to the associated Cas locus.

5. Although types V-A and VI-B are very abundant, very few of their arrays are close to their Cas. Where they are close, they show (similarly to 4) more random distributions (see Supplementary Fig. S8).

We further investigated the orientation and position of pairs of CRISPR arrays in genomes with only one Cas type and exactly two CRISPR arrays (Supplementary Fig. S9 and S10). Remarkably, we found that types I-D, I-E and I-F with two arrays almost exclusively position their arrays one on each side of the Cas locus.

Possessing two arrays (or more) at the respective ends of the Cas locus is likely beneficial for various reasons. Multiple arrays provide redundancy, and their proximity to the Cas locus maintains high efficiency and enables joint regulation. Moreover, two arrays are likely to have more evenly distributed spacer expression compared to a single longer array, since spacers near their leader positions provide higher levels of immunity (Martynov et al. 2017; Weissman et al. 2018b).

Notably, I-E and I-F are very different in the distribution of array orientation. For I-E, the pairs of arrays are almost exclusively both in forward or both in reverse orientation, whereas for type I-F, the distribution of forward and reverse orientation of the two arrays seems to be independent of each other, regardless of whether the arrays are on the same or different sides of the Cas locus (Supplementary Fig. S10). I-D favors different orientations for pairs of arrays but possesses a substantial number of array pairs with the same orientation (Supplementary Fig. S9). III-A behaves differently and typically places both arrays before the Cas and in the same (forward) orientation. The peaks around -2,500 to -5,000 in the type III-A distribution (Figure 7.2) may indicate the presence of a second array next to another array located close to the Cas locus.

Connecting the fact that I-E tends to have arrays in the same orientation with Figure 7.2, noting that most Cas loci of type I-E have one or two arrays, indicates that single arrays close to the Cas are most often after the Cas and forward oriented or before the Cas and reverse oriented. However, for array pairs, one of the arrays is placed close before the Cas and reverse oriented, or, after the Cas and forward oriented. Quite frequently, the second array is much farther away and on the other side of the Cas (around 7,500 bp) in the same orientation. In terms of colors in Figure 7.2: blue and green tend to form (forward oriented) pairs and red and orange tend to form (reverse oriented) pairs.

Our findings suggest that the distribution of array orientations are consistent within most types, with notable exceptions such as III-B. However, there is high variance in organisation

between types, suggesting that the regulation of the transcription of Cas and arrays evolved to the specific needs of Cas types and host organisms (Milicevic et al. 2019).

Some systems, e.g. I-F, show heterogeneity in orientations even for arrays belonging to the same Cas locus (Supplementary Fig. S10 and Milicevic et al. (2019)), which poses questions on how these systems with multiple closely located arrays are regulated.

7.3.4 Arrays vary in length across Cas types and are typically longer when located near Cas genes

The spacer array length tends to depend heavily on the Cas type, although most arrays have less than 50 spacers (average of 15.11, median of 9) and there is a small number of arrays with between 50 and 100 spacers (Supplementary Table S3 and Supplementary Fig. S11 and S12).

In particular, types I-A and I-D have remarkably large arrays, with mean sizes of 51 and 42 spacers, respectively. Extremely large arrays are rare, however, with maxima of 246 and 317 spacers for types I-E and I-F, respectively, and up to 587 spacers for both types I-B and I-U. Types I (excluding I-A/I-D) and type III tend to have averages between 15 and 30, while type II has slightly smaller arrays with averages between 12 (II-C) and 18 (II-B) and types V and VI have far smaller arrays with averages and medians of 10 or lower.

If the hypothesis that spacer loss rates are comparable across different Cas types and species holds (Chapter 5), this could indicate that type V and VI CRISPR systems are less often triggered to acquire new spacers, compared to type I, II, and III.

Splitting the set of all arrays regardless of the repeat sequence into two groups depending on how close they are to the Cas and redoing the statistics yields additional insight (Supplementary Table S4). We found that for most types, arrays close to their Cas are much longer than arrays further away. In particular, type II-C has a large difference between medians 12 (close) and 5 (not close). This holds for most types except I-A, III-B, III-D, V-F, VI-A and VI-C. Thus, one might be tempted to assume that distance to the Cas locus has a large impact on the spacer acquisition rate.

However, this distribution could also be the result of other effects. For example, leaderless arrays have been shown to tend to be shorter (Alkhnbashi et al. 2016) and the predicted arrays that are further away could be more often leaderless. It may also be related to the absence of acquisition machinery (Cas1 and Cas2) in (nearby) Cas loci. In addition, the data likely also includes arrays with distinct repeats as well as misclassified or CRISPR-like elements, which could cause the shift towards shorter arrays. In line with this argument, we find that arrays that are not close to a Cas locus tend to have diverse repeat sequences that differ from the repeats in arrays that are close (Supplementary Figure S15).

7.3.5 Relative spacer acquisition rates for multiple arrays

We further investigated the insertion behavior by comparing the array lengths of arrays within the same genome. In contrast to the global array length distribution, when looking at the lengths of multiple arrays within the same genome, the deletion rate of spacers should be consistent across different arrays. This allows us to compare array lengths as a proxy for spacer acquisition rates within cells; see Methods (Section 7.2) and Chapter 5.

We call these comparative weights *insertion rate weights* and show their distribution in Figure 7.3. We restricted our analysis to genomes with exactly 2, 3 or 4 arrays to reduce other biases.

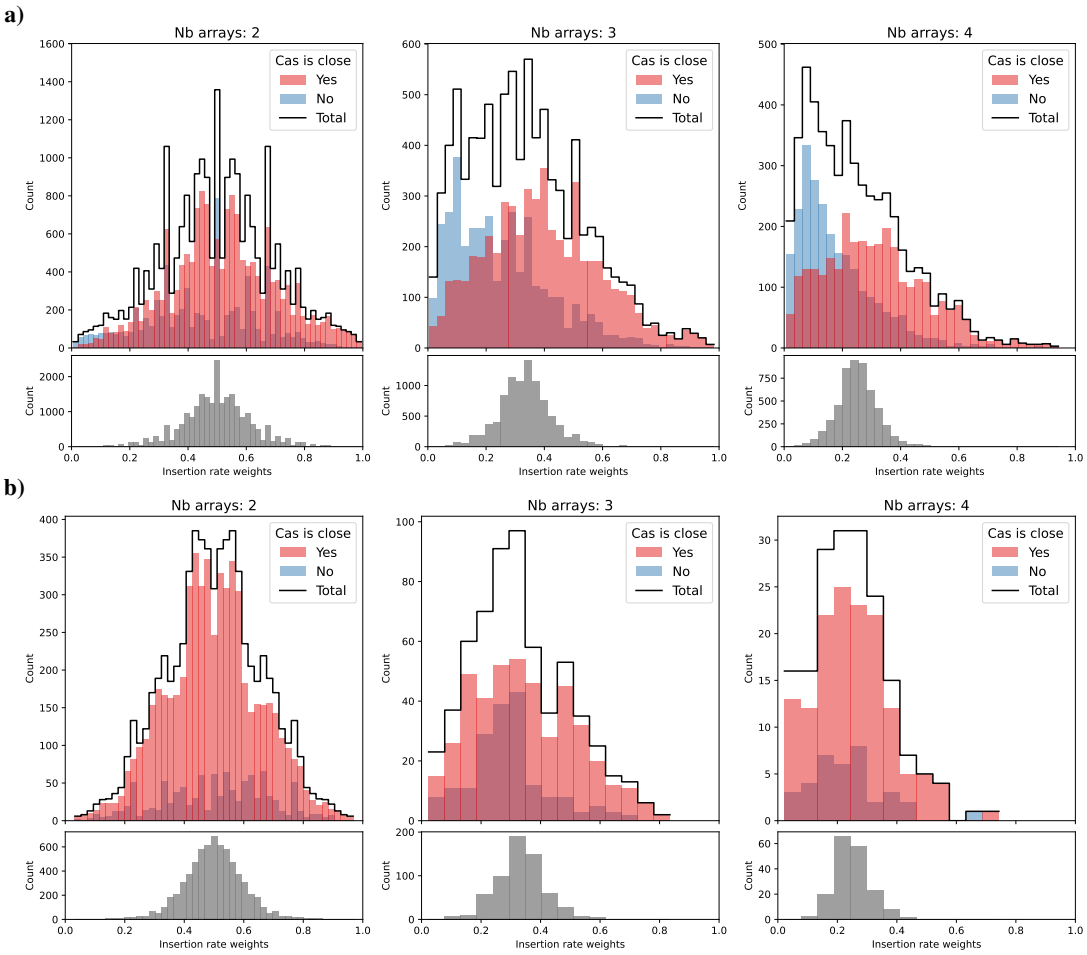


Figure 7.3: Distribution of the estimated insertion rate weights of arrays. At the top of each panel, we show the empirical count of insertion rate weights of arrays for fixed numbers of arrays (genomes with exactly 2, 3 or 4 arrays) colored red, if the respective Cas genes are nearer than 10,000 bp, or colored blue otherwise (genomes with orphan arrays are excluded). The count plots are layered over each other (overlap in dark red). The total counts are shown in black. At the bottom of each panel, in gray, we show the expected neutral distribution of the insertion rate weights, if spacers are inserted at random in available arrays in the genome, for the given amount of samples. Details about the neutral distribution are described in Methods (Section 7.2). Note, that the counts (y-axis) are scaled differently for the observed and expected distribution.
a) shows the weights across the whole dataset and **b)** only for arrays with the same consensus repeat. Note that in a), the distribution of arrays which are further away (dark and light red) dominate the the left side of the plots and the distribution of arrays closer to their Cas genes (blue and dark red) are further to the right. This indidates that arrays close to their Cas genes are generally larger than arrays further away. In b), most arrays are close to the Cas genes, but there is no bias of arrays away from the Cas locus to be longer. This indicates that, even though distance from the Cas genes is relevant for the acquisition rate of arrays, possessing the same consensus repeat is of greater importance.

For comparison, we computed neutral distributions of the insertion rate weights using parameters from the real dataset. In the neutral case, new spacers are inserted in any of the arrays with equal probability, which results in a multinomial distribution of the array lengths. See Methods (Section 7.2) for details about the derivation of the neutral distribution.

As expected, in the neutral case, when all arrays have a similar chance to acquire new spacers, and thus similar lengths, the relative weight concentrates around the inverse of the number of arrays. However, in Figure 7.3a, we see that the overall insertion rate weights are distributed around the expected 0.5/0.33 for 2/3 arrays, respectively, but not around 0.25 for 4 arrays. The distribution has an increasing weight towards the lower end indicating fewer large and more small arrays. In general, the observed distributions have significantly higher variance and are wider.

For the arrays close to their Cas (light red/dark red) and the arrays far away from their Cas (blue/dark red) the distribution of relative lengths is shifted and arrays with close Cas tend to have larger weights than those with a more distant Cas locus. This can also be seen in the distribution of array lengths with respect to their proximity to Cas (Supplementary Table S4, Supplementary Fig. S11 and S12).

One might be tempted to conclude from this pattern that spacer acquisition is more efficient if the acquisition machinery is positioned closer to the array. However, when we further reduced the available arrays to arrays with the same consensus repeat we found that the left shift of the distributions vanishes. In Figure 7.3b, the distribution of arrays more distant to the Cas locus is entirely enveloped by the distribution for close to Cas arrays. Moreover, more weight of the distribution is around 0.5/0.33/0.25 as expected for equally distributed acquisitions among the 2/3/4 arrays and the variance is closer to the neutral expectation. This effect appears to be Cas gene-specific, as shown in Supplemental Table S4: some types exhibit a large decrease in (average) array length for distant arrays, while others show no noticeable decrease.

Nonetheless, this indicates that although the distance to the Cas is correlated with array length, possessing the same (consensus) repeat is far more important for consistent acquisition of new spacers. Consequently, arrays with the same repeat often maintain comparable lengths and are equally likely updated regardless of their distance to the Cas locus.

The relevance of the (first) repeat for spacer acquisition is supported by recent findings that Cas1-Cas2 complexes show higher insertion efficiency by matching to parts of the leader and the first repeat for acquisition in type II-A (Kim et al. 2019). Moreover, repeats were found to be relevant for the processing of crRNA in type II systems. It was found that the leader forms a stem-loop structure with the first leader adjacent repeat which increases crRNA processing efficiency at the leader end and inhibits creation of non-functional 'extraneous' crRNA (Feussner et al. 2025; Liao et al. 2022).

Note that we only analyze the acquisition rates for new spacers, and we can not infer the expression rates of these spacers as crRNA to draw direct conclusions about the conferred immunity or the impact of Cas distance on expression efficiency. Lastly, possessing the same consensus repeat could also be a proxy measure for the arrays to have the same leaders, which have been found to be crucial for acquisition and expression of CRISPR arrays.

We are aware of only one experimental setup in which acquisitions split among two arrays are considered. While the authors also found that both arrays are used for acquisition, the relative acquisition rates are hard to disentangle from a potential copy number bias

towards the origin of replication in a replicating bacterial culture (Peach et al. 2025).

7.3.6 Repeat distances cluster according to Cas types

We computed the (normalized) Levenshtein distance between consensus repeats of the dataset. We found that, in general, the distribution of the normalized consensus repeat distances is bimodal (or even multimodal) with one focal point around zero and one or even more focal points in the distance range 0.4–0.6 for increasing number of arrays and types (Supplementary Fig. S13).

Multiple arrays close to the same Cas locus frequently occur, and, in this case, it is likely beneficial to maintain similar repeats. Moreover, some types are known to use the acquisition machinery of other Cas loci and thus might benefit from similar repeats. Both cases result in the mode around 0.

In general, repeat similarity might not be necessary or even detrimental when supporting multiple CRISPR-Cas systems with different tasks. This, and an increasing number of co-occurring different types of Cas for which only a limited number are compatible likely leads to the additional modes.

To characterize and compare the repeats within Cas types and for different Cas types further, we performed MDS, with respect to the pairwise distances of the consensus repeats, to project the repeats of each array into a 2-d space; see Figure 7.4 and Supplementary Fig. S14, S15.

MDS is a method to translate the pairwise distances of a set of points (here the consensus repeats of each array) into a less-dimensional (here 2-d) space such that the pairwise distances are maintained as well as possible. Thus, MDS allows us to visualize the similarity of the consensus repeats losing as little information about their edit distance as possible; see Methods (Section 7.2) for details about the MDS. Note that the MDS was performed using all arrays, and all Figures are created using the same MDS projection, only different subsets of arrays are shown for visual clarity.

Most type I repeats form clearly distinguishable clusters with some outliers (which occur for all types). However, it is remarkable that types I-B, I-E and I-F, although they have more similar repeats, co-occur only rarely (Figure 7.1) with I-E and I-F in particular avoiding most other Cas types, excluding some of the widely spread V and VI subtypes.

Type V and VI show very wide distributions that populate almost the whole MDS space which is very similar to the seemingly random distribution exhibited by the orphan arrays (Supplementary Fig. S14). This could suggest that they are very widely compatible with other Cas types, an idea that is supported by their relative frequent co-occurence with other types seen in Figure 7.1 and experimental studies (Hoikkala et al. 2021; Laanto et al. 2017) who found that a VI-B array used the acquisition machinery of a II-C locus. However, it can also suggest that many type V and VI arrays might be misclassified or false CRISPR detections (Buyukyoruk et al. 2023; Zhang et al. 2017).

Types II tend to fall somewhere in between types I and V/VI showing wider distributions with some still distinguishable clusters. In particular, type II-C shows multiple clusters.

Supplementary Fig. S15 shows the repeat MDS colored by distance to their Cas locus, revealing that repeats located near their Cas locus cluster tightly, whereas those farther away are highly dispersed. This further reinforces that these arrays might be misclassified as the wrong type, inactive, or false CRISPR detections (Buyukyoruk et al. 2023; Zhang et al. 2017).

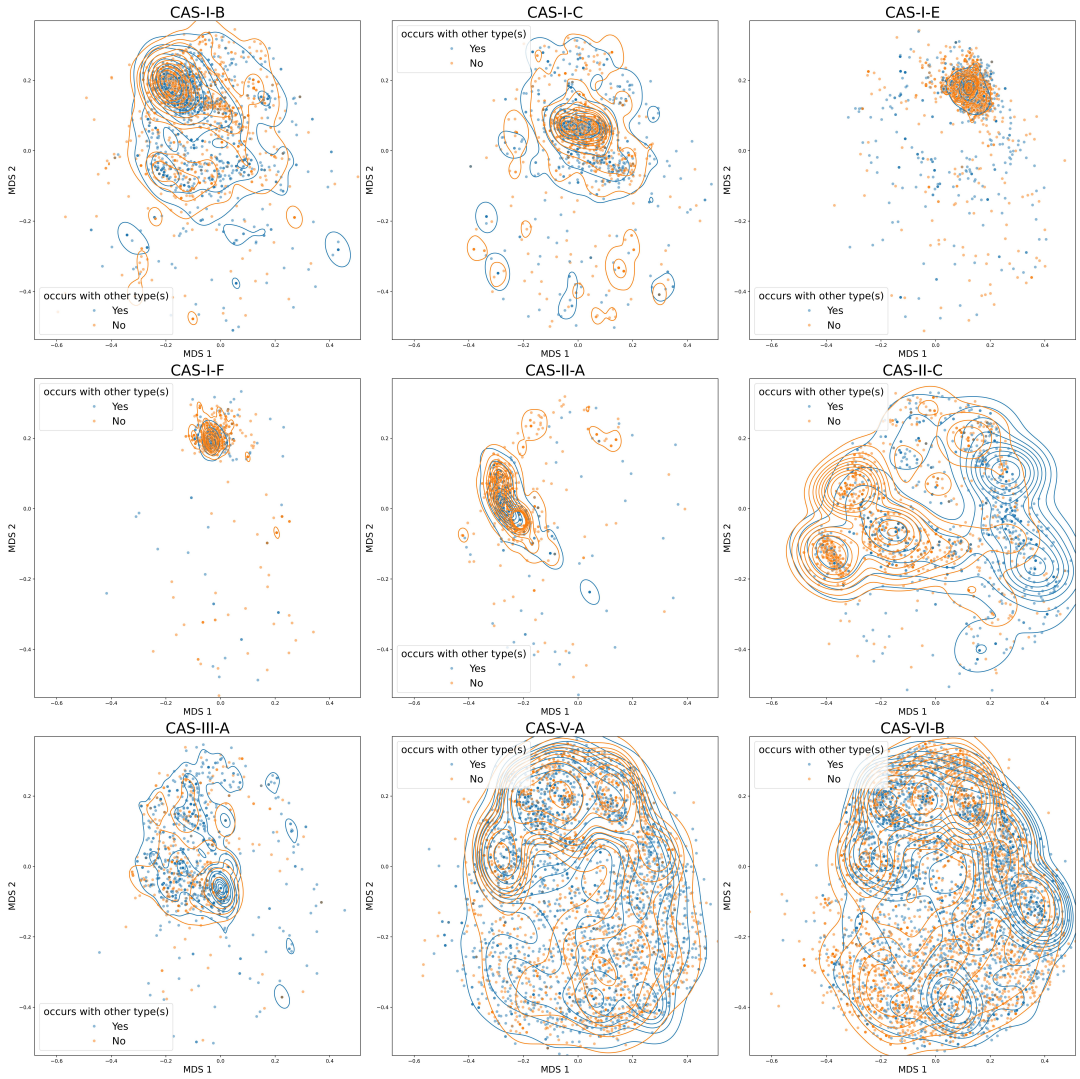


Figure 7.4: Visualization of repeat similarity by projection into 2-d space with multi-dimensional scaling. The plot shows the projection of the consensus repeats (for each array) into 2-d space with multi dimensional scaling (MDS) with respect to the pairwise consensus repeat edit distances (see Section 7.2 for details). The MDS is done globally across all arrays and types, i.e. all shown figures are using the same MDS projection and are in the same coordinate system. However, we separated the Cas types into subplots for visual clarity. Each dot corresponds to the consensus repeat of one array and is colored blue, if other Cas types were found within the same genome, and orange otherwise. Each single dot is semi-transparent, thus more intensive color indicates multiple arrays with the same consensus repeat. To illustrate where the distributions are most concentrated, we show the contours of kernel density estimates for both co- and non-co-occurring types in their respective colors.
We show only a selection of representative Cas types here; see Supplementary Fig. S14 for plots of all Cas types. The types show large differences in their distributions. In particular, type I arrays tend to have very concentrated distinguishable distributions and have closely related repeats. However, type II shows much more variance, in particular, type II-C shows multiple distinguishable clusters. Type V-A and VI-B are distributed across the whole space and seem to share similar repeats with all other types.

7.3.7 Type II-C repeats cluster into subgroups

In Figure 7.5, we show a more detailed breakdown of the MDS projections with marginal distributions for a subgroup of types.

For II-C in Figure 7.5a there are clear differences between repeats of II-C that predominantly co-occur with other types and smaller clusters where no other Cas type is present. This is consistent with the classification by Makarova et al. (2020) where they found two subtypes of II-C arrays. However, considering Supplementary Fig. S15, it becomes clear that arrays in the cluster on the right, which co-occurs with other Cas types, are mostly far away from their Cas locus and thus have remarkably small arrays with often less than 5 spacers; see Supplementary Table S4 and Supplementary Fig. S12. It is possible that these are misclassifications, however, none of the other Cas types shows similar behavior, and their close clustering suggests that this might be a special case for type II-C.

The type II-C system has been shown to be peculiar in multiple ways, e.g. they acquire spacers at the 3' end (Hoikkala et al. 2021; Laanto et al. 2017; Mitrofanov et al. 2026; Zhang et al. 2013), carry promoters in the repeats (Dugar et al. 2018; Hoikkala et al. 2021; Zhang et al. 2013) and show unique characteristics in crRNA expression (Feussner et al. 2025). Moreover, to date, orientation tools fail to reach a consensus on the orientation of many type II-C systems (Chapter 6).

These particularities point towards the unique functionality of type II-C systems, potentially resulting in multiple distinct subtypes and the pattern observed here.

7.3.8 Repeat similarity of co-occurring CRISPR systems

In Figure 7.5b–e, we show the repeat MDS projections for pairs of CRISPR types with high co-occurrence rate (Figure 7.1). They show a differentiated picture, suggesting that beneficial co-occurrence of types can be efficient for very varied repeat agreement.

We see in Figure 7.5b that type III-A has a high overlap with the middle II-C cluster. However, type II-C arrays with co-occuring types (and in particular III-A) are mostly found within the right cluster. Thus, II-C and III-A repeats that co-occur have, in most cases, different repeats.

In general, repeat similarity could be important for efficient spacer acquisition in trans. I-B and III-B clearly have very similar almost exactly overlapping distributions for their repeats (Figure 7.5c). Makarova et al. (2020) found that I-B has spacer acquisition genes, like Cas1 and Cas2, and III-B does not. This suggests that III-B uses the acquisition machinery of I-B. Vink et al. (2021) found results consistent with this in the distribution of spacers and targets of arrays of both types. Similarly, plasmid-encoded adaptation-deficient type IV systems have also been found to frequently co-occurr with host-encoded adaptation-competent systems with leader, repeat and PAM similarity (Pinilla-Redondo et al. 2020).

III-A and V-F share similar repeats for only a subset of co-occurring arrays (Figure 7.5d). According to Makarova et al. (2020) both type III-A and some subtypes of V-F do not necessarily possess integral genes for spacer integration such as Cas1 and Cas2. This suggests that it might be beneficial for III-A and V-F to maintain similar repeats and pool their Cas machinery to allow for new spacer acquisition and to retain functionality.

It is remarkable that, in both Figures 7.5b and d, III-A is shown to have a very concentrated distribution of repeats, but co-occurs with much more universalist types in terms of repeat diversity and negative aloneness score (II-C, V-F, VI-B, Figure 7.1). Moreover, Hoikkala et al. (2021) show that for II-C and VI-B – arrays with only limited repeat similarity –

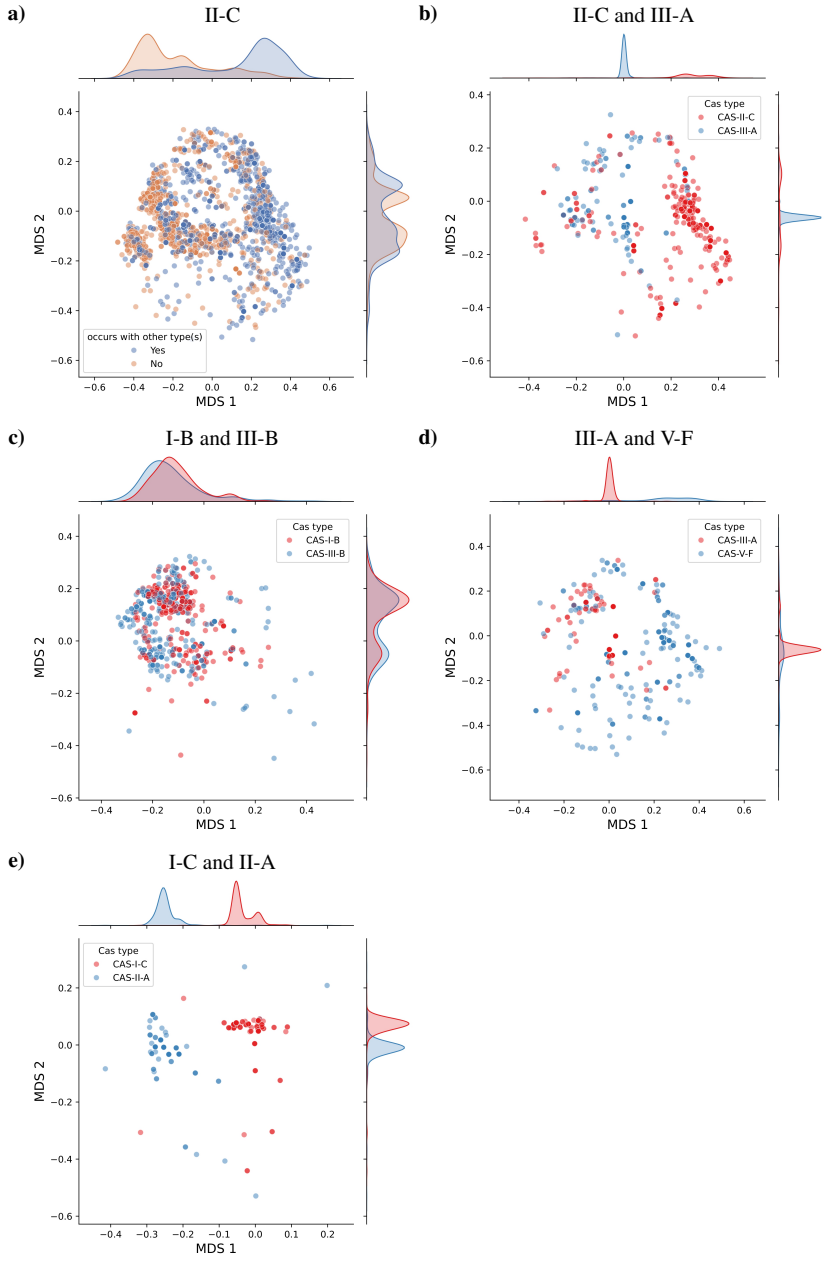


Figure 7.5: Repeat similarity MDS joint scatterplot and marginal distributions. In this plot, we show the MDS projections of the repeats shown in Figure 7.4 with additional kernel density estimates of the marginal distributions for both MDS coordinates for interesting subsets of arrays. See Methods (Section 7.2) for details about the MDS and kernel density estimates.
a) shows only Cas-II-C which clearly shows the existence of different subgroups of repeats. Moreover, we see that they can be separated into groups which occur more frequently and less frequently with other Cas types. **b)** – **e)** show only the repeats of arrays in genomes that posess arrays of both Cas types that have been found to significantly co-occur (see Figure 7.1), as noted in each title. **b)** shows that co-occurring II-C and III-B repeats are most often distant from each other. Most II-C repeats are found in the right cluster and are mostly far away from their Cas and shorter in length (see Supplementary Fig. S12, S15). In **c)**, repeats of types I-B and III-B are very similar and similarly distributed with almost complete overlap. In **d)**, some arrays of type III-A and V-F have similar repeats, but in general are relatively distant from each other, and V-F has a much wider distribution than III-A. I-C and II-A in **e)** clearly have distant distinct repeats.

VI-B can acquire spacers using the II-C machinery.

I-C and II-A are remarkable in the sense that their repeats are related, but they do not overlap in Figure 7.5e. Moreover, they are the types with the highest co-occurrence ratio and according to Makarova et al. (2020) both have functional genes for spacer acquisition. This suggests that these types fulfill complementary and beneficial functions but operate independently for spacer acquisition.

CRISPR systems could also cooperate during interference, utilizing other Cas genes in trans or employing spacers or crRNA from other arrays in their effector complexes. The co-opting of a single Cas6 protein to process crRNA for both a type I-A and two III-B systems was experimentally observed in *Sulfolobus islandicus* (Deng et al. 2013). There is also extensive experimental evidence for the sharing of crRNA between DNA and RNA targeting systems which could provide sustained defense against phages that employ anti-CRISPR tactics (Majumdar et al. 2015). Majumdar et al. (2015) found crRNA from all 7 arrays in the genome of *Pyrococcus furiosus* in the effector complexes of the 3 co-occurring Cas loci of types I-A, I-G and III-B.

Additionally, different CRISPR systems could serve different roles at different stages of phage defense. For example, Silas et al. (2017) found that the type III-B system in *Marinomonas mediterranea* uses crRNA from type I-F arrays in the same system. They propose that the III-B system serves as a backup for the I-F system in the event of phage evasion via PAM mutation since the III-B system is not PAM dependent. Moreover, the type VI interference mechanism (Cas13) induces collateral RNA degradation when a certain amount of target RNA is detected within the cell, rapidly resulting in cell dormancy or death (Vialetto et al. 2022). Thus, the frequently co-occurring VI-B systems could serve as a "fall-back" system for the cell in the event of catastrophic infection with an invasive element when other more specific CRISPR systems have failed to stop the infection.

7.3.9 Summary and open questions

In Table 7.1 we give a short summary of our findings, relevant subtypes where these findings are observable, and state some of the possible mechanisms underlying the observations as suggested in the literature. There are many other open questions that go beyond our findings, such as the impact of the distance to the Cas locus on expression and interference efficiency, as well as the impact of Cas and array orientation on acquisition, expression, and interference efficiency.

The most notable fact is the sheer complexity of the dynamics of CRISPR system co-occurrence and interaction, additionally to the large diversity of CRISPR types in their array and Cas configuration. Naturally, interaction between systems are hard to explore in empirical studies since they introduce additional complexity and biases.

However, it is important to understand why some systems occur more frequently together (Figure 7.1), what the benefit of the co-occurrence of systems is, and through which mechanisms these systems can interact. Figure 7.1 and Supplementary Table S5 offer candidates for such interacting systems that could merit detailed experimental investigation. Moreover, Figure 7.5 illustrates that these systems might interact on different levels of acquisition, expression, and interference.

We mainly focused on genomes with a limited number of Cas types and arrays to mitigate the impact of statistical noise. However, the large number of genomes with diverse Cas systems require further investigation to better understand their interaction. Recent studies such as Hoikkala et al. (2021) and Margolis et al. (2025) and our results indicate that

the interaction between systems is crucial and underexplored. Margolis et al. (2025) investigated the interactions between three CRISPR systems of type I-B, II-C and VI in *Listeria seeligri*. They observed that type VI uses II-C adaptation to insert spacers and I-B primes the acquisitions of VI and II-C. This suggests that communication between multiple systems might be more complex than previously expected.

Together with these results, our findings, including the observation that arrays with identical repeat have similar acquisition activity across vast genomic distances, suggest that we require a more holistic approach considering all CRISPR systems across the genome to understand their function and interaction. Furthermore, the purpose of possessing multiple (for some types very close) CRISPR arrays with various orientation configurations are uncertain. Although several hypotheses have been proposed to explain the distribution of spacers across multiple arrays (Bernheim et al. 2019; Weissman et al. 2018b), much remains to be understood – particularly through experimental studies – about how spacer acquisition, deletion, and interference proceed in systems harboring multiple arrays and Cas types.

7.4 Conclusion

We show that the arrays of CRISPR-Cas systems have a remarkable diversity that can diverge wildly between types. However, some clear patterns emerge.

Although array length decreases for arrays distant from the next Cas locus, hinting at a low spacer acquisition rate, this effect almost completely vanishes if the repeats of the arrays agree. This suggests that arrays with identical repeats are equally likely to acquire new spacers, regardless of their genomic position and distance to the Cas locus. Moreover, types show clear distinct patterns of positioning and orienting their arrays around the Cas locus.

The observed variations might be a result of the selective pressure to diversify the bacterial defense reportoire to avoid or compensate anti-CRISPR mechanisms and persist in the evolutionary arms race with phages.

The different patterns in the frequency of co-occurrence of Cas types and the differences in repeat (dis)similarity and array orientation identified here invite further exploration. Understanding the underlying mechanisms shaping these patterns – including spacer acquisition and inference in trans, layered defense strategies, and redundancies within prokaryotic immune systems – will help to assess the synergies and incompatibilities of prokaryotic defense systems.

Observations and findings - further details and/or functional implications proposed in literature	Related Fig./Tables	Related Cas type(s) (pairs)	Related Studies
CRISPR system co-occurrence	**Fig. 7.1**		(Bernheim et al. 2019)
Some types co-occurr frequently	Table S6	e.g. (I-B, III-B), (I-C, II-A), (I-C, V-A), (II-A, V-A), (II-C, III-A), (III-A, V-F), (III-A, VI-B), (III-A, V-F), ...	
- Acquisition (and priming) in trans (Cas1/Cas2 sharing)			(Hoikkala et al. 2021; Laanto et al. 2017; Majumdar et al. 2015; Margolis et al. 2025; Pinilla-Redondo et al. 2020; Vink et al. 2021)
- Supporting/complementary interference mechanisms			(Deng et al. 2013; Hoikkala et al. 2021; Margolis et al. 2025; Silas et al. 2017; Vialetto et al. 2022; Vink et al. 2021)
Some types tend to occur alone	Table S5	I-B, I-C, I-E, I-F	(Almendros et al. 2016; Touchon et al. 2010)
- due to conflicts between competing systems, e.g. inhibition of HGT?			(Almendros et al. 2016; Liu et al. 2025; Touchon et al. 2010)
CRISPR array orientation relative to Cas loci	**Fig. 7.2, S8**		(Milicevic et al. 2019)
Arrangement often consistent within, but varies between Cas types		I-E, I-F, II-C, III-A, III-B, ...	
- Multiple arrays flanking Cas loci show distinct orientation patterns	Fig. S10	I-B, I-E, I-F, II-A, III-A, ...	
- Array position wrt. Cas locus is type-specific	Fig. S9		
Acquistion frequency between multiple arrays	**Fig. 7.3**		
Spacers are distributed among multiple arrays of same Cas type	Table S1	all types to some extent; notably, I-D, I-E, I-F, III-A	(Bernheim et al. 2019; Majumdar et al. 2015; Martynov et al. 2017; Peach et al. 2025; Weissman et al. 2018b)
Arrays with identical repeats have similar acquisition frequencies, regardless of distance from Cas locus			
- Acquisition more efficient for matching repeats (and leaders)			(Kim et al. 2019)
For most types more distant arrays are shorter...	Table S4	I-B, I-C, I-E, I-F, II, V-A, ...	(Peach et al. 2025)
...some types do not show this effect		I-A, III-B, III-D	
Repeat sequence similarity	**Fig. 7.4, S14**		
Repeat conservation varies by Cas type		I-B, I-C, I-E, I-F, II-A, II-C, III-A, V-A, VI-B	(Kim et al. 2019)
- Repeats influence crRNA processing			(Feussner et al. 2025; Liao et al. 2022)
Larger array-Cas locus distance correlates with higher repeat diversity	Fig. 7.1, S15, Table S4	V-A, V-F, VI-B	(Buyukyoruk et al. 2023; Zhang et al. 2017)
II-C repeats cluster into subgroups	Fig. 7.5a	II-C	(Fehrenbach et al. 2025; Makarova et al. 2020; Makarova et al. 2025)
Various repeat similarity patterns between co-occurring Cas types	Fig. 7.5b–d	(II-C, III-A), (I-B, III-B), (III-A, V-F)	(Liao et al. 2022; Majumdar et al. 2015)
I-C and II-A co-occur, but have distinct repeats	Fig. 7.1, 7.5e	(I-C, II-A)	

Table 7.1: Summary of findings and possible functional implications proposed/supported in the literature. The findings are separated into 4 sections marked in bold. For each finding/section, we provide the relevant figures along with related studies and literature references for the entire section. Similarly, for each point within these sections, we provide additional relevant figures and tables and indicate the Cas type(s) (pairs) for which the observations/findings are most evident. Rows marked by bulletpoints (-) give some additional details and/or functional implications that arise from the observations and are proposed and/or supported by the literature shown in the related studies column.

8

Applications, perspectives, and conclusion

Our contributions in Chapters 4, 5, 6, and 7 reveal the remarkable diversity of CRISPR systems, their evolutionary dynamics, and their integration within broader genomic and ecological contexts, while also providing novel computational tools for their characterization and analysis. Collectively, these findings advance our understanding of CRISPR array evolution, spacer dynamics, orientation, and diversity across diverse systems.

Despite substantial progress, many fundamental questions remain unresolved. In this chapter, we discuss several of these open directions, ranging from limitations in currently available data and the potential of industrial, biotechnological, and medical applications to refinements and extensions of our statistical framework, as well as the application of our methods to related array-like genetic systems such as integrons. In particular, we also highlight the importance of considering CRISPR-Cas systems in a genome-wide context, as interactions between multiple systems and arrays within the same genome suggest that a system-level perspective is necessary to fully understand CRISPR dynamics.

Together, these perspectives highlight both the broader applicability of this work and its considerable potential for further development, with several of these directions already explored in preliminary follow-up studies.

8.1 Data availability and applications in industry, biotechnology and medicine

The effectiveness of statistical approaches is strongly constrained by the availability and diversity of data. Industrial, biotechnological, and medical applications generate extensive datasets with rich variation, offering valuable opportunities to deepen our understanding of CRISPR systems.

Given the major role of diverse bacterial communities in fermented foods, foodborne disease and the emerging issue of antimicrobial resistance, monitoring and controlling microbial communities is critical across food production, livestock, and aquaculture systems (Ganote et al. 2025; Laanto et al. 2017; Landor et al. 2026; Li et al. 2021; O'Sullivan et al. 2019; Palmnäs-Bédard et al. 2023; Pertziger et al. 2025; Somerville et al. 2022a; Somerville et al. 2022b; Wang et al. 2023).

Microbial communities play a central role in human and animal health as pathogens but also in often beneficial microbial communities within the body such as gut microbiomes (Avershina et al. 2026; Camarillo-Guerrero et al. 2021; Ganote et al. 2025; Lam et al. 2019b;

López-Beltrán et al. 2024; Molina-Sánchez et al. 2024; Ndiaye et al. 2025; Zhang et al. 2025). Their dynamics are to a large extent shaped by host-phage interactions, driven by the interplay between phage infection strategies and microbial defense systems, which can be effectively studied through analysis of CRISPR spacer content (Achigar et al. 2017; Camarillo-Guerrero et al. 2021; Dion et al. 2021; Soto-Perez et al. 2019; Vink et al. 2021).

In medical contexts, phage-based therapies have re-emerged as a promising approach for targeting pathogenic and unhealthy microbial communities, as well as for addressing the growing challenge of antibiotic resistance (Benz et al. 2025; Dedrick et al. 2019; O'Sullivan et al. 2019; Seed et al. 2014).

In parallel, CRISPR-based genotyping offers a powerful framework for high-resolution tracking of clinical pathogens, enabling improved surveillance and characterization of infectious agents (Barrangou et al. 2016; Beauruelle et al. 2018; Beauruelle et al. 2021; Lam et al. 2019a; Louwen et al. 2013; Tomida et al. 2017).

From our perspective, time series data of CRISPR arrays are a particularly valuable resource, as they enable high-resolution tracking of spacer acquisitions and deletions and provide a means to test hypotheses on spacer array evolution in controlled settings. An increasing amount of metagenomic time-series data is now available from systems such as biotechnological and pharmaceutical bioreactors (Martínez Arbas et al. 2021), wastewater treatment plants (Guerrero et al. 2021), and acid mine drainage environments (Sun et al. 2016), although complementary long-read data remain scarce.

The study of these abundant sources of genomic data has yielded crucial insights into CRISPR system functionality and bacteria-phage interactions. However, confidently determining spacer order, an essential prerequisite for analysing spacer array evolution, remains difficult in metagenomic data. Long-read sequencing is therefore particularly important for resolving CRISPR arrays in general, and in time-series studies, where it enables reliable reconstruction of spacer dynamics over time. We therefore encourage increased adoption of long-read sequencing approaches, as broader application of these technologies is likely to substantially improve our understanding of CRISPR system evolution and function.

Although we believe that spacer-phage correspondence analysis with *SpacerPlacer* is particularly valuable for resolving dynamics in smaller, local microbial communities, providing a means to assess the impact of individual spacers and to identify potential selection effects, it fundamentally depends on the combined availability of (metagenomic) data from the mobilome and high-quality whole-genome sequencing of the corresponding host organisms.

A major limitation in this context is the incomplete characterization and annotation of the mobilome compared to bacterial genomes. Recent studies report that only around 10% of CRISPR spacers in comprehensive datasets can be mapped to known sources (McGinn et al. 2019; Shmakov et al. 2017). Our preliminary analyses of datasets from local microbial communities yield similarly low mapping rates for phage origins, reinforcing this observation.

As a consequence, both fine-scale comparative studies, aimed at identifying particularly beneficial or detrimental spacers and associated selective pressures within small populations, and large-scale analyses seeking to uncover broader structural selection patterns beyond those described in Chapter 5, remain limited by current data availability. However, with the continued growth of long-read sequencing and metagenomic sampling efforts, these approaches are expected to become increasingly powerful and enable more detailed and comprehensive analyses in the future.

8.2 Refining *SpacerPlacer* and its statistical framework

Building on the available data, several natural avenues emerge to further develop and refine the methods presented in Chapters 4, 5, 6, and 7.

As shown in Chapter 5, the simple distance-based tree estimation implemented in *SpacerPlacer* (Section 5.2.1) already demonstrates promising performance in phylogenetic analysis. However, its effectiveness is currently limited by the availability of higher-resolution data, and its efficiency and accuracy could be substantially improved by integrating established and more advanced phylogenetic inference methods into the algorithm.

The study and visual analysis of spacer content in bacteria-phage systems is an important component of many biological and bioinformatic analyses. While *SpacerPlacer* already provides useful visualizations of array evolution and spacer composition, there remains substantial potential for deeper integration into existing analytical workflows as well as for improved visualization capabilities.

From an accessibility standpoint, although a script for processing output from CRISPRCas-Finder is currently available, tighter and more automated integration with CRISPRCas-Finder and other established CRISPR array detection tools would further enhance usability and broaden adoption. More recently, we have also explored the development of a server-based pipeline combining CRISPRidentify and *SpacerPlacer*, supported by an integrated database for querying and contextualizing individual or groups of arrays in terms of repeat similarity, Cas type, and spacer overlap to improve ancestral reconstructions and parameter estimates. This would substantially extend the analytical capabilities of *SpacerPlacer*.

In terms of visualization, progress in this direction has been made in a Master's thesis by Jan-Martin Romberg, who developed *CRAAnVis*, a tool enabling enhanced interactive visualization of ancestral reconstructions generated by *SpacerPlacer* (Romberg 2024). In particular, it supports spacer grouping and target annotation, thereby facilitating more effective analysis of spacer-phage correspondence.

SpacerPlacer is a flexible framework in the sense that, beyond its core assumptions such as polarized spacer acquisition, it can be adapted to different evolutionary models for both reconstruction and likelihood-based inference. At present, the block deletion model and its associated likelihood function appear to capture the observed deletion patterns well and provide reliable estimates under the assumed framework. However, it is also clear that deletion dynamics are influenced by additional, potentially more complex processes, as indicated for example by the long-tailed distribution of deletion lengths, suggesting that the current model may require refinement.

A limitation to further increasing model complexity is that previous attempts to compute more detailed likelihoods have in fact led to less accurate estimates, despite the fact that the underlying differential equations are analytically tractable. This issue appears to arise from stochastic dependencies between spacer sites, which complicate accurate likelihood computation and bias correction in more elaborate formulations. At present, spacer acquisition is modeled in a deliberately simple, parsimony-based manner, but this could be extended to incorporate more realistic mechanisms, including Cas-type-specific effects, rate heterogeneity, and priming dynamics.

8.3 CRISPR system interactions and the need for a genome-wide perspective

While the mechanisms of CRISPR interference and adaptation have been extensively studied, and Chapter 5 provided important insights into deletion mechanisms, Chapter 7 reveals that significant gaps persist in our understanding of the evolutionary and functional interplay between CRISPR arrays and associated Cas loci. The extraordinary diversity of these interactions raises broader questions regarding the classification, organization, and evolutionary trajectories of CRISPR systems as a whole.

The careful comparison of Chapters 5 and 7 reveals two contrasting yet complementary perspectives on CRISPR systems.

On the one hand, CRISPR array evolution appears remarkably homogeneous: deletion rates are of similar magnitude in the median across all examined Cas types. This consistency suggests that deletion dynamics are largely independent of specific Cas proteins and instead support previously proposed mechanisms based on strand misalignment during DNA replication and/or DNA repair processes (see Section 3.2 and Ding et al. (2020) and Garrett (2021)). However, the long tail in the distribution of deletion lengths suggests that spacer loss is not solely governed by replication slippage, but also involves additional mechanisms, including processes that may induce array collapse.

On the other hand, our findings also demonstrate substantial diversity across numerous aspects of CRISPR arrays. Although overall deletion rates are broadly consistent across Cas types, deletion dynamics themselves exhibit considerable variance, indicating substantial heterogeneity at the level of individual evolutionary trajectories. Beyond this, we observed pronounced variation in array length, genomic positioning relative to associated Cas loci, the co-occurrence of multiple arrays and Cas types, and repeat sequence diversity across a broad and comprehensive collection of genomes. Given that deletion behavior appears largely conserved across Cas types and is likely driven by the intrinsic structural properties of spacer arrays themselves, the extensive variation in array length, architecture and organization is more plausibly explained by differences in spacer acquisition and interference dynamics; see Chapter 7. These differences are likely shaped by variation in repeat sequences, leader regions, promoter structures, and, most importantly, the functional diversity of associated Cas proteins.

In particular, the more exceptional Cas types that exhibit distinct behavior – most notably type II-C, which shows consistent deviations across all our analyses in Chapter 5, 6 and 7 – warrant further investigation.

Recently, mutations within individual CRISPR arrays have been analyzed along both the repeat sequence and array length, revealing an enrichment of mutations in terminal repeats as well as distinct patterns across different Cas types (Mitrofanov et al. 2026). Consistent with our results, type II-C exhibits especially pronounced differences, likely related to its reversed orientation and functional characteristics.

Preliminary analyses conducted in the Bachelor's thesis of Yamin Maazou incorporated simulations of the deletion process in which single or multiple spacer deletions also incorporate segments of adjacent repeats. This mechanism gives rise to characteristic mutation patterns, some of which are observable in CRISPR arrays from the CRISPRCasdb dataset, while others are not consistently reproduced in empirical data (Maazou 2026). Consistent with Mitrofanov et al. (2026), we also observe distinct mutation signatures across Cas types, which merit further investigation. A more detailed analysis of repeat-associated mutation patterns may provide important in sights into deletion mechanisms, potential

selective influences, and the functional relevance of repeat regions in crRNA maturation.

At the same time, many of the observed patterns appear to arise not solely from functional specialization, variation in CRISPR systems, or selective pressures, but also to a considerable extent from inherent statistical variation driven by genomic architecture and fundamental biological or biochemical processes.

For example, patterns such as the apparent increased conservation toward the distal end of CRISPR arrays, previously widely interpreted within the CRISPR research community as evidence of selective preservation of older spacers, may instead largely arise from inherent structural and mechanistic properties rather than direct selective pressures acting on spacer content. Our findings in Chapter 5 suggest that this pattern can be more parsimoniously explained as a consequence of the underlying deletion process itself, with reduced deletion rates toward the distal end mirrored by corresponding dynamics at the acquisition end of the array (see Figure 5.8). Although, as discussed in Section 5.4, selective effects do influence deletion dynamics, they appear to do so in a different manner than previously assumed.

As outlined in Section 7.3.9 and Table 7.1, analyzing individual CRISPR arrays solely in conjunction with their nearest Cas locus appears insufficient to fully explain the extensive diversity observed across CRISPR systems. Our findings suggest that interactions between CRISPR arrays and Cas cassettes may be substantially more complex and interconnected than is captured by such a limited framework.

Recent experimental studies, discussed in Section 2.3.4, suggest that substantial redundancy and functional cooperation, in spacer acquisition, crRNA processing and interference, may exist among co-occurring CRISPR systems (Deng et al. 2013; Hoikkala et al. 2021; Majumdar et al. 2015; Margolis et al. 2025; Silas et al. 2017; Smith et al. 2025; Vialetto et al. 2022; Vink et al. 2021). Several of these observations are supported by our results in Chapter 7. In particular, Figures 7.1 and 7.5 reveal clear patterns of co-occurrence and repeat similarity among experimentally validated cooperating systems, such as types I-B and III-B. At the same time, systems associated with collateral interference (e.g., types III, V, and VI) exhibit a broad distribution, whereas others tend to occur more independently, such as types I-E and I-F.

Complementing these findings, Almendros et al. (2016) and Touchon et al. (2010) demonstrated that an *E. coli* strain harboring a functional type I-E system can actively inhibit the restoration of a type I-F system to functionality; see Section 2.3.2 and 7.3 for details. This observation may reflect selective pressures that disfavor the co-occurrence of certain CRISPR system types, rather than direct antagonistic interactions, and highlights the complexity of their interplay within a single genome.

As noted by Makarova et al. (2020), the subtyping of Cas systems based on Cas gene sequence similarity can be somewhat arbitrary, particularly for class 2 systems and less prevalent types. In a similar vein, our results cast doubt on the common practice of assigning Cas types to CRISPR arrays solely based on the Cas gene composition of the nearest locus, an approach often motivated by the assumption that co-transcription of Cas genes and associated arrays promotes functional efficiency and therefore favors their genomic proximity.

We find evidence that Cas genes and CRISPR arrays may be shared or functionally associated across considerable genomic distances, and that spacer acquisition appears to be relatively evenly distributed among arrays with similar repeat sequences (Figure 7.3). Furthermore, multiple arrays of various orientations may be distributed across and around single Cas loci (Figure 7.2).

Likewise, the observed differences in repeat diversity across Cas type pairs suggest that classification based on repeat similarity alone may be insufficient to reliably characterize interacting CRISPR systems (Chapter 7).

While our results in Chapter 5 indicate that spacer overlap between CRISPR arrays could provide additional discriminatory power, preliminary analyses in Chapter 7 (Supplementary Table S2) reveal generally low levels of overlap with substantial variability across Cas types.

Taken together, these findings suggest that no single criterion is sufficient for robust Cas system characterization. Instead, a comprehensive framework that integrates multiple complementary features is likely required to more accurately capture Cas type relationships and functional organization.

In conclusion, we propose that understanding and classifying CRISPR functionality in prokaryote defense requires a more comprehensive perspective that focuses on interactions between multiple systems within single genomes, rather than analyses restricted to individual Cas types. Such inter-system relationships appear to be more influential than variation within individual types alone.

There are, however, clear reasons why such studies remain rare. Experimental designs involving multiple co-occurring CRISPR systems are inherently difficult, particularly due to the challenge of controlling for confounding effects. Likewise, disentangling these interactions at scale through comparative or statistical analyses is equally demanding, especially given the substantial variability across CRISPR systems, which can obscure underlying selection and interaction signals.

Nevertheless, we argue for more integrative experimental approaches that explicitly address CRISPR system interactions, complemented by advanced statistical analyses at both local (e.g. with *SpacerPlacer*, Chapter 5) and global scales (Chapter 7).

8.4 Applying CRISPR array models to integron evolution

Beyond CRISPR systems, other bacterial genetic elements also exhibit ordered, array-like architectures, and repetitive sequence structures are widespread in both prokaryotic and eukaryotic genomes. However, CRISPR arrays with their short repeat-separated spacer regions and active evolution through polarized acquisitions and deletions appear to be largely unique. Although CRISPR-like structures have been reported in mimiviruses (Levasseur et al. 2016), humans (Van Riet et al. 2022), and plant mitochondria (Mojica et al. 2000), these findings remain highly disputed and are generally not considered evidence of functional CRISPR systems (Buyukyoruk et al. 2023; Claverie et al. 2016).

To our knowledge, the only confirmed structures that closely resemble CRISPR arrays are *integrons*, which are widespread across bacterial species (Ghaly et al. 2021b; Néron et al. 2022). Integrons contain palindromic *attC* repeat structures (57 – 141 bp in length) separating *gene cassettes*, which are typically much larger than CRISPR spacers (generally 500 – 1000 bp, up to a few kb) and often encode one or more functional genes (Escudero et al. 2018).

Located nearby is the *intI* gene, encoding an integrase responsible facilitating the incorporation new gene cassettes and thus functionally analogous to the Cas1-Cas2 adaptation machinery in CRISPR systems. Similar to the leader region of CRISPR arrays, integrons contain an *attI* site recognized by the integrase, which generally marks the insertion point for new gene cassettes (Escudero et al. 2018; Vit et al. 2021). Consequently, integrons exhibit polarized cassette acquisition similar to CRISPR arrays (Escudero et al. 2018; Vit et al. 2021). Other repeat- or cassette-associated systems have been described, but none

share the same combination of length, regular organization, and spacer-/cassette-scale modularity.

Integrons were originally discovered through their association with antibiotic resistance genes, which are frequently carried within these gene cassettes (Deng et al. 2015; Stokes et al. 1989). Subsequent studies revealed that, in addition to carrying a large proportion of silent cassettes with no known function, integrons encode a wide range of functions, including anti-phage defense systems and toxin-antitoxin modules, i.e. potentially selfish genetic elements (Buongermino Pereira et al. 2020; Darracq et al. 2025; Getz et al. 2025; Ghaly et al. 2021a; Richard et al. 2024b).

Broadly, integrons can be divided into two classes: *mobile integrons*, typically plasmid-borne, short, and transferable between hosts, playing a major role in the spread of antibiotic resistance; and *sedentary chromosomal integrons*, which can contain dozens to hundreds of gene cassettes (Néron et al. 2022; Richard et al. 2024b). In some *Vibrio* species, these arrays can span 30 – 150 kb (Boucher et al. 2006).

A promoter is typically located near the *attI* site, leading to stronger expression of gene cassettes close to the *attI*-end compared to more distal cassettes (Collis et al. 1995; Vit et al. 2021). In contrast to CRISPR spacers, however, the larger size of gene cassettes allows them to encode their own promoters. Surprisingly, this appears to occur only rarely and is mostly associated with toxin-antitoxin cassettes (Richard et al. 2024a), although broader usage has been reported in some integrons (Michael et al. 2010).

As noted above, the *intI*-integrase facilitates the integration of new gene cassettes, which exclusively take place at the *attI* site (*attI* × *attC* reaction). Comparative analyses of closely related integrons further provide clear evidence for gene cassette deletions (*attC* × *attC* reaction, catalyzed by the *intI*-integrase), potentially including block deletions (Labbate et al. 2007; Richard et al. 2024a). Thus, integrons evolve through cassette acquisition and deletion processes analogous to those observed in CRISPR arrays.

Despite these structural similarities, the underlying biochemical and evolutionary mechanisms of acquisitions and deletions differ substantially from those of CRISPR arrays (Escudero et al. 2018). Current models suggest that *intI*-integrases mediate cassette excisions when two attC sites recombine within the integron, generating a circular intermediate that can subsequently be *reinserted* at the *attI*-site. Importantly, cassette insertion involves asymmetric recombination on a single DNA strand, followed by replication-based resolution, producing daughter molecules in which only one carries the newly inserted cassette. Excision is thought to proceed analogously, such that reintegration into a copy retaining the original cassette can generate duplicated gene cassettes within the same integron (Escudero et al. 2018; Ghaly et al. 2020; Loot et al. 2012).

This reshuffling and duplication of gene cassettes is thought to be a key mechanism in integron evolution. It is considered to be advantageous because it can reposition beneficial cassettes toward the *attI*-proximal region, where expression is enhanced by the nearby promoter (Vit et al. 2021). More generally, cassette duplication may also protect beneficial genes from loss while simultaneously enabling diversification through mutation accumulation in redundant copies (Ghaly et al. 2020).

Expression of *intI* itself appears to be stress-induced, with cassette excision and integration strongly upregulated during SOS responses (Baharoglu et al. 2010; Guerin et al. 2009; Vit et al. 2021). Under conditions such as antibiotic exposure, this mechanism may therefore promote rapid amplification and increased expression of advantageous resistance genes (Richard et al. 2024b).

The principles governing integron gene cassette evolution show clear parallels to those

of CRISPR arrays. Insertions are polarized and deletions/excisions are believed to be initiated by recombining repeats. In particular, there are functional correspondences between CRISPR repeats and *attC* sites, the leader region and the *attI* site, and spacers and integron gene cassettes. As a result, integron evolution can, in principle, be treated in a similar framework to CRISPR array evolution by modeling polarized cassette acquisition and deletion (as well as more complex events), which can be reconstructed and analyzed using *SpacerPlacer* with only minor modifications to the method.

Thus, in collaboration with the Master's student Fabian Rosner, we extended *SpacerPlacer* in his thesis to accept groups of integrons with overlapping gene cassettes as input and reconstruct their gene cassette evolution (Rosner 2025). This revealed two main challenges.

First, integrons require additional clustering steps, as gene cassettes encode larger genes that accumulate substantially more mutations than short CRISPR spacers. Consequently, identifying and grouping homologous cassettes that share a common origin requires more elaborate similarity criteria.

Second, the copy-by-excision-insertion dynamics in integrons differ fundamentally from the acquisition and deletion processes modeled in *SpacerPlacer*. In ideal cases where the data resolution is sufficient to resolve both original and duplicated cassettes, *SpacerPlacer* would interpret these events as duplications or rearrangements as described in Section 5.3.2.

First experiments on integrons detected in whole genomes using *IntegronFinder* (Néron et al. 2022), performed by Fabian Rosner across a selection of species known to carry integrons, revealed relatively few groups with observable evolutionary signal. In addition, there was no clear evidence for a higher frequency of reshuffling and duplication events, with only a small number of duplications, rearrangements, or coupled insertion-deletion patterns.

Accordingly, our preliminary analysis does not provide strong support for the proposed shuffling and duplication mechanism. However, *SpacerPlacer*, out of the box, may have trouble to detect such events, as the presence of multiple daughter lineages with different configurations of gene cassette insertions, copies and excisions, combined with potential selection effects, may result in reconstructions that explain the observed patterns without requiring explicit invocation of more complex evolutionary events. Thus, if reshuffling and duplication events occur at comparable rates to standard insertion and excision processes, it may ultimately be necessary to explicitly incorporate this mechanism into the evolutionary model.

8.5 Conclusion

As we have seen, CRISPR-Cas systems, and CRISPR arrays in particular, are not passive genomic remnants but dynamic substrates of evolution. Their structure records past interactions with mobile genetic elements, facilitates adaptive immune responses, and reflects both molecular constraints and higher-order genomic organization. Features such as their repetitive architecture, continual turnover through deletions and polarized acquisitions, apparent redundancy, and frequent co-occurrence are not incidental but often contribute directly to their function, robustness, and evolutionary potential.

This perspective becomes particularly apparent in the context of microbial warfare. Here, prokaryotes, phages, and other mobile genetic elements engage in continuous cycles of infection, defense, and counter-defense, generating strong and rapidly changing selective pressures. CRISPR-Cas systems provide one of the most direct and structured molecular records of these evolutionary interactions, encoding past encounters within repetitive

genomic architectures that are continually reshaped through spacer acquisition and loss. As a result, CRISPR arrays contain a rich evolutionary archive that can be interrogated using statistical and bioinformatic methods to uncover the mechanisms, evolution, and ecological roles of CRISPR-Cas systems and their interactions with microbial communities.

This thesis provides a genome-wide and mechanistic perspective on CRISPR array evolution, supported by large-scale analyses and the development of dedicated reconstruction tools. Across diverse datasets, a central theme emerges: while core evolutionary processes such as deletion dynamics are remarkably conserved across Cas types, the overall structure of CRISPR systems is highly diverse and shaped by a combination of molecular constraints, selection and statistical effects.

A key result of this work is that many observed patterns in CRISPR organization are not driven primarily by differences between Cas types, although they do exist (Cas type II-C!). Instead, they can arise from fundamental biochemical mechanisms and the underlying structure of genomic processes. Indeed, processes such as polarized acquisition not only contribute to these patterns but also provide a powerful basis for efficient modeling and analysis of CRISPR arrays. At the same time, substantial variation remains in acquisition behavior, genomic positioning, and system composition, pointing to a layered evolutionary landscape in which local mechanisms and broader genomic context jointly shape CRISPR diversity.

The methods developed in *SpacerPlacer* provide a first step toward both qualitative and quantitative reconstruction of these dynamics and improving our understanding of spacer content evolution. While our models already capture key features of spacer turnover, the results also highlight that important processes, including selection, (time-)heterogeneous evolutionary events, and interactions between coexisting systems, remain only partially resolved and require more refined experimental, statistical and modeling frameworks. Accordingly, these findings argue for a shift toward integrated, genome-wide perspectives on CRISPR-Cas in prokaryotes, combining mechanistic modeling, high-resolution data, and system-level interaction analysis.

While this work has addressed several open questions, the remarkable diversity and evolutionary dynamics of CRISPR-Cas systems continue to offer numerous opportunities for discovery. We hope that the concepts, methods, and insights developed here contribute to a deeper understanding of CRISPR array evolution and CRISPR-Cas systems and provide a foundation for future experimental, theoretical, and computational studies, whether conducted by us or by others.

Bibliography

Abby, S. S., Néron, B., Ménager, H., Touchon, M., and Rocha, E. P. C. (2014). MacSyFinder: A Program to Mine Genomes for Molecular Systems with an Application to CRISPR-Cas Systems. In: *PLoS ONE* 9.10. Ed. by N. V. Torres, e110726. DOI: 10.1371/journal.pone.0110726 (cit. on p. 16).

Achigar, R., Magadán, A. H., Tremblay, D. M., Julia Pianzzola, M., and Moineau, S. (2017). Phage-Host Interactions in Streptococcus Thermophilus: Genome Analysis of Phages Isolated in Uruguay and Ectopic Spacer Acquisition in CRISPR Array. In: *Scientific Reports* 7.1, p. 43438. DOI: 10.1038/srep43438 (cit. on pp. 21, 31, 34, 64, 96, 120).

Achigar, R. et al. (2021). Ectopic Spacer Acquisition in Streptococcus Thermophilus CRISPR3 Array. In: *Microorganisms* 9.3, p. 512. DOI: 10.3390/microorganisms9030512 (cit. on pp. 21, 33, 64, 96, 98).

Agari, Y., Sakamoto, K., Tamakoshi, M., Oshima, T., Kuramitsu, S., and Shinkai, A. (2010). Transcription Profile of Thermus Thermophilus CRISPR Systems after Phage Infection. In: *Journal of Molecular Biology* 395.2, pp. 270–281. DOI: 10.1016/j.jmb.2009.10.057 (cit. on p. 33).

Alkhnbashi, O. S., Shah, S. A., Garrett, R. A., Saunders, S. J., Costa, F., and Backofen, R. (2016). Characterizing Leader Sequences of CRISPR Loci. In: *Bioinformatics* 32.17, pp. i576–i585. DOI: 10.1093/bioinformatics/btw454 (cit. on pp. 14, 17, 21, 82, 84, 96, 100, 108).

Alkhnbashi, O. S., Costa, F., Shah, S. A., Garrett, R. A., Saunders, S. J., and Backofen, R. (2014). CRISPRstrand: Predicting Repeat Orientations to Determine the crRNA-Encoding Strand at CRISPR Loci. In: *Bioinformatics* 30.17, pp. i489–i496. DOI: 10.1093/bioinformatics/btu459 (cit. on pp. 83, 84, 93, 97, 101).

Almendros, C., Guzmán, N. M., García-Martínez, J., and Mojica, F. J. M. (2016). Anti-Cas Spacers in Orphan CRISPR4 Arrays Prevent Uptake of Active CRISPR–Cas I-F Systems. In: *Nature Microbiology* 1.8, p. 16081. DOI: 10.1038/nmicrobiol.2016.81 (cit. on pp. 28, 104, 117, 123).

Altae-Tran, H. et al. (2023). Uncovering the Functional Diversity of Rare CRISPR-Cas Systems with Deep Terascale Clustering. In: *Science* 382.6673, eadi1910. DOI: 10.1126/science.adi1910 (cit. on p. 25).

Andersson, A. F. and Banfield, J. F. (2008). Virus Population Dynamics and Acquired Virus Resistance in Natural Microbial Communities. In: *Science* 320.5879, pp. 1047–1050. DOI: 10.1126/science.1157358 (cit. on p. 7).

Avershina, E., Birkeland, E. E., Bucher-Johannessen, C., and Rounge, T. B. (2026). CRISPR-Cas Immune Repertoires as an Ecological Record of Bacterial Interactions with Mobile Genetic Elements in the Human Gut. DOI: 10.64898/2026.03.18.712547. URL: http://biorxiv.org/lookup/doi/10.64898/2026.03.18.712547 (visited on 03/20/2026). Pre-published (cit. on pp. 9, 119).

Aviram, N., Thornal, A. N., Zeevi, D., and Marraffini, L. A. (2022). Different Modes of Spacer Acquisition by the *Staphylococcus Epidermidis* Type III-A CRISPR-Cas System. In: *Nucleic Acids Research* 50.3, pp. 1661–1672. DOI: 10.1093/nar/gkab1299 (cit. on p. 20).

Babu, M. et al. (2011). A Dual Function of the CRISPR–Cas System in Bacterial Antivirus Immunity and DNA Repair. In: *Molecular Microbiology* 79.2, pp. 484–502. DOI: 10.1111/j.1365-2958.2010.07465.x (cit. on p. 29).

Baharoglu, Z., Bikard, D., and Mazel, D. (2010). Conjugative DNA Transfer Induces the Bacterial SOS Response and Promotes Antibiotic Resistance Development through Integron Activation. In: *PLoS Genetics* 6.10. Ed. by I. Matic, e1001165. DOI: 10.1371/journal.pgen.1001165 (cit. on p. 125).

Bar-On, Y. M., Phillips, R., and Milo, R. (2018). The Biomass Distribution on Earth. In: *Proceedings of the National Academy of Sciences* 115.25, pp. 6506–6511. DOI: 10.1073/pnas.1711842115 (cit. on p. 2).

Barrangou, R. and Dudley, E. G. (2016). CRISPR-Based Typing and Next-Generation Tracking Technologies. In: *Annual Review of Food Science and Technology* 7.1, pp. 395–411. DOI: 10.1146/annurev-food-022814-015729 (cit. on pp. 56, 79, 120).

Barrangou, R. and Van Der Oost, J., eds. (2013). CRISPR-Cas Systems: RNA-Mediated Adaptive Immunity in Bacteria and Archaea. Berlin, Heidelberg: Springer Berlin Heidelberg. ISBN: 978-3-642-34656-9 978-3-642-34657-6. DOI: 10.1007/978-3-642-34657-6 (cit. on pp. 6, 7, 13).

Barrangou, R. et al. (2007). CRISPR Provides Acquired Resistance Against Viruses in Prokaryotes. In: *Science* 315.5819, pp. 1709–1712. DOI: 10.1126/science.1138140 (cit. on pp. 7, 14, 19, 21, 60, 64, 86).

Baumdicker, F., Huebner, A., and Pfaffelhuber, P. (2018). The Independent Loss Model with Ordered Insertions for the Evolution of CRISPR Spacers. In: *Theoretical Population Biology* 119, pp. 72–82. DOI: 10.1016/j.tpb.2017.11.001 (cit. on pp. 31, 38–41, 51, 52, 59, 60, 76).

Beauruelle, C., Pastuszka, A., Mereghetti, L., and Lanotte, P. (2018). Group B Streptococcus Vaginal Carriage in Pregnant Women as Deciphered by Clustered Regularly Interspaced Short Palindromic Repeat Analysis. In: *Journal of Clinical Microbiology* 56.6. Ed. by A. Mellmann, e01949–17. DOI: 10.1128/JCM.01949-17 (cit. on pp. 31, 120).

Beauruelle, C. et al. (2021). CRISPR Typing Increases the Discriminatory Power of Streptococcus Agalactiae Typing Methods. In: *Frontiers in Microbiology* 12, p. 675597. DOI: 10.3389/fmicb.2021.675597 (cit. on pp. 31, 120).

Benson, D. A. et al. (2012). GenBank. In: *Nucleic Acids Research* 41.D1, pp. D36–D42. DOI: 10.1093/nar/gks1195 (cit. on p. 17).

Benz, F. et al. (2025). CRISPR–Cas Therapies Targeting Bacteria. In: *Nature Reviews Bioengineering*. DOI: 10.1038/s44222-025-00311-8 (cit. on pp. 9, 120).

Bernheim, A., Bikard, D., Touchon, M., and Rocha, E. P. C. (2019). Atypical Organizations and Epistatic Interactions of CRISPRs and Cas Clusters in Genomes and Their Mobile Genetic Elements. In: *Nucleic Acids Research*, gkz1091. DOI: 10.1093/nar/gkz1091 (cit. on pp. 100, 104, 105, 116, 117).

Bhoobalan-Chitty, Y., Johansen, T. B., Di Cianni, N., and Peng, X. (2019). Inhibition of Type III CRISPR-Cas Immunity by an Archaeal Virus-Encoded Anti-CRISPR Protein. In: *Cell* 179.2, 448–458.e11. DOI: 10.1016/j.cell.2019.09.003 (cit. on p. 26).

Bikard, D., Hatoum-Aslan, A., Mucida, D., and Marraffini, L. A. (2012). CRISPR Interference Can Prevent Natural Transformation and Virulence Acquisition during In Vivo Bacterial Infection. In: *Cell Host & Microbe* 12.2, pp. 177–186. DOI: 10.1016/j.chom.2012.06.003 (cit. on p. 28).

Biswas, A., Fineran, P. C., and Brown, C. M. (2014). Accurate Computational Prediction of the Transcribed Strand of CRISPR Non-Coding RNAs. In: *Bioinformatics* 30.13, pp. 1805–1813. DOI: 10.1093/bioinformatics/btu114 (cit. on pp. 83, 84).

Biswas, A., Staals, R. H., Morales, S. E., Fineran, P. C., and Brown, C. M. (2016). CRISPRDetect: A Flexible Algorithm to Define CRISPR Arrays. In: *BMC Genomics* 17.1, p. 356. DOI: 10.1186/s12864-016-2627-0 (cit. on pp. 16, 83, 84, 93).

Bland, C. et al. (2007). CRISPR Recognition Tool (CRT): A Tool for Automatic Detection of Clustered Regularly Interspaced Palindromic Repeats. In: *BMC Bioinformatics* 8.1, p. 209. DOI: 10.1186/1471-2105-8-209 (cit. on p. 16).

Blombach, F. et al. (2024). Cbp1 and Cren7 Form Chromatin-like Structures That Ensure Efficient Transcription of Long CRISPR Arrays. In: *Nature Communications* 15.1, p. 1620. DOI: 10.1038/s41467-024-45728-8 (cit. on p. 97).

Blosser, T. R. et al. (2015). Two Distinct DNA Binding Modes Guide Dual Roles of a CRISPR-Cas Protein Complex. In: *Molecular Cell* 58.1, pp. 60–70. DOI: 10.1016/j.molcel.2015.01.028 (cit. on p. 23).

Bolotin, A., Quinquis, B., Sorokin, A., and Ehrlich, S. D. (2005). Clustered Regularly Interspaced Short Palindrome Repeats (CRISPRs) Have Spacers of Extrachromosomal Origin. In: *Microbiology* 151.8, pp. 2551–2561. DOI: 10.1099/mic.0.28048-0 (cit. on pp. 7, 33, 35).

Bondy-Denomy, J., Pawluk, A., Maxwell, K. L., and Davidson, A. R. (2013). Bacteriophage Genes That Inactivate the CRISPR/Cas Bacterial Immune System. In: *Nature* 493.7432, pp. 429–432. DOI: 10.1038/nature11723 (cit. on p. 26).

Bonsma-Fisher, M. and Goyal, S. (2023). Dynamics of Immune Memory and Learning in Bacterial Communities. In: *eLife* 12, e81692. DOI: 10.7554/eLife.81692 (cit. on pp. 13, 25–27).

Borges, A. L., Zhang, J. Y., Rollins, M. F., Osuna, B. A., Wiedenheft, B., and Bondy-Denomy, J. (2018). Bacteriophage Cooperation Suppresses CRISPR-Cas3 and Cas9 Immunity. In: *Cell* 174.4, 917–925.e10. DOI: 10.1016/j.cell.2018.06.013 (cit. on p. 26).

Boucher, Y. et al. (2006). Recovery and Evolutionary Analysis of Complete Integron Gene Cassette Arrays from Vibrio. In: *BMC Evolutionary Biology* 6.1, p. 3. DOI: 10.1186/1471-2148-6-3 (cit. on p. 125).

Bradde, S., Mora, T., and Walczak, A. M. (2019). Cost and Benefits of Clustered Regularly Interspaced Short Palindromic Repeats Spacer Acquisition. In: *Philosophical Transactions of the Royal Society B: Biological Sciences* 374.1772, p. 20180095. DOI: 10.1098/rstb.2018.0095 (cit. on p. 25).

Bradde, S., Nourmohammad, A., Goyal, S., and Balasubramanian, V. (2020). The Size of the Immune Repertoire of Bacteria. In: *Proceedings of the National Academy of Sciences* 117.10, pp. 5144–5151. DOI: 10.1073/pnas.1903666117 (cit. on pp. 28, 34).

Bradde, S., Vucelja, M., Teşileanu, T., and Balasubramanian, V. (2017). Dynamics of Adaptive Immunity against Phage in Bacterial Populations. In: *PLOS Computational Biology* 13.4. Ed. by N. L. Komarova, e1005486. DOI: 10.1371/journal.pcbi.1005486 (cit. on p. 25).

Brouns, S. J. J. et al. (2008). Small CRISPR RNAs Guide Antiviral Defense in Prokaryotes. In: *Science* 321.5891, pp. 960–964. DOI: 10.1126/science.1159689 (cit. on pp. 7, 19, 22).

Buongermino Pereira, M., Österlund, T., Eriksson, K. M., Backhaus, T., Axelson-Fisk, M., and Kristiansson, E. (2020). A Comprehensive Survey of Integron-Associated Genes Present in Metagenomes. In: *BMC Genomics* 21.1, p. 495. DOI: 10.1186/s12864-020-06830-5 (cit. on p. 125).

Buyukyoruk, M., Henriques, W. S., and Wiedenheft, B. (2023). Clarifying CRISPR: Why Repeats Identified in the Human Genome Should Not Be Considered CRISPRs. In: *The CRISPR Journal* 6.3, pp. 216–221. DOI: 10.1089/crispr.2022.0106 (cit. on pp. 17, 111, 117, 124).

Buyukyoruk, M., Krishna, P., Santiago-Frangos, A., and Wiedenheft, B. (2025). Discovery of Diverse CRISPR Leader Motifs, Putative Functions, and Applications for Enhanced CRISPR Detection and Subtype Annotation. In: *The CRISPR Journal*, crispr.2024.0093. DOI: 10.1089/crispr.2024.0093 (cit. on pp. 14, 16, 17, 82, 100).

Bzymek, M. and Lovett, S. T. (2001). Instability of Repetitive DNA Sequences: The Role of Replication in Multiple Mechanisms. In: *Proceedings of the National Academy of Sciences* 98.15, pp. 8319–8325. DOI: 10.1073/pnas.111008398 (cit. on pp. 34, 61).

Cady, K. C., Bondy-Denomy, J., Heussler, G. E., Davidson, A. R., and O'Toole, G. A. (2012). The CRISPR/Cas Adaptive Immune System of Pseudomonas Aeruginosa Mediates Resistance to Naturally Occurring and Engineered Phages. In: *Journal of Bacteriology* 194.21, pp. 5728–5738. DOI: 10.1128/JB.01184-12 (cit. on pp. 19, 26).

Camarillo-Guerrero, L. F., Almeida, A., Rangel-Pineros, G., Finn, R. D., and Lawley, T. D. (2021). Massive Expansion of Human Gut Bacteriophage Diversity. In: *Cell* 184.4, 1098–1109.e9. DOI: 10.1016/j.cell.2021.01.029 (cit. on pp. 31, 119, 120).

Cañez, C., Selle, K., Goh, Y. J., and Barrangou, R. (2019). Outcomes and Characterization of Chromosomal Self-Targeting by Native CRISPR-Cas Systems in *Streptococcus Thermophilus*. In: *FEMS Microbiology Letters* 366.9, fnz105. DOI: 10.1093/femsle/fnz105 (cit. on p. 36).

Carte, J., Pfister, N. T., Compton, M. M., Terns, R. M., and Terns, M. P. (2010). Binding and Cleavage of CRISPR RNA by Cas6. In: *RNA* 16.11, pp. 2181–2188. DOI: 10.1261/rna.2230110 (cit. on p. 22).

Carte, J., Wang, R., Li, H., Terns, R. M., and Terns, M. P. (2008). Cas6 Is an Endoribonuclease That Generates Guide RNAs for Invader Defense in Prokaryotes. In: *Genes & Development* 22.24, pp. 3489–3496. DOI: 10.1101/gad.1742908 (cit. on p. 22).

Chabas, H., Müller, V., Bonhoeffer, S., and Regoes, R. R. (2022). Epidemiological and Evolutionary Consequences of Different Types of CRISPR-Cas Systems. In: *PLOS Computational Biology* 18.7. Ed. by A. L. Hill, e1010329. DOI: 10.1371/journal.pcbi.1010329 (cit. on p. 27).

Chabas, H. et al. (2018). Evolutionary Emergence of Infectious Diseases in Heterogeneous Host Populations. In: *PLOS Biology* 16.9. Ed. by J. Drake, e2006738. DOI: 10.1371/journal.pbio.2006738 (cit. on p. 25).

Charpentier, E., Richter, H., Van Der Oost, J., and White, M. F. (2015). Biogenesis Pathways of RNA Guides in Archaeal and Bacterial CRISPR-Cas Adaptive Immunity. In: *FEMS Microbiology Reviews* 39.3, pp. 428–441. DOI: 10.1093/femsre/fuv023 (cit. on p. 22).

Chen, H., Mayer, A., and Balasubramanian, V. (2022). A Scaling Law in CRISPR Repertoire Sizes Arises from the Avoidance of Autoimmunity. In: *Current Biology* 32.13, 2897–2907.e5. DOI: 10.1016/j.cub.2022.05.021 (cit. on pp. 27, 28, 31, 34, 56, 71, 76).

Childs, L. M., England, W. E., Young, M. J., Weitz, J. S., and Whitaker, R. J. (2014). CRISPR-Induced Distributed Immunity in Microbial Populations. In: *PLoS ONE* 9.7. Ed. by W. E. Johnson, e101710. DOI: 10.1371/journal.pone.0101710 (cit. on p. 25).

Childs, L. M., Held, N. L., Young, M. J., Whitaker, R. J., and Weitz, J. S. (2012). Multiscale Model of CRISPR-Induced Coevolutionary Dynamics: Diversification at the Interface of Lamarck and Darwin. In: *Evolution* 66.7, pp. 2015–2029. DOI: 10.1111/j.1558-5646.2012.01595.x (cit. on p. 25).

Claverie, J.-M. and Abergel, C. (2016). CRISPR-Cas-like System in Giant Viruses: Why MIMIVIRE Is Not Likely to Be an Adaptive Immune System. In: *Virologica Sinica* 31.3, pp. 193–196. DOI: 10.1007/s12250-016-3801-x (cit. on p. 124).

Collias, D. and Beisel, C. L. (2021). CRISPR Technologies and the Search for the PAM-Free Nuclease. In: *Nature Communications* 12.1, p. 555. DOI: 10.1038/s41467-020-20633-y (cit. on p. 96).

Collins, A. J. and Whitaker, R. J. (2023). CRISPR Comparison Toolkit: Rapid Identification, Visualization, and Analysis of CRISPR Array Diversity. In: *The CRISPR Journal* 6.4, pp. 386–400. DOI: 10.1089/crispr.2022.0080 (cit. on pp. 31, 37, 51, 52, 71, 76, 96).

Collis, C. M. and Hall, R. M. (1995). Expression of Antibiotic Resistance Genes in the Integrated Cassettes of Integrons. In: *Antimicrobial Agents and Chemotherapy* 39.1, pp. 155–162. DOI: 10.1128/AAC.39.1.155 (cit. on p. 125).

Common, J., Sünderhauf, D., Van Houte, S., and Westra, E. R. (2020). Diversity in CRISPR-Based Immunity Protects Susceptible Genotypes by Restricting Phage Spread and Evolution. In: *Journal of Evolutionary Biology* 33.8, pp. 1097–1108. DOI: 10.1111/jeb.13638 (cit. on p. 25).

Cong, L. et al. (2013). Multiplex Genome Engineering Using CRISPR/Cas Systems. In: *Science* 339.6121, pp. 819–823. DOI: 10.1126/science.1231143 (cit. on p. 8).

Couvin, D. et al. (2018). CRISPRCasFinder, an Update of CRISRFinder, Includes a Portable Version, Enhanced Performance and Integrates Search for Cas Proteins. In: *Nucleic Acids Research* 46.W1, W246–W251. DOI: 10.1093/nar/gky425 (cit. on pp. 16, 57, 83, 84, 93, 96).

Crawley, A. B., Henriksen, J. R., and Barrangou, R. (2018). CRISPRdisco: An Automated Pipeline for the Discovery and Analysis of CRISPR-Cas Systems. In: *The CRISPR Journal* 1.2, pp. 171–181. DOI: 10.1089/crispr.2017.0022 (cit. on p. 16).

Danovaro, R. et al. (2011). Marine Viruses and Global Climate Change. In: *FEMS Microbiology Reviews* 35.6, pp. 993–1034. DOI: 10.1111/j.1574-6976.2010.00258.x (cit. on p. 3).

Darracq, B. et al. (2025). Sedentary Chromosomal Integrons as Biobanks of Bacterial Antiphage Defense Systems. In: *Science* 388.6747, eads0768. DOI: 10.1126/science.ads0768 (cit. on p. 125).

Datsenko, K. A., Pougach, K., Tikhonov, A., Wanner, B. L., Severinov, K., and Semenova, E. (2012). Molecular Memory of Prior Infections Activates the CRISPR/Cas Adaptive Bacterial Immunity System. In: *Nature Communications* 3.1, p. 945. DOI: 10.1038/ncomms1937 (cit. on pp. 18, 20).

Dedrick, R. M. et al. (2019). Engineered Bacteriophages for Treatment of a Patient with a Disseminated Drug-Resistant Mycobacterium Abscessus. In: *Nature Medicine* 25.5, pp. 730–733. DOI: 10.1038/s41591-019-0437-z (cit. on pp. 9, 120).

Deecker, S. R. and Ensminger, A. W. (2020). Type I-F CRISPR-Cas Distribution and Array Dynamics in Legionella Pneumophila. In: *G3 Genes—Genomes—Genetics* 10.3, pp. 1039–1050. DOI: 10.1534/g3.119.400813 (cit. on pp. 19, 20, 31, 33, 35, 36, 100).

Deltcheva, E. et al. (2011). CRISPR RNA Maturation by Trans-Encoded Small RNA and Host Factor RNase III. In: *Nature* 471.7340, pp. 602–607. DOI: 10.1038/nature09886 (cit. on pp. 8, 22).

Deng, L., Garrett, R. A., Shah, S. A., Peng, X., and She, Q. (2013). A Novel Interference Mechanism by a Type IIIB CRISPR-Cmr Module in *Sulfolobus*. In: *Molecular Microbiology* 87.5, pp. 1088–1099. DOI: 10.1111/mmi.12152 (cit. on pp. 20, 22, 23, 26, 29, 115, 117, 123).

Deng, Y. et al. (2015). Resistance Integrons: Class 1, 2 and 3 Integrons. In: *Annals of Clinical Microbiology and Antimicrobials* 14.1, p. 45. DOI: 10.1186/s12941-015-0100-6 (cit. on p. 125).

Deveau, H. et al. (2008). Phage Response to CRISPR-Encoded Resistance in Streptococcus Thermophilus. In: *Journal of Bacteriology* 190.4, pp. 1390–1400. DOI: 10.1128/jb.01412-07 (cit. on pp. 19, 26, 31, 34).

Ding, T., Huang, C., Liang, Z., Ma, X., Wang, N., and Huo, Y.-X. (2020). Reversed Paired-gRNA Plasmid Cloning Strategy for Efficient Genome Editing in Escherichia Coli. In: *Microbial Cell Factories* 19.1, p. 63. DOI: 10.1186/s12934-020-01321-4 (cit. on pp. 34, 35, 122).

Ding, W., Baumdicker, F., and Neher, R. A. (2018). panX: Pan-Genome Analysis and Exploration. In: *Nucleic Acids Research* 46.1, e5–e5. DOI: 10.1093/nar/gkx977 (cit. on pp. 59, 89).

Dion, M., Labrie, S., Shah, S., and Moineau, S. (2018). CRISPRStudio: A User-Friendly Software for Rapid CRISPR Array Visualization. In: *Viruses* 10.11, p. 602. DOI: 10.3390/v10110602 (cit. on pp. 31, 37).

Dion, M. B., Plante, P.-L., Zufferey, E., Shah, S. A., Corbeil, J., and Moineau, S. (2021). Streamlining CRISPR Spacer-Based Bacterial Host Predictions to Decipher the Viral Dark Matter. In: *Nucleic Acids Research* 49.6, pp. 3127–3138. DOI: 10.1093/nar/gkab133 (cit. on pp. 21, 31, 120).

Dong, L. et al. (2019). An Anti-CRISPR Protein Disables Type V Cas12a by Acetylation. In: *Nature Structural & Molecular Biology* 26.4, pp. 308–314. DOI: 10.1038/s41594-019-0206-1 (cit. on p. 26).

Dugar, G. et al. (2013). High-Resolution Transcriptome Maps Reveal Strain-Specific Regulatory Features of Multiple Campylobacter Jejuni Isolates. In: *PLoS Genetics* 9.5. Ed. by D. Hughes, e1003495. DOI: 10.1371/journal.pgen.1003495 (cit. on pp. 22, 33).

Dugar, G. et al. (2018). CRISPR RNA-Dependent Binding and Cleavage of Endogenous RNAs by the Campylobacter Jejuni Cas9. In: *Molecular Cell* 69.5, 893–905.e7. DOI: 10.1016/j.molcel.2018.01.032 (cit. on pp. 22, 33, 84, 98, 113).

Dy, R. L., Richter, C., Salmond, G. P., and Fineran, P. C. (2014). Remarkable Mechanisms in Microbes to Resist Phage Infections. In: *Annual Review of Virology* 1.1, pp. 307–331. DOI: 10.1146/annurev-virology-031413-085500 (cit. on pp. 3, 4).

East-Seletsky, A., O'Connell, M. R., Burstein, D., Knott, G. J., and Doudna, J. A. (2017). RNA Targeting by Functionally Orthogonal Type VI-A CRISPR-Cas Enzymes. In: *Molecular Cell* 66.3, 373–383.e3. DOI: 10.1016/j.molcel.2017.04.008 (cit. on p. 24).

Edgar, R. C. (2007). PILER-CR: Fast and Accurate Identification of CRISPR Repeats. In: *BMC Bioinformatics* 8.1, p. 18. DOI: 10.1186/1471-2105-8-18 (cit. on p. 16).

Elmore, J. R. et al. (2013). Programmable Plasmid Interference by the CRISPR-Cas System in Thermococcus Kodakarensis. In: *RNA Biology* 10.5, pp. 828–840. DOI: 10.4161/rna.24084 (cit. on p. 21).

Elmore, J. R. et al. (2016). Bipartite Recognition of Target RNAs Activates DNA Cleavage by the Type III-B CRISPR–Cas System. In: *Genes & Development* 30.4, pp. 447–459. DOI: 10.1101/gad.272153.115 (cit. on pp. 20, 26).

Enright, M. C., Day, N. P. J., Davies, C. E., Peacock, S. J., and Spratt, B. G. (2000). Multilocus Sequence Typing for Characterization of Methicillin-Resistant and Methicillin-Susceptible Clones of *Staphylococcus Aureus*. In: *Journal of Clinical Microbiology* 38.3, pp. 1008–1015. DOI: 10.1128/JCM.38.3.1008-1015.2000 (cit. on p. 79).

Erdmann, S. and Garrett, R. A. (2012). Selective and Hyperactive Uptake of Foreign DNA by Adaptive Immune Systems of an Archaeon via Two Distinct Mechanisms. In: *Molecular Microbiology* 85.6, pp. 1044–1056. DOI: 10.1111/j.1365-2958.2012.08171.x (cit. on pp. 19, 21, 33, 96, 97).

Escudero, J. A., Loot, C., and Mazel, D. (2018). Integrons as Adaptive Devices. In: *Molecular Mechanisms of Microbial Evolution*. Ed. by P. H. Rampelotto. Cham: Springer International Publishing, pp. 199–239. DOI: 10.1007/978-3-319-69078-0_9 (cit. on pp. 124, 125).

Faure, G. et al. (2019). CRISPR–Cas in Mobile Genetic Elements: Counter-Defence and Beyond. In: *Nature Reviews Microbiology* 17.8, pp. 513–525. DOI: 10.1038/s41579-019-0204-7 (cit. on pp. 3, 24, 27, 29).

Fehrenbach, A., Mitrofanov, A., Alkhnbashi, O. S., Backofen, R., and Baumdicker, F. (2024). SpacerPlacer: Ancestral Reconstruction of CRISPR Arrays Reveals the Evolutionary Dynamics of Spacer Deletions. In: *Nucleic Acids Research*, gkae772. DOI: 10.1093/nar/gkae772 (cit. on pp. 10, 18, 37, 38, 45, 48–50, 52, 55, 88, 89, 100).

– (2025). An Evolutionary Approach to Predict the Orientation of CRISPR Arrays. In: *bioRxiv*. DOI: 10.1101/2025.05.09.653049 (cit. on pp. 10, 81, 117).

Fehrenbach, A., Mitrofanov, A., Backofen, R., and Baumdicker, F. (2026). The Complexity of Multiple CRISPR Arrays in Strains with (Co-Occurring) CRISPR Systems. In: *microLife* 7. Ed. by H. Wolfgang, uqaf042. DOI: 10.1093/femsml/uqaf042 (cit. on pp. 10, 14, 17, 99).

Felsenstein, J. (1992). Phylogenies from Restriction Sites: A Maximum-Likelihood Approach. In: *Evolution* 46.1, pp. 159–173. DOI: 10.1111/j.1558-5646.1992.tb01991.x (cit. on pp. 49, 67).

– (2004). Inferring Phylogenies. Sunderland, Mass: Sinauer Associates. 664 pp. ISBN: 978-0-87893-177-4 (cit. on p. 38).

Feussner, M. et al. (2025). Disparate Mechanisms Counteract Extraneous CRISPR RNA Production in Type II-C CRISPR-Cas Systems. In: *microLife* 6, uqaf007. DOI: 10.1093/femsml/uqaf007 (cit. on pp. 22, 33, 110, 113, 117).

Fillol-Salom, A. et al. (2022). Bacteriophages Benefit from Mobilizing Pathogenicity Islands Encoding Immune Systems against Competitors. In: *Cell* 185.17, 3248–3262.e20. DOI: 10.1016/j.cell.2022.07.014 (cit. on p. 3).

Fineran, P. C. et al. (2014). Degenerate Target Sites Mediate Rapid Primed CRISPR Adaptation. In: *Proceedings of the National Academy of Sciences* 111.16. DOI: 10.1073/pnas.1400071111 (cit. on pp. 20, 26).

Foss, D. V., Hochstrasser, M. L., and Wilson, R. C. (2019). Clinical Applications of CRISPR-based Genome Editing and Diagnostics. In: *Transfusion* 59.4, pp. 1389–1399. DOI: 10.1111/trf.15126 (cit. on p. 8).

Fusco, S., Liguori, R., Limauro, D., Bartolucci, S., She, Q., and Contursi, P. (2015). Transcriptome Analysis of Sulfolobus Solfataricus Infected with Two Related Fuselloviruses Reveals Novel Insights into the Regulation of CRISPR-Cas System. In: *Biochimie* 118, pp. 322–332. DOI: 10.1016/j.biochi.2015.04.006 (cit. on p. 33).

Ganguly, C., Rostami, S., Long, K., Aribam, S. D., and Rajan, R. (2024). Unity among the Diverse RNA-Guided CRISPR-Cas Interference Mechanisms. In: *Journal of Biological Chemistry* 300.6, p. 107295. DOI: 10.1016/j.jbc.2024.107295 (cit. on p. 19).

Ganote, C. L., Caesar, L., Rice, D. W., Whitaker, R. J., and Newton, I. L. G. (2025). Evolutionary Trends in *Bombella Apis* CRISPR-Cas Systems. In: *mSystems* 10.7. Ed. by J. A. Gilbert, e00166–25. DOI: 10.1128/msystems.00166-25 (cit. on pp. 9, 119).

Gansner, E. R. and North, S. C. (2000). An Open Graph Visualization System and Its Applications to Software Engineering. In: *Software: Practice and Experience* 30.11, pp. 1203–1233. DOI: 10.1002/1097-024X(200009)30:11<1203::AID-SPE338>3.0.CO;2-N (cit. on p. 64).

Garneau, J. E. et al. (2010). The CRISPR/Cas Bacterial Immune System Cleaves Bacteriophage and Plasmid DNA. In: *Nature* 468.7320, pp. 67–71. DOI: 10.1038/nature09523 (cit. on pp. 7, 23).

Garrett, R. et al. (2015). CRISPR-Cas Adaptive Immune Systems of the Sulfolobales: Unravelling Their Complexity and Diversity. In: *Life* 5.1, pp. 783–817. DOI: 10.3390/life5010783 (cit. on pp. 21, 33, 96, 97).

Garrett, S., Shiimori, M., Watts, E. A., Clark, L., Graveley, B. R., and Terns, M. P. (2020). Primed CRISPR DNA Uptake in Pyrococcus Furiosus. In: *Nucleic Acids Research* 48.11, pp. 6120–6135. DOI: 10.1093/nar/gkaa381 (cit. on p. 20).

Garrett, S. C. (2021). Pruning and Tending Immune Memories: Spacer Dynamics in the CRISPR Array. In: *Frontiers in Microbiology* 12, p. 664299. DOI: 10.3389/fmicb.2021.664299 (cit. on pp. 13, 32, 34, 35, 56, 60, 61, 71, 77, 82, 122).

Garside, E. L. et al. (2012). Cas5d Processes Pre-crRNA and Is a Member of a Larger Family of CRISPR RNA Endonucleases. In: *RNA* 18.11, pp. 2020–2028. DOI: 10.1261/rna.033100.112 (cit. on p. 22).

Gasiunas, G., Barrangou, R., Horvath, P., and Siksnys, V. (2012). Cas9–crRNA Ribonucleoprotein Complex Mediates Specific DNA Cleavage for Adaptive Immunity in Bacteria. In: *Proceedings of the National Academy of Sciences* 109.39. DOI: 10.1073/pnas.1208507109 (cit. on pp. 8, 22).

Georjon, H. and Bernheim, A. (2023). The Highly Diverse Antiphage Defence Systems of Bacteria. In: *Nature Reviews Microbiology* 21.10, pp. 686–700. DOI: 10.1038/s41579-023-00934-x (cit. on pp. 3, 4).

Getz, L. J., Fairburn, S. R., Vivian Liu, Y., Qian, A. L., and Maxwell, K. L. (2025). Integrons Are Anti-Phage Defence Libraries in Vibrio Parahaemolyticus. In: *Nature Microbiology* 10.3, pp. 724–733. DOI: 10.1038/s41564-025-01927-7 (cit. on p. 125).

Ghaly, T. M., Geoghegan, J. L., Tetu, S. G., and Gillings, M. R. (2020). The Peril and Promise of Integrons: Beyond Antibiotic Resistance. In: *Trends in Microbiology* 28.6, pp. 455–464. DOI: 10.1016/j.tim.2019.12.002 (cit. on p. 125).

Ghaly, T. M., Gillings, M. R., Penesyan, A., Qi, Q., Rajabal, V., and Tetu, S. G. (2021a). The Natural History of Integrons. In: *Microorganisms* 9.11, p. 2212. DOI: 10.3390/microorganisms9112212 (cit. on p. 125).

Ghaly, T. M., Tetu, S. G., and Gillings, M. R. (2021b). Predicting the Taxonomic and Environmental Sources of Integron Gene Cassettes Using Structural and Sequence Homology of attC Sites. In: *Communications Biology* 4.1, p. 946. DOI: 10.1038/s42003-021-02489-0 (cit. on p. 124).

Goldberg, G. W., Jiang, W., Bikard, D., and Marraffini, L. A. (2014). Conditional Tolerance of Temperate Phages via Transcription-Dependent CRISPR-Cas Targeting. In: *Nature* 514.7524, pp. 633–637. DOI: 10.1038/nature13637 (cit. on pp. 23, 28, 30).

González-Delgado, A., Mestre, M. R., Martínez-Abarca, F., and Toro, N. (2019). Spacer Acquisition from RNA Mediated by a Natural Reverse Transcriptase-Cas1 Fusion Protein Associated with a Type III-D CRISPR–Cas System in Vibrio Vulnificus. In: *Nucleic Acids Research* 47.19, pp. 10202–10211. DOI: 10.1093/nar/gkz746 (cit. on p. 19).

Gophna, U., Kristensen, D. M., Wolf, Y. I., Popa, O., Drevet, C., and Koonin, E. V. (2015). No Evidence of Inhibition of Horizontal Gene Transfer by CRISPR–Cas on Evolutionary Timescales. In: *The ISME Journal* 9.9, pp. 2021–2027. DOI: 10.1038/ismej.2015.20 (cit. on p. 28).

Goren, M. G., Doron, S., Globus, R., Amitai, G., Sorek, R., and Qimron, U. (2016). Repeat Size Determination by Two Molecular Rulers in the Type I-E CRISPR Array. In: *Cell Reports* 16.11, pp. 2811–2818. DOI: 10.1016/j.celrep.2016.08.043 (cit. on p. 21).

Grissa, I., Vergnaud, G., and Pourcel, C. (2007). The CRISPRdb Database and Tools to Display CRISPRs and to Generate Dictionaries of Spacers and Repeats. In: *BMC Bioinformatics* 8.1, p. 172. DOI: 10.1186/1471-2105-8-172 (cit. on p. 17).

Gudbergsdottir, S. et al. (2011). Dynamic Properties of the Sulfolobus CRISPR/Cas and CRISPR/Cmr Systems When Challenged with Vector-Borne Viral and Plasmid Genes and Protospacers: Dynamics of Sulfolobus CRISPR-Based Systems. In: *Molecular Microbiology* 79.1, pp. 35–49. DOI: 10.1111/j.1365-2958.2010.07452.x (cit. on pp. 26, 34).

Guerin, É. et al. (2009). The SOS Response Controls Integron Recombination. In: *Science* 324.5930, pp. 1034–1034. DOI: 10.1126/science.1172914 (cit. on p. 125).

Guerrero, L. D., Pérez, M. V., Orellana, E., Piuri, M., Quiroga, C., and Erijman, L. (2021). Long-Run Bacteria-Phage Coexistence Dynamics under Natural Habitat Conditions in an Environmental Biotechnology System. In: *The ISME Journal* 15.3, pp. 636–648. DOI: 10.1038/s41396-020-00802-z (cit. on pp. 9, 26, 56, 80, 120).

Guillemet, M. et al. (2022). Competition and Coevolution Drive the Evolution and the Diversification of CRISPR Immunity. In: *Nature Ecology & Evolution* 6.10, pp. 1480–1488. DOI: 10.1038/s41559-022-01841-9 (cit. on p. 25).

Haft, D. H., Selengut, J., Mongodin, E. F., and Nelson, K. E. (2005). A Guild of 45 CRISPR-Associated (Cas) Protein Families and Multiple CRISPR/Cas Subtypes Exist in Prokaryotic Genomes. In: *PLoS Computational Biology* 1.6. Ed. by J. A. Eisen, e60. DOI: 10.1371/journal.pcbi.0010060 (cit. on p. 15).

Hampton, H. G., Watson, B. N. J., and Fineran, P. C. (2020). The Arms Race between Bacteria and Their Phage Foes. In: *Nature* 577.7790, pp. 327–336. DOI: 10.1038/s41586-019-1894-8 (cit. on pp. 3, 4).

Han, P. and Deem, M. W. (2017). Non-Classical Phase Diagram for Virus Bacterial Coevolution Mediated by Clustered Regularly Interspaced Short Palindromic Repeats. In: *Journal of The Royal Society Interface* 14.127, p. 20160905. DOI: 10.1098/rsif.2016.0905 (cit. on p. 25).

Han, P., Niestemski, L. R., Barrick, J. E., and Deem, M. W. (2013). Physical Model of the Immune Response of Bacteria against Bacteriophage through the Adaptive CRISPR-Cas Immune System. In: *Physical Biology* 10.2, p. 025004. DOI: 10.1088/1478-3975/10/2/025004 (cit. on pp. 25, 31, 71, 76).

Hargreaves, K. R., Flores, C. O., Lawley, T. D., and Clokie, M. R. J. (2014). Abundant and Diverse Clustered Regularly Interspaced Short Palindromic Repeat Spacers in Clostridium Difficile Strains and Prophages Target Multiple Phage Types within This Pathogen. In: *mBio* 5.5. Ed. by J. Paul and F. Azam, e01045–13. DOI: 10.1128/mBio.01045-13 (cit. on pp. 3, 26, 27, 29).

Hatoum-Aslan, A., Maniv, I., and Marraffini, L. A. (2011). Mature Clustered, Regularly Interspaced, Short Palindromic Repeats RNA (crRNA) Length Is Measured by a Ruler Mechanism Anchored at the Precursor Processing Site. In: *Proceedings of the National Academy of Sciences* 108.52, pp. 21218–21222. DOI: 10.1073/pnas.1112832108 (cit. on p. 22).

Hayes, R. P. et al. (2016). Structural Basis for Promiscuous PAM Recognition in Type I–E Cascade from E. Coli. In: *Nature* 530.7591, pp. 499–503. DOI: 10.1038/nature16995 (cit. on p. 23).

He, F., Vestergaard, G., Peng, W., She, Q., and Peng, X. (2016). CRISPR-Cas Type I-A Cascade Complex Couples Viral Infection Surveillance to Host Transcriptional Regulation in the Dependence of Csa3b. In: *Nucleic Acids Research*, gkw1265. DOI: 10.1093/nar/gkw1265 (cit. on p. 33).

He, J. and Deem, M. W. (2010). Heterogeneous Diversity of Spacers within CRISPR (Clustered Regularly Interspaced Short Palindromic Repeats). In: *Physical Review Letters* 105.12, p. 128102. DOI: 10.1103/PhysRevLett.105.128102 (cit. on pp. 25, 31, 51).

He, Y. et al. (2018). Cas1 and Cas2 From the Type II-C CRISPR-Cas System of Riemerella Anatipestifer Are Required for Spacer Acquisition. In: *Frontiers in Cellular and Infection Microbiology* 8, p. 195. DOI: 10.3389/fcimb.2018.00195 (cit. on p. 18).

Held, N. L., Herrera, A., Cadillo-Quiroz, H., and Whitaker, R. J. (2010). CRISPR Associated Diversity within a Population of Sulfolobus Islandicus. In: *PLoS ONE* 5.9. Ed. by P. J. Planet, e12988. DOI: 10.1371/journal.pone.0012988 (cit. on pp. 31, 33–35).

Heler, R. et al. (2015). Cas9 Specifies Functional Viral Targets during CRISPR–Cas Adaptation. In: *Nature* 519.7542, pp. 199–202. DOI: 10.1038/nature14245 (cit. on pp. 19, 28, 33).

Hendrix, R. W., Smith, M. C. M., Burns, R. N., Ford, M. E., and Hatfull, G. F. (1999). Evolutionary Relationships among Diverse Bacteriophages and Prophages: All the World's a Phage. In: *Proceedings of the National Academy of Sciences* 96.5, pp. 2192–2197. DOI: 10.1073/pnas.96.5.2192 (cit. on p. 2).

Hermans, P. W., Van Soolingen, D., Bik, E. M., De Haas, P. E., Dale, J. W., and Van Embden, J. D. (1991). Insertion Element IS987 from Mycobacterium Bovis BCG Is Located in a Hot-Spot Integration Region for Insertion Elements in Mycobacterium Tuberculosis Complex Strains. In: *Infection and Immunity* 59.8, pp. 2695–2705. DOI: 10.1128/iai.59.8.2695-2705.1991 (cit. on p. 6).

Heussler, G. E., Cady, K. C., Koeppen, K., Bhuju, S., Stanton, B. A., and O'Toole, G. A. (2015). Clustered Regularly Interspaced Short Palindromic Repeat-Dependent, Biofilm-Specific Death of Pseudomonas Aeruginosa Mediated by Increased Expression of Phage-Related Genes. In: *mBio* 6.3. Ed. by V. Sperandio, e00129–15. DOI: 10.1128/mBio.00129-15 (cit. on p. 29).

Hille, F., Richter, H., Wong, S. P., Bratovič, M., Ressel, S., and Charpentier, E. (2018). The Biology of CRISPR-Cas: Backward and Forward. In: *Cell* 172.6, pp. 1239–1259. DOI: 10.1016/j.cell.2017.11.032 (cit. on pp. 13, 14, 18–20, 22–24, 28).

Hoikkala, V. et al. (2021). Cooperation between Different CRISPR-Cas Types Enables Adaptation in an RNA-Targeting System. In: *mBio* 12.2, 10.1128/mbio.03338–20. DOI: 10.1128/mbio.03338-20 (cit. on pp. 19, 20, 22, 29, 33, 84, 98, 111, 113, 115, 117, 123).

Hooton, S. P. T. and Connerton, I. F. (2015). Campylobacter Jejuni Acquire New Host-Derived CRISPR Spacers When in Association with Bacteriophages Harboring a CRISPR-like Cas4 Protein. In: *Frontiers in Microbiology* 5. DOI: 10.3389/fmicb.2014.00744 (cit. on pp. 22, 27, 33, 84).

Horvath, P. et al. (2008). Diversity, Activity, and Evolution of CRISPR Loci in Streptococcus Thermophilus. In: *Journal of Bacteriology*. DOI: 10.1128/jb.01415-07 (cit. on pp. 7, 31, 35, 78).

Houenoussi, K., Boukheloua, R., Vernadet, J.-P., Gautheret, D., Vergnaud, G., and Pourcel, C. (2020). TOP the Transcription Orientation Pipeline and Its Use to Investigate the Transcription of Non-Coding Regions: Assessment with CRISPR Direct Repeats and Intergenic Sequences. DOI: 10.1101/2020.01.15.903914. URL: http://biorxiv.org/lookup/doi/10.1101/2020.01.15.903914 (visited on 07/19/2024). Pre-published (cit. on pp. 82, 83).

Høyland-Kroghsbo, N. M. et al. (2017). Quorum Sensing Controls the *Pseudomonas Aeruginosa* CRISPR-Cas Adaptive Immune System. In: *Proceedings of the National Academy of Sciences* 114.1, pp. 131–135. DOI: 10.1073/pnas.1617415113 (cit. on p. 33).

Høyland-Kroghsbo, N. M., Muñoz, K. A., and Bassler, B. L. (2018). Temperature, by Controlling Growth Rate, Regulates CRISPR-Cas Activity in Pseudomonas Aeruginosa. In: *mBio* 9.6. Ed. by J. Vogel, e02184–18. DOI: 10.1128/mBio.02184-18 (cit. on pp. 13, 33).

Hwang, S. and Maxwell, K. L. (2019). Meet the Anti-CRISPRs: Widespread Protein Inhibitors of CRISPR-Cas Systems. In: *The CRISPR Journal* 2.1, pp. 23–30. DOI: 10.1089/crispr.2018.0052 (cit. on p. 26).

Hyatt, D., Chen, G.-L., LoCascio, P. F., Land, M. L., Larimer, F. W., and Hauser, L. J. (2010). Prodigal: Prokaryotic Gene Recognition and Translation Initiation Site Identification. In: *BMC Bioinformatics* 11.1, p. 119. DOI: 10.1186/1471-2105-11-119 (cit. on p. 16).

Iranzo, J., Lobkovsky, A. E., Wolf, Y. I., and Koonin, E. V. (2013). Evolutionary Dynamics of the Prokaryotic Adaptive Immunity System CRISPR-Cas in an Explicit Ecological Context. In: *Journal of Bacteriology* 195.17, pp. 3834–3844. DOI: 10.1128/JB.00412-13 (cit. on pp. 13, 25–27, 33).

Ishino, Y., Shinagawa, H., Makino, K., Amemura, M., and Nakata, A. (1987). Nucleotide Sequence of the Iap Gene, Responsible for Alkaline Phosphatase Isozyme Conversion in Escherichia Coli, and Identification of the Gene Product. In: *Journal of Bacteriology* 169.12, pp. 5429–5433. DOI: 10.1128/jb.169.12.5429-5433.1987 (cit. on p. 6).

Ishino, Y., Krupovic, M., and Forterre, P. (2018). History of CRISPR-Cas from Encounter with a Mysterious Repeated Sequence to Genome Editing Technology. In: *Journal of Bacteriology* 200.7. Ed. by W. Margolin. DOI: 10.1128/JB.00580-17 (cit. on pp. 6–8).

Jackson, S. A., McKenzie, R. E., Fagerlund, R. D., Kieper, S. N., Fineran, P. C., and Brouns, S. J. J. (2017). CRISPR-Cas: Adapting to Change. In: *Science* 356.6333, eaal5056. DOI: 10.1126/science.aal5056 (cit. on pp. 13, 96).

Jansen, Ruud., Embden, J. D. A. V., Gaastra, Wim., and Schouls, Leo. M. (2002). Identification of Genes That Are Associated with DNA Repeats in Prokaryotes. In: *Molecular Microbiology* 43.6, pp. 1565–1575. DOI: 10.1046/j.1365-2958.2002.02839.x (cit. on pp. 6, 7, 15, 35).

Jiang, W., Maniv, I., Arain, F., Wang, Y., Levin, B. R., and Marraffini, L. A. (2013). Dealing with the Evolutionary Downside of CRISPR Immunity: Bacteria and Beneficial Plasmids. In: *PLoS Genetics* 9.9. Ed. by I. Matic, e1003844. DOI: 10.1371/journal.pgen.1003844 (cit. on pp. 28, 34).

Jiang, W., Samai, P., and Marraffini, L. A. (2016). Degradation of Phage Transcripts by CRISPR-Associated RNases Enables Type III CRISPR-Cas Immunity. In: *Cell* 164.4, pp. 710–721. DOI: 10.1016/j.cell.2015.12.053 (cit. on p. 24).

Jinek, M., Chylinski, K., Fonfara, I., Hauer, M., Doudna, J. A., and Charpentier, E. (2012). A Programmable Dual-RNA–Guided DNA Endonuclease in Adaptive Bacterial Immunity. In: *Science* 337.6096, pp. 816–821. DOI: 10.1126/science.1225829 (cit. on pp. 8, 22, 23).

Karah, N., Samuelsen, Ø., Zarrilli, R., Sahl, J. W., Wai, S. N., and Uhlin, B. E. (2015). CRISPR-Cas Subtype I-Fb in Acinetobacter Baumannii: Evolution and Utilization for Strain Subtyping. In: *PLOS ONE* 10.2, e0118205. DOI: 10.1371/journal.pone.0118205 (cit. on pp. 31, 79).

Karimi, Z., Ahmadi, A., Najafi, A., and Ranjbar, R. (2018). Bacterial CRISPR Regions: General Features and Their Potential for Epidemiological Molecular Typing Studies. In: *The Open Microbiology Journal* 12, pp. 59–70. DOI: 10.2174/1874285801812010059 (cit. on pp. 31, 56, 79).

Karlin, S. and Taylor, H. M. (1975). A First Course in Stochastic Processes. Second. Academic press. ISBN: 978-0-12-398552-1 (cit. on pp. 46, 48).

Katoh, K., Rozewicki, J., and Yamada, K. D. (2019). MAFFT Online Service: Multiple Sequence Alignment, Interactive Sequence Choice and Visualization. In: *Briefings in Bioinformatics* 20.4, pp. 1160–1166. DOI: 10.1093/bib/bbx108 (cit. on p. 58).

Kazlauskiene, M., Kostiuk, G., Venclovas, Č., Tamulaitis, G., and Siksnys, V. (2017). A Cyclic Oligonucleotide Signaling Pathway in Type III CRISPR-Cas Systems. In: *Science* 357.6351, pp. 605–609. DOI: 10.1126/science.aao0100 (cit. on pp. 23, 24).

Kieper, S. N., Almendros, C., and Brouns, S. J. J. (2019). Conserved Motifs in the CRISPR Leader Sequence Control Spacer Acquisition Levels in Type I-D CRISPR-Cas Systems. In: *FEMS Microbiology Letters* 366.11, fnz129. DOI: 10.1093/femsle/fnz129 (cit. on pp. 19, 21).

Kim, J. G., Garrett, S., Wei, Y., Graveley, B. R., and Terns, M. P. (2019). CRISPR DNA Elements Controlling Site-Specific Spacer Integration and Proper Repeat Length by a Type II CRISPR–Cas System. In: *Nucleic Acids Research* 47.16, pp. 8632–8648. DOI: 10.1093/nar/gkz677 (cit. on pp. 14, 21, 110, 117).

Knott, G. J. et al. (2019). Broad-Spectrum Enzymatic Inhibition of CRISPR-Cas12a. In: *Nature Structural & Molecular Biology* 26.4, pp. 315–321. DOI: 10.1038/s41594-019-0208-z (cit. on p. 26).

Komor, A. C., Badran, A. H., and Liu, D. R. (2017). CRISPR-Based Technologies for the Manipulation of Eukaryotic Genomes. In: *Cell* 168.1–2, pp. 20–36. DOI: 10.1016/j.cell.2016.10.044 (cit. on p. 9).

Koonin, E. V., Makarova, K. S., and Zhang, F. (2017). Diversity, Classification and Evolution of CRISPR-Cas Systems. In: *Current Opinion in Microbiology* 37, pp. 67–78. DOI: 10.1016/j.mib.2017.05.008 (cit. on p. 13).

Koonin, E. V. and Wolf, Y. I. (2009). Is Evolution Darwinian or/and Lamarckian? In: *Biology Direct* 4.1, p. 42. DOI: 10.1186/1745-6150-4-42 (cit. on p. 6).

Koonin, E. V., Gootenberg, J. S., and Abudayyeh, O. O. (2023). Discovery of Diverse CRISPR-Cas Systems and Expansion of the Genome Engineering Toolbox. In: *Biochemistry*, acs.biochem.3c00159. DOI: 10.1021/acs.biochem.3c00159 (cit. on p. 9).

Koonin, E. V. and Makarova, K. S. (2024). CRISPR in Mobile Genetic Elements: Counter-Defense, Inter-Element Competition and RNA-Guided Transposition. In: *BMC Biology* 22.1, p. 295. DOI: 10.1186/s12915-024-02090-x (cit. on pp. 3, 24, 27, 29).

Koonin, E. V. and Wolf, Y. I. (2008). Genomics of Bacteria and Archaea: The Emerging Dynamic View of the Prokaryotic World. In: *Nucleic Acids Research* 36.21, pp. 6688–6719. DOI: 10.1093/nar/gkn668. PMID: 18948295 (cit. on p. 2).

– (2015). Evolution of the CRISPR-Cas Adaptive Immunity Systems in Prokaryotes: Models and Observations on Virus–Host Coevolution. In: *Molecular BioSystems* 11.1, pp. 20–27. DOI: 10.1039/C4MB00438H (cit. on p. 76).

Kunin, V., Sorek, R., and Hugenholtz, P. (2007). Evolutionary Conservation of Sequence and Secondary Structures in CRISPR Repeats. In: *Genome Biology* 8.4, R61. DOI: 10.1186/gb-2007-8-4-r61 (cit. on p. 14).

Künne, T. et al. (2016). Cas3-Derived Target DNA Degradation Fragments Fuel Primed CRISPR Adaptation. In: *Molecular Cell* 63.5, pp. 852–864. DOI: 10.1016/j.molcel.2016.07.011 (cit. on p. 21).

Kupczok, A. and Bollback, J. P. (2013). Probabilistic Models for CRISPR Spacer Content Evolution. In: *BMC Evolutionary Biology* 13.1, p. 54. DOI: 10.1186/1471-2148-13-54 (cit. on pp. 31, 38, 51, 52, 59, 60, 71, 73, 76).

Kupczok, A., Landan, G., and Dagan, T. (2015). The Contribution of Genetic Recombination to CRISPR Array Evolution. In: *Genome Biology and Evolution* 7.7, pp. 1925–1939. DOI: 10.1093/gbe/evv113 (cit. on pp. 31, 34, 51, 52, 71, 74).

Laanto, E., Hoikkala, V., Ravantti, J., and Sundberg, L.-R. (2017). Long-Term Genomic Coevolution of Host-Parasite Interaction in the Natural Environment. In: *Nature Communications* 8.1, p. 111. DOI: 10.1038/s41467-017-00158-7 (cit. on pp. 9, 22, 26, 33, 84, 98, 111, 113, 117, 119).

Labbate, M., Boucher, Y., Joss, M. J., Michael, C. A., Gillings, M. R., and Stokes, H. W. (2007). Use of Chromosomal Integron Arrays as a Phylogenetic Typing System for Vibrio Cholerae Pandemic Strains. In: *Microbiology* 153.5, pp. 1488–1498. DOI: 10.1099/mic.0.2006/001065-0 (cit. on p. 125).

Labrie, S. J., Samson, J. E., and Moineau, S. (2010). Bacteriophage Resistance Mechanisms. In: *Nature Reviews Microbiology* 8.5, pp. 317–327. DOI: 10.1038/nrmicro2315 (cit. on pp. 3, 4).

Lam, T. J. and Ye, Y. (2019a). CRISPRs for Strain Tracking and Their Application to Microbiota Transplantation Data Analysis. In: *The CRISPR Journal* 2.1, pp. 41–50. DOI: 10.1089/crispr.2018.0046 (cit. on pp. 9, 31, 56, 120).

– (2019b). Long Reads Reveal the Diversification and Dynamics of CRISPR Reservoir in Microbiomes. In: *BMC Genomics* 20.1, p. 567. DOI: 10.1186/s12864-019-5922-8 (cit. on pp. 31, 33, 56, 80, 119).

Lander, E. S. (2016). The Heroes of CRISPR. In: *Cell* 164.1–2, pp. 18–28. DOI: 10.1016/j.cell.2015.12.041 (cit. on pp. 6, 7).

Landor, L. A. I., Ruffo, V., Sundberg, L.-R., Madsen, L., and Middelboe, M. (2026). Phage-Host Interactions in *Flavobacterium* Pathogens and the Potential of Phage-Based Disease Control in Salmonid Aquaculture. In: *Reviews in Aquaculture* 18.1, e70115. DOI: 10.1111/raq.70115 (cit. on pp. 9, 119).

Landsberger, M. et al. (2018). Anti-CRISPR Phages Cooperate to Overcome CRISPR-Cas Immunity. In: *Cell* 174.4, 908–916.e12. DOI: 10.1016/j.cell.2018.05.058 (cit. on p. 26).

Lange, S. J., Alkhnbashi, O. S., Rose, D., Will, S., and Backofen, R. (2013). CRISPRmap: An Automated Classification of Repeat Conservation in Prokaryotic Adaptive Immune Systems. In: *Nucleic Acids Research* 41.17, pp. 8034–8044. DOI: 10.1093/nar/gkt606 (cit. on p. 83).

Leenay, R. T. and Beisel, C. L. (2017). Deciphering, Communicating, and Engineering the CRISPR PAM. In: *Journal of Molecular Biology* 429.2, pp. 177–191. DOI: 10.1016/j.jmb.2016.11.024 (cit. on pp. 19, 20, 26).

Levasseur, A., Bekliz, M., Chabrière, E., Pontarotti, P., La Scola, B., and Raoult, D. (2016). MIMIVIRE Is a Defence System in Mimivirus That Confers Resistance to Virophage. In: *Nature* 531.7593, pp. 249–252. DOI: 10.1038/nature17146 (cit. on p. 124).

Levin, B. R., Moineau, S., Bushman, M., and Barrangou, R. (2013). The Population and Evolutionary Dynamics of Phage and Bacteria with CRISPR–Mediated Immunity. In: *PLoS Genetics* 9.3. Ed. by D. Hughes, e1003312. DOI: 10.1371/journal.pgen.1003312 (cit. on pp. 26, 27).

Levy, A. et al. (2015). CRISPR Adaptation Biases Explain Preference for Acquisition of Foreign DNA. In: *Nature* 520.7548, pp. 505–510. DOI: 10.1038/nature14302 (cit. on pp. 19, 20).

Li, C., Wang, Y., Gao, Y., Li, C., Ma, B., and Wang, H. (2021). Antimicrobial Resistance and CRISPR Typing Among Salmonella Isolates From Poultry Farms in China. In: *Frontiers in Microbiology* 12, p. 730046. DOI: 10.3389/fmicb.2021.730046 (cit. on pp. 9, 119).

Li, M., Wang, R., and Xiang, H. (2014). Haloarcula Hispanica CRISPR Authenticates PAM of a Target Sequence to Prime Discriminative Adaptation. In: *Nucleic Acids Research* 42.11, pp. 7226–7235. DOI: 10.1093/nar/gku389 (cit. on p. 20).

Li, R. et al. (2016). Type I CRISPR-Cas Targets Endogenous Genes and Regulates Virulence to Evade Mammalian Host Immunity. In: *Cell Research* 26.12, pp. 1273–1287. DOI: 10.1038/cr.2016.135 (cit. on p. 29).

Liao, C. et al. (2022). Spacer Prioritization in CRISPR-Cas9 Immunity Is Enabled by the Leader RNA. In: *Nature Microbiology* 7.4, pp. 530–541. DOI: 10.1038/s41564-022-01074-3 (cit. on pp. 14, 21, 22, 100, 110, 117).

Lier, C. et al. (2015). Analysis of the Type II-A CRISPR-Cas System of Streptococcus Agalactiae Reveals Distinctive Features According to Genetic Lineages. In: *Frontiers in Genetics* 6. DOI: 10.3389/fgene.2015.00214 (cit. on pp. 31, 33, 35).

Lillestøl, R., Redder, P., Garrett, R. A., and Brügger, K. (2006). A Putative Viral Defence Mechanism in Archaeal Cells. In: *Archaea* 2.1, pp. 59–72. DOI: 10.1155/2006/542818 (cit. on pp. 7, 31, 33–35).

Liu, Y., Botelho, J., and Iranzo, J. (2025). Timescale and Genetic Linkage Explain the Variable Impact of Defense Systems on Horizontal Gene Transfer. In: *Genome Research*, genome, gr.279300.124v2. DOI: 10.1101/gr.279300.124 (cit. on pp. 28, 104, 117).

Loenen, W. A. M., Dryden, D. T. F., Raleigh, E. A., Wilson, G. G., and Murray, N. E. (2014). Highlights of the DNA Cutters: A Short History of the Restriction Enzymes. In: *Nucleic Acids Research* 42.1, pp. 3–19. DOI: 10.1093/nar/gkt990 (cit. on p. 4).

Loot, C., Ducos-Galand, M., Escudero, J. A., Bouvier, M., and Mazel, D. (2012). Replicative Resolution of Integron Cassette Insertion. In: *Nucleic Acids Research* 40.17, pp. 8361–8370. DOI: 10.1093/nar/gks620 (cit. on p. 125).

López-Beltrán, A., Botelho, J., and Iranzo, J. (2024). Dynamics of CRISPR-Mediated Virus-Host Interactions in the Human Gut Microbiome. In: *The ISME Journal*, wrae134. DOI: 10.1093/ismejo/wrae134 (cit. on pp. 9, 56, 78, 98, 100, 120).

Lopez-Sanchez, M.-J. et al. (2012). The Highly Dynamic CRISPR1 System of *Streptococcus Agalactiae* Controls the Diversity of Its Mobilome: *Streptococcus Agalactiae* CRISPR Dynamics. In: *Molecular Microbiology* 85.6, pp. 1057–1071. DOI: 10.1111/j.1365-2958.2012.08172.x (cit. on pp. 31, 33–35, 52, 60, 73, 76, 77).

Louwen, R. et al. (2013). A Novel Link between Campylobacter Jejuni Bacteriophage Defence, Virulence and Guillain–Barré Syndrome. In: *European Journal of Clinical Microbiology & Infectious Diseases* 32.2, pp. 207–226. DOI: 10.1007/s10096-012-1733-4 (cit. on pp. 29, 120).

Lovett, S. T. (2004). Encoded Errors: Mutations and Rearrangements Mediated by Misalignment at Repetitive DNA Sequences. In: *Molecular Microbiology* 52.5, pp. 1243–1253. DOI: 10.1111/j.1365-2958.2004.04076.x (cit. on pp. 34, 35, 61, 77).

Maazou, Y. (2026). "Simulation and Analysis of Mutation Patterns in CRISPR Repeats". BA thesis. University of Tübingen (cit. on p. 122).

Majumdar, S. et al. (2015). Three CRISPR-Cas Immune Effector Complexes Coexist in *Pyrococcus Furiosus*. In: *RNA* 21.6, pp. 1147–1158. DOI: 10.1261/rna.049130.114 (cit. on pp. 29, 30, 115, 117, 123).

Makarova, K. S. (2002). A DNA Repair System Specific for Thermophilic Archaea and Bacteria Predicted by Genomic Context Analysis. In: *Nucleic Acids Research* 30.2, pp. 482–496. DOI: 10.1093/nar/30.2.482 (cit. on p. 7).

Makarova, K. S., Grishin, N. V., Shabalina, S. A., Wolf, Y. I., and Koonin, E. V. (2006). A Putative RNA-Interference-Based Immune System in Prokaryotes: Computational Analysis of the Predicted Enzymatic Machinery, Functional Analogies with Eukaryotic RNAi, and Hypothetical Mechanisms of Action. In: *Biology Direct* 1.1, p. 7. DOI: 10.1186/1745-6150-1-7 (cit. on pp. 7, 15).

Makarova, K. S. et al. (2011). Evolution and Classification of the CRISPR–Cas Systems. In: *Nature Reviews Microbiology* 9.6, pp. 467–477. DOI: 10.1038/nrmicro2577 (cit. on pp. 8, 15, 23).

Makarova, K. S. et al. (2015). An Updated Evolutionary Classification of CRISPR-Cas Systems. In: *Nature reviews. Microbiology* 13.11, pp. 722–736. DOI: 10.1038/nrmicro3569 (cit. on pp. 13, 15, 19, 24, 60).

Makarova, K. S. et al. (2020). Evolutionary Classification of CRISPR–Cas Systems: A Burst of Class 2 and Derived Variants. In: *Nature Reviews Microbiology* 18.2, pp. 67–83. DOI: 10.1038/s41579-019-0299-x (cit. on pp. 15, 16, 23, 24, 29, 60, 96, 98, 100, 113, 115, 117, 123).

Makarova, K. S. et al. (2025). An Updated Evolutionary Classification of CRISPR–Cas Systems Including Rare Variants. In: *Nature Microbiology*. DOI: 10.1038/s41564-025-02180-8 (cit. on pp. 13, 15–19, 23–25, 27, 100, 117).

Mali, P. et al. (2013). RNA-Guided Human Genome Engineering via Cas9. In: *Science* 339.6121, pp. 823–826. DOI: 10.1126/science.1232033 (cit. on p. 8).

Mandin, P., Repoila, F., Vergassola, M., Geissmann, T., and Cossart, P. (2007). Identification of New Noncoding RNAs in Listeria Monocytogenes and Prediction of mRNA Targets. In: *Nucleic Acids Research* 35.3, pp. 962–974. DOI: 10.1093/nar/gkl1096 (cit. on p. 29).

Manica, A., Zebec, Z., Teichmann, D., and Schleper, C. (2011). *In Vivo* Activity of CRISPR-mediated Virus Defence in a Hyperthermophilic Archaeon. In: *Molecular Microbiology* 80.2, pp. 481–491. DOI: 10.1111/j.1365-2958.2011.07586.x (cit. on p. 26).

Manning, A. J. and Kuehn, M. J. (2011). Contribution of Bacterial Outer Membrane Vesicles to Innate Bacterial Defense. In: *BMC Microbiology* 11.1, p. 258. DOI: 10.1186/1471-2180-11-258 (cit. on p. 3).

Margolis, S. R. and Meeske, A. J. (2025). Crosstalk between Three CRISPR-Cas Types Enables Primed Type VI-A Adaptation in Listeria Seeligeri. In: *Cell Host & Microbe* 33.9, 1550–1560.e4. DOI: 10.1016/j.chom.2025.05.020 (cit. on pp. 20, 29, 30, 115–117, 123).

Marraffini, L. A. and Sontheimer, E. J. (2008). CRISPR Interference Limits Horizontal Gene Transfer in Staphylococci by Targeting DNA. In: *Science* 322.5909, pp. 1843–1845. DOI: 10.1126/science.1165771 (cit. on pp. 7, 28).

– (2010). Self versus Non-Self Discrimination during CRISPR RNA-Directed Immunity. In: *Nature* 463.7280, pp. 568–571. DOI: 10.1038/nature08703 (cit. on pp. 20, 26).

Martel, B. and Moineau, S. (2014). CRISPR-Cas: An Efficient Tool for Genome Engineering of Virulent Bacteriophages. In: *Nucleic Acids Research* 42.14, pp. 9504–9513. DOI: 10.1093/nar/gku628 (cit. on p. 26).

Martínez Arbas, S. et al. (2021). Roles of Bacteriophages, Plasmids and CRISPR Immunity in Microbial Community Dynamics Revealed Using Time-Series Integrated Meta-Omics. In: *Nature Microbiology* 6.1, pp. 123–135. DOI: 10.1038/s41564-020-00794-8 (cit. on pp. 9, 56, 80, 120).

Martynov, A., Severinov, K., and Ispolatov, I. (2017). Optimal Number of Spacers in CRISPR Arrays. In: *PLOS Computational Biology* 13.12. Ed. by C. O. Wilke, e1005891. DOI: 10.1371/journal.pcbi.1005891 (cit. on pp. 14, 21, 34, 51, 76, 77, 100, 107, 117).

Mayo-Muñoz, D., Pinilla-Redondo, R., Camara-Wilpert, S., Birkholz, N., and Fineran, P. C. (2024). Inhibitors of Bacterial Immune Systems: Discovery, Mechanisms and Applications. In: *Nature Reviews Genetics* 25.4, pp. 237–254. DOI: 10.1038/s41576-023-00676-9 (cit. on p. 3).

McGinn, J. and Marraffini, L. A. (2016). CRISPR-Cas Systems Optimize Their Immune Response by Specifying the Site of Spacer Integration. In: *Molecular Cell* 64.3, pp. 616–623. DOI: 10.1016/j.molcel.2016.08.038 (cit. on pp. 14, 21, 33, 36, 97, 100).

– (2019). Molecular Mechanisms of CRISPR–Cas Spacer Acquisition. In: *Nature Reviews Microbiology* 17.1, pp. 7–12. DOI: 10.1038/s41579-018-0071-7 (cit. on pp. 13, 18, 20, 21, 86, 120).

Meeske, A. J. and Marraffini, L. A. (2018). RNA Guide Complementarity Prevents Self-Targeting in Type VI CRISPR Systems. In: *Molecular Cell* 71.5, 791–801.e3. DOI: 10.1016/j.molcel.2018.07.013 (cit. on pp. 20, 26).

Meeske, A. J., Nakandakari-Higa, S., and Marraffini, L. A. (2019). Cas13-Induced Cellular Dormancy Prevents the Rise of CRISPR-Resistant Bacteriophage. In: *Nature* 570.7760, pp. 241–245. DOI: 10.1038/s41586-019-1257-5 (cit. on p. 24).

Michael, C. A. and Labbate, M. (2010). Gene Cassette Transcription in a Large Integron-Associated Array. In: *BMC Genetics* 11.1, p. 82. DOI: 10.1186/1471-2156-11-82 (cit. on p. 125).

Milicevic, O., Repac, J., Bozic, B., Djordjevic, M., and Djordjevic, M. (2019). A Simple Criterion for Inferring CRISPR Array Direction. In: *Frontiers in Microbiology* 10, p. 2054. DOI: 10.3389/fmicb.2019.02054 (cit. on pp. 82–84, 108, 117).

Millman, A. et al. (2022). An Expanded Arsenal of Immune Systems That Protect Bacteria from Phages. In: *Cell Host & Microbe* 30.11, 1556–1569.e5. DOI: 10.1016/j.chom.2022.09.017 (cit. on p. 3).

Mitrofanov, A., Alkhnbashi, O. S., Shmakov, S. A., Makarova, K. S., Koonin, E. V., and Backofen, R. (2021). CRISPRidentify: Identification of CRISPR Arrays Using Machine Learning Approach. In: *Nucleic Acids Research* 49.4, e20–e20. DOI: 10.1093/nar/gkaa1158 (cit. on pp. 16, 83, 84, 93, 96, 101).

Mitrofanov, A., Beisel, C. L., Baumdicker, F., Alkhnbashi, O. S., and Backofen, R. (2026). Comprehensive Analysis of CRISPR Array Repeat Mutations Reveals Subtype-Specific Patterns and Links to Spacer Dynamics. In: *microLife* 7. Ed. by L. Randau, uqaf050. DOI: 10.1093/femsml/uqaf050 (cit. on pp. 14, 35, 84, 98, 113, 122).

Modell, J. W., Jiang, W., and Marraffini, L. A. (2017). CRISPR–Cas Systems Exploit Viral DNA Injection to Establish and Maintain Adaptive Immunity. In: *Nature* 544.7648, pp. 101–104. DOI: 10.1038/nature21719 (cit. on p. 20).

Mohanraju, P., Makarova, K. S., Zetsche, B., Zhang, F., Koonin, E. V., and Van Der Oost, J. (2016). Diverse Evolutionary Roots and Mechanistic Variations of the CRISPR-Cas Systems. In: *Science* 353.6299, aad5147. DOI: 10.1126/science.aad5147 (cit. on p. 13).

Mojica, F. J. M., Díez-Villaseñor, C., García-Martínez, J., and Almendros, C. (2009). Short Motif Sequences Determine the Targets of the Prokaryotic CRISPR Defence System. In: *Microbiology* 155.3, pp. 733–740. DOI: 10.1099/mic.0.023960-0 (cit. on p. 19).

Mojica, F. J. M., Juez, G., and Rodriguez-Valera, F. (1993). Transcription at Different Salinities of *Haloferax Mediterranei* Sequences Adjacent to Partially Modified *Pst* I Sites. In: *Molecular Microbiology* 9.3, pp. 613–621. DOI: 10.1111/j.1365-2958.1993.tb01721.x (cit. on p. 6).

Mojica, F., Ferrer, C., Juez, G., and Rodríguez-Valera, F. (1995). Long Stretches of Short Tandem Repeats Are Present in the Largest Replicons of the *Archaea Haloferax Mediterranei* and *Haloferax Volcanii* and Could Be Involved in Replicon Partitioning. In: *Molecular Microbiology* 17.1, pp. 85–93. DOI: 10.1111/j.1365-2958.1995.mmi_17010085.x (cit. on p. 6).

Mojica, F. J. M. and Díez-Villaseñor, C. (2010). The on–off Switch of CRISPR Immunity against Phages in *Escherichia Coli*. In: *Molecular Microbiology* 77.6, pp. 1341–1345. DOI: 10.1111/j.1365-2958.2010.07326.x (cit. on p. 28).

Mojica, F. J. M., Díez-Villaseñor, C., Soria, E., and Juez, G. (2000). Biological Significance of a Family of Regularly Spaced Repeats in the Genomes of Archaea, Bacteria and Mitochondria. In: *Molecular Microbiology* 36.1, pp. 244–246. DOI: 10.1046/j.1365-2958.2000.01838.x (cit. on pp. 6, 124).

Mojica, F. J. M. and Rodriguez-Valera, F. (2016). The Discovery of CRISPR in Archaea and Bacteria. In: *The FEBS Journal* 283.17, pp. 3162–3169. DOI: 10.1111/febs.13766 (cit. on pp. 6, 7).

Mojica, F. J., Díez-Villaseñor, C., García-Martínez, J., and Soria, E. (2005). Intervening Sequences of Regularly Spaced Prokaryotic Repeats Derive from Foreign Genetic Elements. In: *Journal of Molecular Evolution* 60.2, pp. 174–182. DOI: 10.1007/s00239-004-0046-3 (cit. on pp. 7, 82).

Molina-Sánchez, M. D. et al. (2024). Adaptive Immunity of Type VI CRISPR-Cas Systems Associated with Reverse Transcriptase-Cas1 Fusion Proteins. In: *Nucleic Acids Research* 52.22, pp. 14229–14243. DOI: 10.1093/nar/gkae1154. PMID: 39673266 (cit. on pp. 19, 120).

Morange, M. (2015a). What History Tells Us XXXIX. CRISPR-Cas: From a Prokaryotic Immune System to a Universal Genome Editing Tool. In: *Journal of Biosciences* 40.5, pp. 829–832. DOI: 10.1007/s12038-015-9575-8 (cit. on pp. 6, 7).

– (2015b). What History Tells Us XXXVII. CRISPR-Cas: The Discovery of an Immune System in Prokaryotes. In: *Journal of Biosciences* 40.2, pp. 221–223. DOI: 10.1007/s12038-015-9532-6 (cit. on p. 6).

Morley, D., Broniewski, J. M., Westra, E. R., Buckling, A., and Van Houte, S. (2017). Host Diversity Limits the Evolution of Parasite Local Adaptation. In: *Molecular Ecology* 26.7, pp. 1756–1763. DOI: 10.1111/mec.13917 (cit. on p. 26).

Mosterd, C. and Moineau, S. (2020). Characterization of a Type II-A CRISPR-Cas System in *Streptococcus Mutans*. In: *mSphere* 5.3. Ed. by M. L. Marco, e00235–20. DOI: 10.1128/mSphere.00235-20 (cit. on pp. 21, 33, 64, 97).

Mushegian, A. R. (2020). Are There 1031 Virus Particles on Earth, or More, or Fewer? In: *Journal of Bacteriology* 202.9, e00052–20. DOI: 10.1128/JB.00052-20. PMID: 32071093 (cit. on p. 2).

Muzyukina, P. and Soutourina, O. (2024). CRISPR Genotyping Methods: Tracing the Evolution from Spoligotyping to Machine Learning. In: *Biochimie*. The Rise of RNA: From Fundamental Research to Therapeutic Applications 217, pp. 66–73. DOI: 10.1016/j.biochi.2023.07.017 (cit. on pp. 31, 56, 79).

Nakata, A., Amemura, M., and Makino, K. (1989). Unusual Nucleotide Arrangement with Repeated Sequences in the Escherichia Coli K-12 Chromosome. In: *Journal of Bacteriology* 171.6, pp. 3553–3556. DOI: 10.1128/jb.171.6.3553-3556.1989 (cit. on p. 6).

Nam, K. H. et al. (2012). Cas5d Protein Processes Pre-crRNA and Assembles into a Cascade-like Interference Complex in Subtype I-C/Dvulg CRISPR-Cas System. In: *Structure* 20.9, pp. 1574–1584. DOI: 10.1016/j.str.2012.06.016 (cit. on p. 22).

Nature Biotechnology (2025). Court Reignites CRISPR Patent Dispute. In: *Nature Biotechnology* 43.6, pp. 835–835. DOI: 10.1038/s41587-025-02717-6 (cit. on p. 8).

Ndiaye, M., Bonilla-Rosso, G., Mazel, F., and Engel, P. (2025). Phage Diversity Mirrors Bacterial Strain Diversity in the Honey Bee Gut Microbiota. In: *Nature Communications* 16.1, p. 9738. DOI: 10.1038/s41467-025-64706-2 (cit. on p. 120).

Néron, B., Littner, E., Haudiquet, M., Perrin, A., Cury, J., and Rocha, E. (2022). IntegronFinder 2.0: Identification and Analysis of Integrons across Bacteria, with a Focus on Antibiotic Resistance in

Klebsiella. In: *Microorganisms* 10.4, p. 700. DOI: 10.3390/microorganisms10040700 (cit. on pp. 124–126).

Nethery, M. A. and Barrangou, R. (2019). CRISPR Visualizer: Rapid Identification and Visualization of CRISPR Loci via an Automated High-Throughput Processing Pipeline. In: *RNA Biology* 16.4, pp. 577–584. DOI: 10.1080/15476286.2018.1493332 (cit. on pp. 31, 37).

Nguyen, L.-T., Schmidt, H. A., Von Haeseler, A., and Minh, B. Q. (2015). IQ-TREE: A Fast and Effective Stochastic Algorithm for Estimating Maximum-Likelihood Phylogenies. In: *Molecular Biology and Evolution* 32.1, pp. 268–274. DOI: 10.1093/molbev/msu300 (cit. on pp. 59, 89).

Nicholson, T. J., Jackson, S. A., Croft, B. I., Staals, R. H., Fineran, P. C., and Brown, C. M. (2019). Bioinformatic Evidence of Widespread Priming in Type I and II CRISPR-Cas Systems. In: *RNA Biology* 16.4, pp. 566–576. DOI: 10.1080/15476286.2018.1509662 (cit. on p. 20).

Nickel, L. et al. (2013). Two CRISPR-Cas Systems in Methanosarcina Mazei Strain Gö1 Display Common Processing Features despite Belonging to Different Types I and III. In: *RNA Biology* 10.5, pp. 779–791. DOI: 10.4161/rna.23928 (cit. on p. 19).

Niewoehner, O. et al. (2017). Type III CRISPR–Cas Systems Produce Cyclic Oligoadenylate Second Messengers. In: *Nature* 548.7669, pp. 543–548. DOI: 10.1038/nature23467 (cit. on p. 24).

Nobel Prize Outreach (2026a). The Nobel Prize in Chemistry 2020. NobelPrize.org. URL: https://www.nobelprize.org/prizes/chemistry/2020/summary/ (visited on 01/03/2026) (cit. on pp. 1, 4, 8).

– (2026b). The Nobel Prize in Physiology or Medicine 1978. NobelPrize.org. URL: https://www.nobelprize.org/prizes/medicine/1978/summary/ (visited on 01/03/2026) (cit. on pp. 1, 4).

Nobrega, F. L., Walinga, H., Dutilh, B. E., and Brouns, S. J. J. (2020). Prophages Are Associated with Extensive CRISPR–Cas Auto-Immunity. In: *Nucleic Acids Research* 48.21, pp. 12074–12084. DOI: 10.1093/nar/gkaa1071 (cit. on pp. 27, 28).

Nuñez, J. K., Kranzusch, P. J., Noeske, J., Wright, A. V., Davies, C. W., and Doudna, J. A. (2014). Cas1–Cas2 Complex Formation Mediates Spacer Acquisition during CRISPR–Cas Adaptive Immunity. In: *Nature Structural & Molecular Biology* 21.6, pp. 528–534. DOI: 10.1038/nsmb.2820 (cit. on pp. 7, 18).

Nuñez, J. K., Bai, L., Harrington, L. B., Hinder, T. L., and Doudna, J. A. (2016). CRISPR Immunological Memory Requires a Host Factor for Specificity. In: *Molecular Cell* 62.6, pp. 824–833. DOI: 10.1016/j.molcel.2016.04.027 (cit. on p. 21).

Nuñez, J. K., Lee, A. S. Y., Engelman, A., and Doudna, J. A. (2015). Integrase-Mediated Spacer Acquisition during CRISPR–Cas Adaptive Immunity. In: *Nature* 519.7542, pp. 193–198. DOI: 10.1038/nature14237 (cit. on p. 21).

Nussenzweig, P. M. and Marraffini, L. A. (2020). Molecular Mechanisms of CRISPR-Cas Immunity in Bacteria. In: *Annual Review of Genetics* 54.1, pp. 93–120. DOI: 10.1146/annurev-genet-022120-112523 (cit. on pp. 5, 13, 19–24, 28, 33).

Nussenzweig, P. M., McGinn, J., and Marraffini, L. A. (2019). Cas9 Cleavage of Viral Genomes Primes the Acquisition of New Immunological Memories. In: *Cell Host & Microbe* 26.4, 515–526.e6. DOI: 10.1016/j.chom.2019.09.002 (cit. on p. 20).

O'Sullivan, L., Bolton, D., McAuliffe, O., and Coffey, A. (2019). Bacteriophages in Food Applications: From Foe to Friend. In: *Annual Review of Food Science and Technology* 10.1, pp. 151–172. DOI: 10.1146/annurev-food-032818-121747 (cit. on pp. 9, 119, 120).

Oliveira, P. H., Touchon, M., and Rocha, E. P. (2014). The Interplay of Restriction-Modification Systems with Mobile Genetic Elements and Their Prokaryotic Hosts. In: *Nucleic Acids Research* 42.16, pp. 10618–10631. DOI: 10.1093/nar/gku734 (cit. on p. 4).

Ortega-Sanz, I., Rubio, A., and Pérez-Pulido, A. J. (2025). Bacteriophages in Pseudomonas Aeruginosa Evade the CRISPR-Cas I-F System by Depletion of PAM Sequences. In: *Microbial Genomics* 11.6. DOI: 10.1099/mgen.0.001423 (cit. on p. 26).

Padilha, V. A., Alkhnbashi, O. S., Shah, S. A., de Carvalho, A. C. P. L. F., and Backofen, R. (2020). CRISPRcasIdentifier: Machine Learning for Accurate Identification and Classification of CRISPR-Cas Systems. In: *GigaScience* 9.6, giaa062. DOI: 10.1093/gigascience/giaa062 (cit. on pp. 16, 60, 101).

Paez-Espino, D. et al. (2015). CRISPR Immunity Drives Rapid Phage Genome Evolution in Streptococcus Thermophilus. In: *mBio* 6.2. Ed. by J. K. Jansson, e00262–15. DOI: 10.1128/mBio.00262-15 (cit. on pp. 19, 26, 33).

Palmer, K. L. and Gilmore, M. S. (2010). Multidrug-Resistant Enterococci Lack CRISPR- *Cas*. In: *mBio* 1.4. Ed. by R. Losick, e00227–10. DOI: 10.1128/mBio.00227-10 (cit. on p. 28).

Palmnäs-Bédard, M., de Santa Izabel, A., Dicksved, J., and Landberg, R. (2023). Characterization of the Bacterial Composition of 47 Fermented Foods in Sweden. In: *Foods* 12.20, p. 3827. DOI: 10.3390/foods12203827. PMID: 37893721 (cit. on pp. 9, 119).

Patterson, A. G., Yevstigneyeva, M. S., and Fineran, P. C. (2017). Regulation of CRISPR–Cas Adaptive Immune Systems. In: *Current Opinion in Microbiology* 37, pp. 1–7. DOI: 10.1016/j.mib.2017.02.004 (cit. on p. 33).

Patterson, A. G. et al. (2016). Quorum Sensing Controls Adaptive Immunity through the Regulation of Multiple CRISPR-Cas Systems. In: *Molecular Cell* 64.6, pp. 1102–1108. DOI: 10.1016/j.molcel.2016.11.012 (cit. on p. 33).

Pavlova, Y. S., Paez-Espino, D., Morozov, A. Y., and Belalov, I. S. (2021). Searching for Fat Tails in CRISPR-Cas Systems: Data Analysis and Mathematical Modeling. In: *PLOS Computational Biology* 17.3. Ed. by O. A. Igoshin, e1008841. DOI: 10.1371/journal.pcbi.1008841 (cit. on pp. 51, 76).

Peach, L. J., Zhang, H., Weaver, B. P., and Boedicker, J. Q. (2025). Assessing Spacer Acquisition Rates in E. Coli Type I-E CRISPR Arrays. In: *Frontiers in Microbiology* 15. DOI: 10.3389/fmicb.2024.1498959 (cit. on pp. 33, 111, 117).

Perez-Rodriguez, R. et al. (2011). Envelope Stress Is a Trigger of CRISPR RNA-Mediated DNA Silencing in Escherichia Coli: Envelope Stress Triggers CRISPR Silencing. In: *Molecular Microbiology* 79.3, pp. 584–599. DOI: 10.1111/j.1365-2958.2010.07482.x (cit. on p. 33).

Pertziger, E. et al. (2025). Classification and Estimation of Dietary Live Microorganisms and Fermented Foods Intake in Swiss Adults. In: *The Journal of Nutrition* 155.8, pp. 2717–2728. DOI: 10.1016/j.tjnut.2025.05.042 (cit. on pp. 9, 119).

Pinilla-Redondo, R. et al. (2020). Type IV CRISPR–Cas Systems Are Highly Diverse and Involved in Competition between Plasmids. In: *Nucleic Acids Research* 48.4, pp. 2000–2012. DOI: 10.1093/nar/gkz1197 (cit. on pp. 3, 19, 24, 25, 27, 29, 104, 113, 117).

Pinilla-Redondo, R. et al. (2022). CRISPR-Cas Systems Are Widespread Accessory Elements across Bacterial and Archaeal Plasmids. In: *Nucleic Acids Research* 50.8, pp. 4315–4328. DOI: 10.1093/nar/gkab859 (cit. on pp. 3, 24, 27, 29, 100, 104).

Plagens, A. et al. (2014). In Vitro Assembly and Activity of an Archaeal CRISPR-Cas Type I-A Cascade Interference Complex. In: *Nucleic Acids Research* 42.8, pp. 5125–5138. DOI: 10.1093/nar/gku120 (cit. on p. 22).

Pougach, K. et al. (2010). Transcription, Processing and Function of CRISPR Cassettes in *Escherichia Coli*. In: *Molecular Microbiology* 77.6, pp. 1367–1379. DOI: 10.1111/j.1365-2958.2010.07265.x (cit. on pp. 14, 28, 82).

Pourcel, C., Salvignol, G., and Vergnaud, G. (2005). CRISPR Elements in Yersinia Pestis Acquire New Repeats by Preferential Uptake of Bacteriophage DNA, and Provide Additional Tools for Evolutionary Studies. In: *Microbiology* 151.3, pp. 653–663. DOI: 10.1099/mic.0.27437-0 (cit. on pp. 7, 21, 31, 34, 35).

Pourcel, C. et al. (2019). CRISPRCasdb a Successor of CRISPRdb Containing CRISPR Arrays and Cas Genes from Complete Genome Sequences, and Tools to Download and Query Lists of Repeats and Spacers. In: *Nucleic Acids Research*, gkz915. DOI: 10.1093/nar/gkz915 (cit. on pp. 13, 17, 56, 57, 86, 88).

Prakash, A. and Kumar, M. (2022). Transcriptional Analysis of CRISPR I-B Arrays of Leptospira Interrogans Serovar Lai and Its Processing by Cas6. In: *Frontiers in Microbiology* 13, p. 960559. DOI: 10.3389/fmicb.2022.960559 (cit. on pp. 22, 97).

Pul, Ü., Wurm, R., Arslan, Z., Geißen, R., Hofmann, N., and Wagner, R. (2010). Identification and Characterization of *E. Coli* CRISPR- *Cas* Promoters and Their Silencing by H-NS. In: *Molecular Microbiology* 75.6, pp. 1495–1512. DOI: 10.1111/j.1365-2958.2010.07073.x (cit. on p. 28).

Pupko, T., Pe, I., Shamir, R., and Graur, D. (2000). A Fast Algorithm for Joint Reconstruction of Ancestral Amino Acid Sequences. In: *Molecular Biology and Evolution* 17.6, pp. 890–896. DOI: 10.1093/oxfordjournals.molbev.a026369 (cit. on pp. 41, 43, 62).

Pyenson, N. C. and Marraffini, L. A. (2020). Co-Evolution within Structured Bacterial Communities Results in Multiple Expansion of CRISPR Loci and Enhanced Immunity. In: *eLife* 9, e53078. DOI: 10.7554/eLife.53078 (cit. on p. 26).

Quax, T. E. F. et al. (2013). Massive Activation of Archaeal Defense Genes during Viral Infection. In: *Journal of Virology* 87.15, pp. 8419–8428. DOI: 10.1128/JVI.01020-13 (cit. on p. 33).

Rao, C., Chin, D., and Ensminger, A. W. (2017). Priming in a Permissive Type I-C CRISPR-Cas System Reveals Distinct Dynamics of Spacer Acquisition and Loss. In: *RNA (New York, N.Y.)* 23.10, pp. 1525–1538. DOI: 10.1261/rna.062083.117. PMID: 28724535 (cit. on pp. 20, 34, 35).

Rao, C. et al. (2016). Active and Adaptive *Legionella* CRISPR-Cas Reveals a Recurrent Challenge to the Pathogen. In: *Cellular Microbiology* 18.10, pp. 1319–1338. DOI: 10.1111/cmi.12586 (cit. on pp. 19, 33).

Redding, S. et al. (2015). Surveillance and Processing of Foreign DNA by the Escherichia Coli CRISPR-Cas System. In: *Cell* 163.4, pp. 854–865. DOI: 10.1016/j.cell.2015.10.003 (cit. on pp. 20, 23).

Reyes-Robles, T. et al. (2018). Vibrio Cholerae Outer Membrane Vesicles Inhibit Bacteriophage Infection. In: *Journal of Bacteriology* 200.15. Ed. by V. J. DiRita. DOI: 10.1128/JB.00792-17 (cit. on p. 3).

Richard, E. et al. (2024a). Belt and Braces: Two Escape Ways to Maintain the Cassette Reservoir of Large Chromosomal Integrons. In: *PLOS Genetics* 20.4. Ed. by A. B. Dalia, e1011231. DOI: 10.1371/journal.pgen.1011231 (cit. on p. 125).

Richard, E. et al. (2024b). Cassette Recombination Dynamics within Chromosomal Integrons Are Regulated by Toxin-Antitoxin Systems. In: *Science Advances* 10.2, eadj3498. DOI: 10.1126/sciadv.adj3498 (cit. on p. 125).

Richter, C. et al. (2014). Priming in the Type I-F CRISPR-Cas System Triggers Strand-Independent Spacer Acquisition, Bi-Directionally from the Primed Protospacer. In: *Nucleic Acids Research* 42.13, pp. 8516–8526. DOI: 10.1093/nar/gku527 (cit. on pp. 20, 21).

Richter, H., Zoephel, J., Schermuly, J., Maticzka, D., Backofen, R., and Randau, L. (2012). Characterization of CRISPR RNA Processing in Clostridium Thermocellum and Methanococcus Maripaludis. In: *Nucleic Acids Research* 40.19, pp. 9887–9896. DOI: 10.1093/nar/gks737 (cit. on p. 22).

Romberg, J.-M. (2024). "Visualizing CRISPR Array Evolution". MA thesis. University of Tübingen (cit. on p. 121).

Rosner, F. (2025). "Reconstruction of Evolutionary Events in Bacterial Integrons". MA thesis. University of Tübingen (cit. on p. 126).

Rostøl, J. T. and Marraffini, L. A. (2019). Non-Specific Degradation of Transcripts Promotes Plasmid Clearance during Type III-A CRISPR–Cas Immunity. In: *Nature Microbiology* 4.4, pp. 656–662. DOI: 10.1038/s41564-018-0353-x (cit. on p. 24).

Rousset, F. et al. (2022). Phages and Their Satellites Encode Hotspots of Antiviral Systems. In: *Cell Host & Microbe* 30.5, 740–753.e5. DOI: 10.1016/j.chom.2022.02.018 (cit. on p. 3).

Russel, J., Pinilla-Redondo, R., Mayo-Muñoz, D., Shah, S. A., and Sørensen, S. J. (2020). CRISPRCasTyper: Automated Identification, Annotation, and Classification of CRISPR-Cas Loci. In: *The CRISPR Journal* 3.6, pp. 462–469. DOI: 10.1089/crispr.2020.0059 (cit. on p. 16).

Rutkauskas, M., Sinkunas, T., Songailiene, I., Tikhomirova, M. S., Siksnys, V., and Seidel, R. (2015). Directional R-Loop Formation by the CRISPR-Cas Surveillance Complex Cascade Provides Efficient Off-Target Site Rejection. In: *Cell Reports* 10.9, pp. 1534–1543. DOI: 10.1016/j.celrep.2015.01.067 (cit. on p. 23).

Rykachevsky, A. et al. (2021). SCRAMBLER: A Tool for De Novo CRISPR Array Reconstruction and Its Application for Analysis of the Structure of Prokaryotic Populations. In: *The CRISPR Journal* 4.5, pp. 673–685. DOI: 10.1089/crispr.2021.0012 (cit. on pp. 79, 80).

Sagulenko, P., Puller, V., and Neher, R. A. (2018). TreeTime: Maximum-Likelihood Phylodynamic Analysis. In: *Virus Evolution* 4.1. DOI: 10.1093/ve/vex042 (cit. on p. 62).

Samai, P., Pyenson, N., Jiang, W., Goldberg, G. W., Hatoum-Aslan, A., and Marraffini, L. A. (2015). Co-Transcriptional DNA and RNA Cleavage during Type III CRISPR-Cas Immunity. In: *Cell* 161.5, pp. 1164–1174. DOI: 10.1016/j.cell.2015.04.027 (cit. on pp. 23, 24, 30).

Sampson, T. R., Saroj, S. D., Llewellyn, A. C., Tzeng, Y.-L., and Weiss, D. S. (2013). A CRISPR/Cas System Mediates Bacterial Innate Immune Evasion and Virulence. In: *Nature* 497.7448, pp. 254–257. DOI: 10.1038/nature12048 (cit. on pp. 29, 84).

– (2019). Author Correction: A CRISPR/Cas System Mediates Bacterial Innate Immune Evasion and Virulence. In: *Nature* 570.7760, E30–E31. DOI: 10.1038/s41586-019-1253-9 (cit. on p. 84).

Samson, J. E., Magadán, A. H., Sabri, M., and Moineau, S. (2013). Revenge of the Phages: Defeating Bacterial Defences. In: *Nature Reviews Microbiology* 11.10, pp. 675–687. DOI: 10.1038/nrmicro3096 (cit. on p. 4).

Sapranauskas, R., Gasiunas, G., Fremaux, C., Barrangou, R., Horvath, P., and Siksnys, V. (2011). The Streptococcus Thermophilus CRISPR/Cas System Provides Immunity in Escherichia Coli. In: *Nucleic Acids Research* 39.21, pp. 9275–9282. DOI: 10.1093/nar/gkr606 (cit. on pp. 8, 26).

Sashital, D. G., Wiedenheft, B., and Doudna, J. A. (2012). Mechanism of Foreign DNA Selection in a Bacterial Adaptive Immune System. In: *Molecular Cell* 46.5, pp. 606–615. DOI: 10.1016/j.molcel.2012.03.020 (cit. on p. 23).

Savitskaya, E. et al. (2017). Dynamics of *Escherichia Coli* Type I-E CRISPR Spacers over 42 000 Years. In: *Molecular Ecology* 26.7, pp. 2019–2026. DOI: 10.1111/mec.13961 (cit. on p. 29).

Schmidt, F., Cherepkova, M. Y., and Platt, R. J. (2018). Transcriptional Recording by CRISPR Spacer Acquisition from RNA. In: *Nature* 562.7727, pp. 380–385. DOI: 10.1038/s41586-018-0569-1 (cit. on p. 19).

Schröder, L. D., Mitrofanov, A., Koeksal, R., Uhl, M., and Backofen, R. (2026). Biological Foundation Models Enable CRISPR Array Detection Without Metagenomic Assembly. DOI: 10.64898/2026.02.16.706169. URL: http://biorxiv.org/lookup/doi/10.64898/2026.02.16.706169 (visited on 02/27/2026). Pre-published (cit. on p. 17).

Schwengers, O., Jelonek, L., Dieckmann, M. A., Beyvers, S., Blom, J., and Goesmann, A. (2021). Bakta: Rapid and Standardized Annotation of Bacterial Genomes via Alignment-Free Sequence Identification. In: *Microbial Genomics* 7.11. DOI: 10.1099/mgen.0.000685 (cit. on pp. 59, 89).

Seed, K. D. et al. (2014). Evolutionary Consequences of Intra-Patient Phage Predation on Microbial Populations. In: *eLife* 3, e03497. DOI: 10.7554/eLife.03497 (cit. on pp. 3, 9, 120).

Seed, K. D., Lazinski, D. W., Calderwood, S. B., and Camilli, A. (2013). A Bacteriophage Encodes Its Own CRISPR/Cas Adaptive Response to Evade Host Innate Immunity. In: *Nature* 494.7438, pp. 489–491. DOI: 10.1038/nature11927 (cit. on pp. 27, 29).

Sefcikova, J., Roth, M., Yu, G., and Li, H. (2017). Cas6 Processes Tight and Relaxed Repeat RNA via Multiple Mechanisms: A Hypothesis. In: *BioEssays* 39.6, p. 1700019. DOI: 10.1002/bies.201700019 (cit. on p. 22).

Semenova, E. et al. (2011). Interference by Clustered Regularly Interspaced Short Palindromic Repeat (CRISPR) RNA Is Governed by a Seed Sequence. In: *Proceedings of the National Academy of Sciences* 108.25, pp. 10098–10103. DOI: 10.1073/pnas.1104144108 (cit. on pp. 19, 20, 26).

Semenova, E. et al. (2016). Highly Efficient Primed Spacer Acquisition from Targets Destroyed by the *Escherichia Coli* Type I-E CRISPR-Cas Interfering Complex. In: *Proceedings of the National Academy of Sciences* 113.27, pp. 7626–7631. DOI: 10.1073/pnas.1602639113 (cit. on p. 20).

Shah, S. A., Erdmann, S., Mojica, F. J., and Garrett, R. A. (2013). Protospacer Recognition Motifs: Mixed Identities and Functional Diversity. In: *RNA Biology* 10.5, pp. 891–899. DOI: 10.4161/rna.23764 (cit. on pp. 19, 20, 27).

Shao, Y. and Li, H. (2013). Recognition and Cleavage of a Nonstructured CRISPR RNA by Its Processing Endoribonuclease Cas6. In: *Structure* 21.3, pp. 385–393. DOI: 10.1016/j.str.2013.01.010 (cit. on p. 22).

Shao, Y. et al. (2016). A Non-Stem-Loop CRISPR RNA Is Processed by Dual Binding Cas6. In: *Structure* 24.4, pp. 547–554. DOI: 10.1016/j.str.2016.02.009 (cit. on p. 22).

Shariat, N. and Dudley, E. G. (2014). CRISPRs: Molecular Signatures Used for Pathogen Subtyping. In: *Applied and Environmental Microbiology* 80.2, pp. 430–439. DOI: 10.1128/AEM.02790-13 (cit. on pp. 31, 56).

Al-Shayeb, B. et al. (2020). Clades of Huge Phages from across Earth's Ecosystems. In: *Nature* 578.7795, pp. 425–431. DOI: 10.1038/s41586-020-2007-4 (cit. on pp. 3, 27, 29).

Shiimori, M., Garrett, S. C., Graveley, B. R., and Terns, M. P. (2018). Cas4 Nucleases Define the PAM, Length, and Orientation of DNA Fragments Integrated at CRISPR Loci. In: *Molecular Cell* 70.5, 814–824.e6. DOI: 10.1016/j.molcel.2018.05.002 (cit. on p. 19).

Shivram, H., Cress, B. F., Knott, G. J., and Doudna, J. A. (2021). Controlling and Enhancing CRISPR Systems. In: *Nature Chemical Biology* 17.1, pp. 10–19. DOI: 10.1038/s41589-020-00700-7 (cit. on p. 9).

Shmakov, S. et al. (2015). Discovery and Functional Characterization of Diverse Class 2 CRISPR-Cas Systems. In: *Molecular Cell* 60.3, pp. 385–397. DOI: 10.1016/j.molcel.2015.10.008 (cit. on pp. 15, 22, 24).

Shmakov, S. A., Sitnik, V., Makarova, K. S., Wolf, Y. I., Severinov, K. V., and Koonin, E. V. (2017). The CRISPR Spacer Space Is Dominated by Sequences from Species-Specific Mobilomes. In: *mBio* 8.5, e01397–17. DOI: 10.1128/mBio.01397-17 (cit. on pp. 21, 22, 97, 120).

Silas, S. et al. (2016). Direct CRISPR Spacer Acquisition from RNA by a Natural Reverse Transcriptase–Cas1 Fusion Protein. In: *Science* 351.6276, aad4234. DOI: 10.1126/science.aad4234 (cit. on p. 19).

Silas, S. et al. (2017). Type III CRISPR-Cas Systems Can Provide Redundancy to Counteract Viral Escape from Type I Systems. In: *eLife* 6, e27601. DOI: 10.7554/eLife.27601 (cit. on pp. 19, 20, 24, 29, 30, 115, 117, 123).

Sinkunas, T. et al. (2013). In Vitro Reconstitution of Cascade-mediated CRISPR Immunity in Streptococcus Thermophilus. In: *The EMBO Journal* 32.3, pp. 385–394. DOI: 10.1038/emboj.2012.352 (cit. on p. 23).

Skanata, A. and Kussell, E. (2021). Ecological Memory Preserves Phage Resistance Mechanisms in Bacteria. In: *Nature Communications* 12.1, p. 6817. DOI: 10.1038/s41467-021-26609-w (cit. on pp. 25, 27, 28).

Skennerton, C. T., Imelfort, M., and Tyson, G. W. (2013). Crass: Identification and Reconstruction of CRISPR from Unassembled Metagenomic Data. In: *Nucleic Acids Research* 41.10, e105–e105. DOI: 10.1093/nar/gkt183 (cit. on p. 17).

Smith, L. M. and Fineran, P. C. (2025). Type I CRISPR-Cas Immunity Primes Type III Spacer Acquisition. In: *Cell Host & Microbe* 33.9, 1561–1576.e6. DOI: 10.1016/j.chom.2025.07.021 (cit. on pp. 20, 24, 29, 30, 123).

Somerville, V. et al. (2022a). Extensive Diversity and Rapid Turnover of Phage Defense Repertoires in Cheese-Associated Bacterial Communities. In: *Microbiome* 10.1, p. 137. DOI: 10.1186/s40168-022-01328-6 (cit. on pp. 9, 33, 119).

Somerville, V. et al. (2022b). Functional Strain Redundancy and Persistent Phage Infection in Swiss Hard Cheese Starter Cultures. In: *The ISME Journal* 16.2, pp. 388–399. DOI: 10.1038/s41396-021-01071-0 (cit. on pp. 9, 33, 119).

Soto-Perez, P., Bisanz, J. E., Berry, J. D., Lam, K. N., Bondy-Denomy, J., and Turnbaugh, P. J. (2019). CRISPR-Cas System of a Prevalent Human Gut Bacterium Reveals Hyper-Targeting against Phages in a Human Virome Catalog. In: *Cell Host & Microbe* 26.3, 325–335.e5. DOI: 10.1016/j.chom.2019.08.008 (cit. on pp. 31, 120).

Staals, R. H. J., Jackson, S. A., Biswas, A., Brouns, S. J. J., Brown, C. M., and Fineran, P. C. (2016). Interference-Driven Spacer Acquisition Is Dominant over Naive and Primed Adaptation in a Native CRISPR–Cas System. In: *Nature Communications* 7.1, p. 12853. DOI: 10.1038/ncomms12853 (cit. on pp. 20, 21).

Staals, R. H. et al. (2014). RNA Targeting by the Type III-A CRISPR-Cas Csm Complex of Thermus Thermophilus. In: *Molecular Cell* 56.4, pp. 518–530. DOI: 10.1016/j.molcel.2014.10.005 (cit. on p. 23).

Stella, G. and Marraffini, L. (2024). Type III CRISPR-Cas: Beyond the Cas10 Effector Complex. In: *Trends in Biochemical Sciences* 49.1, pp. 28–37. DOI: 10.1016/j.tibs.2023.10.006 (cit. on p. 24).

Stern, A., Keren, L., Wurtzel, O., Amitai, G., and Sorek, R. (2010). Self-Targeting by CRISPR: Gene Regulation or Autoimmunity? In: *Trends in Genetics* 26.8, pp. 335–340. DOI: 10.1016/j.tig.2010.05.008 (cit. on p. 27).

Sternberg, S. H., LaFrance, B., Kaplan, M., and Doudna, J. A. (2015). Conformational Control of DNA Target Cleavage by CRISPR–Cas9. In: *Nature* 527.7576, pp. 110–113. DOI: 10.1038/nature15544 (cit. on p. 23).

Stokes, H. W. and Hall, R. M. (1989). A Novel Family of Potentially Mobile DNA Elements Encoding Site-specific Gene-integration Functions: Integrons. In: *Molecular Microbiology* 3.12, pp. 1669–1683. DOI: 10.1111/j.1365-2958.1989.tb00153.x (cit. on p. 125).

Stout, E. A., Sanozky-Dawes, R., Goh, Y. J., Crawley, A. B., Klaenhammer, T. R., and Barrangou, R. (2018). Deletion-Based Escape of CRISPR-Cas9 Targeting in Lactobacillus Gasseri. In: *Microbiology* 164.9, pp. 1098–1111. DOI: 10.1099/mic.0.000689 (cit. on pp. 33, 34, 71).

Stringer, A. M., Baniulyte, G., Lasek-Nesselquist, E., Seed, K. D., and Wade, J. T. (2020). Transcription Termination and Antitermination of Bacterial CRISPR Arrays. In: *eLife* 9. Ed. by B. Wiedenheft, G. Storz, and J. Bondy-Denomy, e58182. DOI: 10.7554/eLife.58182 (cit. on p. 98).

Strotskaya, A. et al. (2017). The Action of *Escherichia Coli* CRISPR–Cas System on Lytic Bacteriophages with Different Lifestyles and Development Strategies. In: *Nucleic Acids Research*, gkx042. DOI: 10.1093/nar/gkx042 (cit. on pp. 26, 27).

Sun, C. L., Barrangou, R., Thomas, B. C., Horvath, P., Fremaux, C., and Banfield, J. F. (2013). Phage Mutations in Response to CRISPR Diversification in a Bacterial Population. In: *Environmental Microbiology* 15.2, pp. 463–470. DOI: 10.1111/j.1462-2920.2012.02879.x (cit. on p. 26).

Sun, C. L., Thomas, B. C., Barrangou, R., and Banfield, J. F. (2016). Metagenomic Reconstructions of Bacterial CRISPR Loci Constrain Population Histories. In: *The ISME Journal* 10.4, pp. 858–870. DOI: 10.1038/ismej.2015.162 (cit. on pp. 26, 56, 80, 120).

Sun, L., Wang, J., Yan, F., Wang, G., Li, Y., and Huang, J. (2022). CrisprVi: A Software for Visualizing and Analyzing CRISPR Sequences of Prokaryotes. In: *BMC Bioinformatics* 23.S3, p. 172. DOI: 10.1186/s12859-022-04716-9 (cit. on pp. 31, 37).

Suttle, C. A. (1994). The Significance of Viruses to Mortality in Aquatic Microbial Communities. In: *Microbial Ecology* 28.2, pp. 237–243. DOI: 10.1007/BF00166813 (cit. on p. 3).

Suttle, C. A. (2005). Viruses in the Sea. In: *Nature* 437.7057, pp. 356–361. DOI: 10.1038/nature04160 (cit. on pp. 2, 3).

– (2007). Marine Viruses — Major Players in the Global Ecosystem. In: *Nature Reviews Microbiology* 5.10, pp. 801–812. DOI: 10.1038/nrmicro1750 (cit. on p. 3).

Swarts, D. C. and Jinek, M. (2019). Mechanistic Insights into the Cis- and Trans-Acting DNase Activities of Cas12a. In: *Molecular Cell* 73.3, 589–600.e4. DOI: 10.1016/j.molcel.2018.11.021 (cit. on p. 23).

Swarts, D. C., Mosterd, C., Van Passel, M. W. J., and Brouns, S. J. J. (2012). CRISPR Interference Directs Strand Specific Spacer Acquisition. In: *PLoS ONE* 7.4. Ed. by I. Mokrousov, e35888. DOI: 10.1371/journal.pone.0035888 (cit. on p. 21).

Talibli, F. and Voß, B. (2025). Metagenomic CRISPR Array Analysis Tool: A Novel Graph-Based Approach to Finding CRISPR Arrays in Metagenomic Datasets. In: *microLife* 6, uqaf016. DOI: 10.1093/femsml/uqaf016. PMID: 40801057 (cit. on p. 17).

Tesson, F. et al. (2022). Systematic and Quantitative View of the Antiviral Arsenal of Prokaryotes. In: *Nature Communications* 13.1, p. 2561. DOI: 10.1038/s41467-022-30269-9 (cit. on p. 3).

Toledo-Arana, A. et al. (2009). The Listeria Transcriptional Landscape from Saprophytism to Virulence. In: *Nature* 459.7249, pp. 950–956. DOI: 10.1038/nature08080 (cit. on p. 29).

Tomida, J. et al. (2017). Diversity and Microevolution of CRISPR Loci in Helicobacter Cinaedi. In: *PLOS ONE* 12.10. Ed. by W. R. Hess, e0186241. DOI: 10.1371/journal.pone.0186241 (cit. on pp. 31, 56, 79, 120).

Toms, A. and Barrangou, R. (2017). On the Global CRISPR Array Behavior in Class I Systems. In: *Biology Direct* 12.1, p. 20. DOI: 10.1186/s13062-017-0193-2 (cit. on p. 77).

Touchon, M., Charpentier, S., Clermont, O., Rocha, E. P. C., Denamur, E., and Branger, C. (2011). CRISPR Distribution within the Escherichia Coli Species Is Not Suggestive of Immunity-Associated Diversifying Selection. In: *Journal of Bacteriology* 193.10, pp. 2460–2467. DOI: 10.1128/JB.01307-10 (cit. on p. 28).

Touchon, M. and Rocha, E. P. C. (2010). The Small, Slow and Specialized CRISPR and Anti-CRISPR of Escherichia and Salmonella. In: *PLoS ONE* 5.6. Ed. by L. Randau, e11126. DOI: 10.1371/journal.pone.0011126 (cit. on pp. 28, 104, 117, 123).

Trasanidou, D., Gerós, A. S., Mohanraju, P., Nieuwenweg, A. C., Nobrega, F. L., and Staals, R. H. J. (2019). Keeping CRISPR in Check: Diverse Mechanisms of Phage-Encoded Anti- CRISPRs. In: *FEMS Microbiology Letters* 366.9, fnz098. DOI: 10.1093/femsle/fnz098 (cit. on p. 26).

Treangen, T. J., Abraham, A.-L., Touchon, M., and Rocha, E. P. (2009). Genesis, Effects and Fates of Repeats in Prokaryotic Genomes. In: *FEMS Microbiology Reviews* 33.3, pp. 539–571. DOI: 10.1111/j.1574-6976.2009.00169.x (cit. on pp. 34, 35, 61).

Vale, P. F., Lafforgue, G., Gatchitch, F., Gardan, R., Moineau, S., and Gandon, S. (2015). Costs of CRISPR-Cas-Mediated Resistance in *Streptococcus Thermophilus*. In: *Proceedings of the Royal Society B: Biological Sciences* 282.1812, p. 20151270. DOI: 10.1098/rspb.2015.1270 (cit. on pp. 28, 34).

Van Houte, S., Buckling, A., and Westra, E. R. (2016a). Evolutionary Ecology of Prokaryotic Immune Mechanisms. In: *Microbiology and Molecular Biology Reviews* 80.3, pp. 745–763. DOI: 10.1128/MMBR.00011-16 (cit. on pp. 3, 4, 13).

Van Houte, S. et al. (2016b). The Diversity-Generating Benefits of a Prokaryotic Adaptive Immune System. In: *Nature* 532.7599, pp. 385–388. DOI: 10.1038/nature17436 (cit. on p. 25).

Van Riet, J. et al. (2022). CRISPRs in the Human Genome Are Differentially Expressed between Malignant and Normal Adjacent to Tumor Tissue. In: *Communications Biology* 5.1, p. 338. DOI: 10.1038/s42003-022-03249-4 (cit. on p. 124).

Vercoe, R. B. et al. (2013). Cytotoxic Chromosomal Targeting by CRISPR/Cas Systems Can Reshape Bacterial Genomes and Expel or Remodel Pathogenicity Islands. In: *PLoS Genetics* 9.4. Ed. by D. Hughes, e1003454. DOI: 10.1371/journal.pgen.1003454 (cit. on pp. 27, 29).

Vialetto, E., Yu, Y., Collins, S. P., Wandera, K. G., Barquist, L., and Beisel, C. L. (2022). A Target Expression Threshold Dictates Invader Defense and Prevents Autoimmunity by CRISPR-Cas13. In: *Cell Host & Microbe* 30.8, 1151–1162.e6. DOI: 10.1016/j.chom.2022.05.013 (cit. on pp. 24, 28–30, 115, 117, 123).

Villiger, L., Joung, J., Koblan, L., Weissman, J., Abudayyeh, O. O., and Gootenberg, J. S. (2024). CRISPR Technologies for Genome, Epigenome and Transcriptome Editing. In: *Nature Reviews Molecular Cell Biology* 25.6, pp. 464–487. DOI: 10.1038/s41580-023-00697-6 (cit. on p. 9).

Vink, J. N. A., Baijens, J. H. L., and Brouns, S. J. J. (2021). PAM-Repeat Associations and Spacer Selection Preferences in Single and Co-Occurring CRISPR-Cas Systems. In: *Genome Biology* 22.1, p. 281. DOI: 10.1186/s13059-021-02495-9 (cit. on pp. 19, 20, 23, 26, 29, 31, 82–84, 88, 93, 96, 98, 100, 104, 113, 117, 120, 123).

Viswanathan, P., Murphy, K., Julien, B., Garza, A. G., and Kroos, L. (2007). Regulation of *Dev* , an Operon That Includes Genes Essential for *Myxococcus Xanthus* Development and CRISPR-Associated Genes and Repeats. In: *Journal of Bacteriology* 189.10, pp. 3738–3750. DOI: 10.1128/JB.00187-07 (cit. on p. 29).

Vit, C. et al. (2021). Cassette Recruitment in the Chromosomal Integron of *Vibrio Cholerae*. In: *Nucleic Acids Research* 49.10, pp. 5654–5670. DOI: 10.1093/nar/gkab412 (cit. on pp. 124, 125).

Vlot, M. et al. (2018). Bacteriophage DNA Glucosylation Impairs Target DNA Binding by Type I and II but Not by Type V CRISPR–Cas Effector Complexes. In: *Nucleic Acids Research* 46.2, pp. 873–885. DOI: 10.1093/nar/gkx1264 (cit. on p. 27).

Wang, J. et al. (2015). Structural and Mechanistic Basis of PAM-Dependent Spacer Acquisition in CRISPR-Cas Systems. In: *Cell* 163.4, pp. 840–853. DOI: 10.1016/j.cell.2015.10.008 (cit. on p. 19).

Wang, J. Y. and Doudna, J. A. (2023). CRISPR Technology: A Decade of Genome Editing Is Only the Beginning. In: *Science* 379.6629, eadd8643. DOI: 10.1126/science.add8643 (cit. on pp. 8, 9, 13, 82, 119).

Wang, J. Y., Pausch, P., and Doudna, J. A. (2022). Structural Biology of CRISPR–Cas Immunity and Genome Editing Enzymes. In: *Nature Reviews Microbiology* 20.11, pp. 641–656. DOI: 10.1038/s41579-022-00739-4 (cit. on pp. 9, 13).

Watson, B. N. J. et al. (2019). Different Genetic and Morphological Outcomes for Phages Targeted by Single or Multiple CRISPR-Cas Spacers. In: *Philosophical Transactions of the Royal Society B: Biological Sciences* 374.1772, p. 20180090. DOI: 10.1098/rstb.2018.0090 (cit. on p. 26).

Watson, B. N. J., Staals, R. H. J., and Fineran, P. C. (2018). CRISPR-Cas-Mediated Phage Resistance Enhances Horizontal Gene Transfer by Transduction. In: *mBio* 9.1. Ed. by J. Bondy-Denomy and M. S. Gilmore, e02406–17. DOI: 10.1128/mBio.02406-17 (cit. on p. 28).

Wei, Y., Chesne, M. T., Terns, R. M., and Terns, M. P. (2015a). Sequences Spanning the Leader-Repeat Junction Mediate CRISPR Adaptation to Phage in *Streptococcus Thermophilus*. In: *Nucleic Acids Research* 43.3, pp. 1749–1758. DOI: 10.1093/nar/gku1407 (cit. on pp. 14, 21, 36).

Wei, Y., Terns, R. M., and Terns, M. P. (2015b). Cas9 Function and Host Genome Sampling in Type II-A CRISPR–Cas Adaptation. In: *Genes & Development* 29.4, pp. 356–361. DOI: 10.1101/gad.257550.114 (cit. on p. 19).

Weinberger, A. D., Wolf, Y. I., Lobkovsky, A. E., Gilmore, M. S., and Koonin, E. V. (2012a). Viral Diversity Threshold for Adaptive Immunity in Prokaryotes. In: *mBio* 3.6. Ed. by S. J. Giovannoni, e00456–12. DOI: 10.1128/mBio.00456-12 (cit. on pp. 13, 25–27, 31).

Weinberger, A. D. et al. (2012b). Persisting Viral Sequences Shape Microbial CRISPR-Based Immunity. In: *PLoS Computational Biology* 8.4. Ed. by C. Von Mering, e1002475. DOI: 10.1371/journal.pcbi.1002475 (cit. on pp. 25–27, 31, 33).

Weissman, J. L. et al. (2018a). Immune Loss as a Driver of Coexistence during Host-Phage Coevolution. In: *The ISME Journal* 12.2, pp. 585–597. DOI: 10.1038/ismej.2017.194 (cit. on p. 27).

Weissman, Jl., Fagan, W. F., and Johnson, P. L. (2018b). Selective Maintenance of Multiple CRISPR Arrays Across Prokaryotes. In: *The CRISPR Journal* 1.6, pp. 405–413. DOI: 10.1089/crispr.2018.0034 (cit. on pp. 14, 21, 27, 28, 33, 34, 77, 79, 100, 104, 105, 107, 116, 117).

Westra, E. R., Buckling, A., and Fineran, P. C. (2014). CRISPR–Cas Systems: Beyond Adaptive Immunity. In: *Nature Reviews Microbiology* 12.5, pp. 317–326. DOI: 10.1038/nrmicro3241 (cit. on p. 28).

Westra, E. R., Dowling, A. J., Broniewski, J. M., and van Houte, S. (2016). Evolution and Ecology of CRISPR. In: *Annual Review of Ecology, Evolution, and Systematics* 47.1, pp. 307–331. DOI: 10.1146/annurev-ecolsys-121415-032428 (cit. on pp. 3, 13, 56).

Westra, E. R. et al. (2010). H-NS-mediated Repression of CRISPR-based Immunity in *Escherichia Coli* K12 Can Be Relieved by the Transcription Activator LeuO. In: *Molecular Microbiology* 77.6, pp. 1380–1393. DOI: 10.1111/j.1365-2958.2010.07315.x (cit. on p. 28).

Westra, E. R. et al. (2015). Parasite Exposure Drives Selective Evolution of Constitutive versus Inducible Defense. In: *Current Biology* 25.8, pp. 1043–1049. DOI: 10.1016/j.cub.2015.01.065 (cit. on p. 26).

Whitman, W. B., Coleman, D. C., and Wiebe, W. J. (1998). Prokaryotes: The Unseen Majority. In: *Proceedings of the National Academy of Sciences* 95.12, pp. 6578–6583. DOI: 10.1073/pnas.95.12.6578 (cit. on p. 2).

Wiedenheft, B. et al. (2011). RNA-Guided Complex from a Bacterial Immune System Enhances Target Recognition through Seed Sequence Interactions. In: *Proceedings of the National Academy of Sciences* 108.25, pp. 10092–10097. DOI: 10.1073/pnas.1102716108 (cit. on p. 20).

Wimmer, F. and Beisel, C. L. (2020). CRISPR-Cas Systems and the Paradox of Self-Targeting Spacers. In: *Frontiers in Microbiology* 10, p. 3078. DOI: 10.3389/fmicb.2019.03078 (cit. on p. 27).

Wright, A. V. and Doudna, J. A. (2016). Protecting Genome Integrity during CRISPR Immune Adaptation. In: *Nature Structural & Molecular Biology* 23.10, pp. 876–883. DOI: 10.1038/nsmb.3289 (cit. on pp. 14, 19, 21).

Wright, A. V., Liu, J.-J., Knott, G. J., Doxzen, K. W., Nogales, E., and Doudna, J. A. (2017). Structures of the CRISPR Genome Integration Complex. In: *Science* 357.6356, pp. 1113–1118. DOI: 10.1126/science.aao0679 (cit. on pp. 18, 21).

Wu, W. Y. et al. (2022). Adaptation by Type V-A and V-B CRISPR-Cas Systems Demonstrates Conserved Protospacer Selection Mechanisms Between Diverse CRISPR-Cas Types. In: *The CRISPR journal* 5.4, pp. 536–547. DOI: 10.1089/crispr.2021.0150. PMID: 35833800 (cit. on p. 19).

Wu, Y. et al. (2024). Bacterial Defense Systems Exhibit Synergistic Anti-Phage Activity. In: *Cell Host & Microbe* 32.4, 557–572.e6. DOI: 10.1016/j.chom.2024.01.015 (cit. on p. 103).

Xiao, Y., Ng, S., Nam, K. H., and Ke, A. (2017a). How Type II CRISPR–Cas Establish Immunity through Cas1–Cas2-Mediated Spacer Integration. In: *Nature* 550.7674, pp. 137–141. DOI: 10.1038/nature24020 (cit. on pp. 14, 18, 21).

Xiao, Y. et al. (2017b). Structure Basis for Directional R-Loop Formation and Substrate Handover Mechanisms in Type I CRISPR-Cas System. In: *Cell* 170.1, 48–60.e11. DOI: 10.1016/j.cell.2017.06.012 (cit. on p. 23).

Yang, J. et al. (2024). Structural Basis for the Activity of the Type VII CRISPR–Cas System. In: *Nature* 633.8029, pp. 465–472. DOI: 10.1038/s41586-024-07815-0 (cit. on p. 25).

Yoganand, K., Sivathanu, R., Nimkar, S., and Anand, B. (2017). Asymmetric Positioning of Cas1–2 Complex and Integration Host Factor Induced DNA Bending Guide the Unidirectional Homing of Protospacer in CRISPR-Cas Type I-E System. In: *Nucleic Acids Research* 45.1, pp. 367–381. DOI: 10.1093/nar/gkw1151 (cit. on p. 21).

Yosef, I., Goren, M. G., and Qimron, U. (2012). Proteins and DNA Elements Essential for the CRISPR Adaptation Process in Escherichia Coli. In: *Nucleic Acids Research* 40.12, pp. 5569–5576. DOI: 10.1093/nar/gks216 (cit. on pp. 7, 14, 18, 19, 21, 34, 36, 82).

Young, J. C. et al. (2012). Phage-Induced Expression of CRISPR-Associated Proteins Is Revealed by Shotgun Proteomics in Streptococcus Thermophilus. In: *PLoS ONE* 7.5. Ed. by H. Bruggemann, e38077. DOI: 10.1371/journal.pone.0038077 (cit. on p. 33).

Zaayman, M. and Wheatley, R. M. (2022). Fitness Costs of CRISPR-Cas Systems in Bacteria. In: *Microbiology* 168.7. DOI: 10.1099/mic.0.001209 (cit. on p. 27).

Zegans, M. E., Wagner, J. C., Cady, K. C., Murphy, D. M., Hammond, J. H., and O'Toole, G. A. (2009). Interaction between Bacteriophage DMS3 and Host CRISPR Region Inhibits Group Behaviors of *Pseudomonas Aeruginosa*. In: *Journal of Bacteriology* 191.1, pp. 210–219. DOI: 10.1128/JB.00797-08 (cit. on p. 29).

Zetsche, B. et al. (2015). Cpf1 Is a Single RNA-Guided Endonuclease of a Class 2 CRISPR-Cas System. In: *Cell* 163.3, pp. 759–771. DOI: 10.1016/j.cell.2015.09.038 (cit. on p. 23).

Zhang, A.-N., Gaston, J. M., Cárdenas, P., Zhao, S., Gu, X., and Alm, E. J. (2025). CRISPR-Cas Spacer Acquisition Is a Rare Event in Human Gut Microbiome. In: *Cell Genomics* 5.1, p. 100725. DOI: 10.1016/j.xgen.2024.100725 (cit. on pp. 33, 120).

Zhang, Q. and Ye, Y. (2017). Not All Predicted CRISPR–Cas Systems Are Equal: Isolated Cas Genes and Classes of CRISPR like Elements. In: *BMC Bioinformatics* 18.1, p. 92. DOI: 10.1186/s12859-017-1512-4 (cit. on pp. 17, 111, 117).

Zhang, X., Garrett, S., Graveley, B. R., and Terns, M. P. (2022). Unique Properties of Spacer Acquisition by the Type III-A CRISPR-Cas System. In: *Nucleic Acids Research* 50.3, pp. 1562–1582. DOI: 10.1093/nar/gkab1193 (cit. on pp. 19, 98).

Zhang, Y. et al. (2013). Processing-Independent CRISPR RNAs Limit Natural Transformation in Neisseria Meningitidis. In: *Molecular Cell* 50.4, pp. 488–503. DOI: 10.1016/j.molcel.2013.05.001 (cit. on pp. 22, 33, 84, 98, 113).