Characteristics of Pyrosequencing Data – Analysis, Methods, and Tools

Susanne Mignon Balzer

PhD thesis

Department of Informatics
University of Bergen
2013

Preface

Everything will be okay in the end. If it's not okay, it's not the end.

John Lennon

When I told my colleagues at Vodafone - where I was working back in 2008 - that I was moving to Norway, they were not surprised. I guess I had a reputation of being restless and always on the hunt for adventures. When I then told them that I was starting a PhD on salmon lice (which was the plan back then, and the project that funded my research), they simply wouldn't believe me. For them, it sounded like a silly thing to do. For me, it sounded like an enormous challenge.

Well, I like challenges. But most of all, I really wanted to move to Norway. Enjoy life in nature. Do watersports and wintersports to my heart's content. Meet new people, learn a new language. Discover new places and have new experiences.

But first, I had to familiarize myself with salmon lice and a lot of background knowledge in bioinformatics. My last biology lesson was in 1996! At the beginning, it was very tough. At the end, it was still tough.

And I procrastinated. A lot. Here are the top 3 things I procrastinated with:

3. Did a 3D jigsaw puzzle of Mont Saint-Michel

2. Dug out my water color paint set from elementary school

1. Scanned hundreds of my dad's photographic slides from the 1960s

I did not only procrastinate though, I also learned a lot about life. Here are the top 3 things I learned during my PhD:

3. Writing smart thoughts and ideas into a file is helpful, but not if you name it "Simsalabim.doc" and forget about it for the next two years

2. The present perfect of "jeg smelter helt" (I am totally melting) is not "jeg smalt helt" (I totally exploded)

1. You should remove stickers from new trousers unless you want someone to find a sticker on your butt saying "QC #1 PASS"

Almost five years after the decision to start a PhD, I am still glad to have started this adventure, of which only one part is coming to an end.

Bergen, 2013,

Susanne Mignon Balzer

Acknowledgements

First of all, I wish to thank the Institute of Marine Research, the University of Bergen, and the Research Council of Norway for giving me the opportunity to pursue a PhD. I especially appreciate the chance to travel and present at international conferences in Barcelona, Gothenburg, Vienna, and Ghent.

Special thanks go to Ketil for being a patient and encouraging advisor. Thank you for picking me up from the airport when I moved to Norway, thank you for apples and oranges, and thank you for believing in me when I had pretty much given up. To Inge, for close support throughout my whole PhD and for record-breaking e-mail response times. Thank you for trusting in me when I wanted to organize the BREW workshop and for taking the time to choose pasta dishes for the participants. It has always impressed me how much you care about the people around you.

To Frank, for accepting my decision to join Inge's group. To Markus, for numerous skype sessions on the JATAC paper, and for your extreme patience in explaining biology to me. It still sounds like Chinese, but I picked up a few words during the last months. To Christopher and Lex, for fruitful discussions on 454 sequencing data. To Anders, for Java code and snowboarding in Voss. To Animesh, for coauthoring the Flowsim paper.

To everyone at CBU for "adopting" a lonely soul without university affiliation, for lunch breaks and extremely successful gingerbread house competitions. To my boss and my former colleagues at the Norwegian Marine Data Center, for countless cups of coffee.

To Matt, Laura, Regan, and Tara, for both exhaustive and last-minute proofreading. I hope I will be able to return the favor one day!

To everyone at the PhD forum for hugs, kicks in the ass and chats. I would have never ever finished my thesis without you.

To all my friends, both in Germany and Norway, for good times, deep conversations, and shoulders to cry on. Without you I would not be who I am. Thank you for travel company, afternoon sailing on the city fjord, and numerous kiteboarding weekends in Finse. Thank you for a legendary 30th birthday party (until 10.30 am!). Thank you to Dag and Ann-Helen, for renting out this lovely place to me. Could you imagine any better place to write a PhD thesis, overlooking the fjord from the terrace?

To Elin, for board game and tapas evenings, and for inspiring me with maps. It was through you that I realized what I really want to work with, and I am having a lot of fun in my new job!

To Tom, for countless text messages, exhilarating breakfasts with your charming little girls, and a deep belief in happy endings.

My warmest thanks go to René, the biggest optimist in my life. Thank you for supporting me in everything I do and want to do. You are one of the best things that has ever happened to me.

Motivation and Aims of Thesis

DNA sequencing methods are used to determine the order of nucleotides in a molecule or set of molecules (e.g. in a genome), and they are crucial for the study of biological systems. For almost thirty years, Sanger sequencing was the primary DNA sequencing technology. With the release of the pyrosequencing platform in 2005, 454 Life Sciences provided researchers with a new, powerful technology for large-scale DNA sequencing. 454 sequencing is currently the only sequencing technology that yields reads with lengths comparable to traditional Sanger sequencing at low error rates, producing reads with a mode length of 700 base pairs (bp) as opposed to approximately 800 bp from Sanger sequencers. This makes 454 sequencing particularly well suited for *de novo* whole genome assembly and metagenomics as well as for a number of other biological fields and applications.

At the time of its release, the 454 platform enabled the production of unprecedented amounts of sequencing data in a highly automated, straightforward fashion. This introduces the risk that the technology is seen as a black box by many biologists and bioinformaticians, mostly because manual inspection of the sequences has become infeasible. The detrimental effect of errors and artifacts on data quality is often neglected or underestimated. In general, researchers rarely have the time and resources to judge the extent to which low quality data is harmful to downstream data analysis, or even to perform sensitivity analyses prior to their actual project study.

There are few papers that deal with 454 data quality directly. Although there have been attempts to reveal the most common and most intrinsic errors in pyrosequencing, the sheer number of papers using 454 sequencing for different purposes makes it impossible to tackle all potential problems. Important details about how researchers deal with inaccuracies in 454 data are often well hidden in the methods section of the numerous application papers, revealing a vast collection of application-specific approaches to data cleaning. As a consequence, there are hundreds of tools and pipelines that are – at least in theory – targeted to 454 data cleaning, some of them originally developed for Sanger sequencing or other platforms and less suited for 454 data. One has to keep in mind that each sequencing platform represents a complex interplay of enzymology, chemistry and software engineering and therefore has its own intrinsic error patterns and sequence characteristics, which highly influence how the reads should be processed and utilized for data analysis. Sequencing statistics such as per-base quality scores are often not comparable across platforms and do not sufficiently represent the true variability of uncertainty.

Another caveat with many of the existing tools is the large number of parameters that can be tuned and options that can be specified. The performance of the tools depends on these settings and thus on the skills of the user. One and the same tool or technology may perform well when operated by experts, while published results and data accuracy cannot be reproduced by less experienced users.

Additionally, it is ultimately left to the researcher to judge the extent to which a project requires data cleaning. This involves not only determining in which order and with which strictness the cleaning steps are performed but also evaluating the tradeoff between quality filtering on the one hand and the retained amount of usable data on the other hand. This task cannot be performed by either biologists or computer scientists alone but requires collaboration between the two groups due to the analytical difficulties raised by the massive amounts of data generated by contemporary sequencers.

In the context of the issues mentioned above, this PhD project aims to enable a comprehensive understanding of error patterns and sequencing artifacts in 454 data. Analyzing and quantifying the impact of errors and artifacts in the context of a variety of applications provides approaches that enable one to gain more information from data, allowing researchers to make use of the findings for developing new data cleaning pipelines.

Summary

The introduction of this thesis provides background knowledge on the 454 sequencing technology and a detailed review of the most relevant sequencing artifacts. Chapter 1 puts the 454 sequencing technology into a historical context. Chapter 2 gives an overview of where 454 sequencing is applied, focusing on the most common application areas. Chapter 3 provides a detailed description of how 454 sequencing works, from library preparation to sequencing, imaging and data output. Here, the distinction between the different detail levels of sequencing information is crucial since data aggregation involves information loss. Chapter 4 describes where errors and artifacts can arise, how they are manifested in the sequencing data, and what impact they can have on downstream analyses. Finally, Chapter 5 puts the contributions into their respective analytical contexts and discusses their relevance for the research community.

The first paper, published in *Bioinformatics* in September 2010 and presented at the European Conference on Computational Biology (ECCB) in Belgium the same year, comprises of the exploration, modeling and simulation of 454 data. Under the title "Characteristics of 454 pyrosequencing data – enabling realistic simulation with Flowsim", we present a detailed analysis of sequencing data and a simulation tool that facilitates the design of sequencing projects. The tool can be used to examine and quantify the impact of read length, coverage, sequencing errors and signal degradation on genome assembly. Furthermore, it enables the testing and benchmarking of known and novel algorithms, methods and tools in a number of application areas such as whole genome assembly, read alignment, read correction, single-nucleotide polymorphism (SNP) identification and metagenomics.

The second paper, "Systematic exploration of error sources in pyrosequencing flowgram data", was published in *Bioinformatics* in July 2011 and presented at the Intelligent Systems for Molecular Biology (ISMB)/ECCB conference in Austria the same year. We added several features and modules to the existing simulation pipeline. Those were based on the observation of several error sources such as copy-

ing errors introduced through polymerase chain reaction (PCR), a method used in 454 sequencing for amplification of the templates. These errors appear as mutations and are virtually impossible to distinguish from true sequence variants.

Similar to the second paper, the third paper, “Filtering duplicate reads from 454 pyrosequencing data”, focuses on a single error type, namely artificially duplicated reads. Our JATAC tool enables removal of this artifact on the most detailed sequencing data level, outperforming existing tools. The paper was published in *Bioinformatics* in April 2013.

List of Publications

Modeling and Simulation

“Characteristics of 454 pyrosequencing data – enabling realistic simulation with flowsim”
Balzer S, Malde K, Lanzén A, Sharma A and Jonassen I
Published in *Bioinformatics* in 2010.
Erratum published in *Bioinformatics* in 2011.

Error Sources

“Systematic exploration of error sources in pyrosequencing flowgram data”
Balzer S, Malde K and Jonassen I
Published in *Bioinformatics* in 2011.

Duplicate Read Removal

“Filtering duplicate reads from 454 pyrosequencing data”
Balzer S, Malde K, Grohme MA and Jonassen I
Published in *Bioinformatics* in 2013.

Contents

I Introduction 1

1 DNA Sequencing in the Post-Sanger Era 3
1.1 Next-Generation Sequencing 4
1.2 Spoilt for Choice - Which Platform to Use? 8

2 454 Sequencing - Milestones and Applications 13
2.1 An Overview of Applications and Fields 13
2.2 Shotgun vs. Paired End Reads 16
2.3 Sequencing Coverage 18
2.4 Genome Assembly 20
2.4.1 Approaches and Issues 22
2.4.2 Assembly Tools for 454 Reads 23
2.4.3 Assembly Quality 25
2.4.4 First Genome Assemblies with 454 Sequencing 27
2.5 Metagenomics 28

3 454 Sequencing – The Basics 33
3.1 What is Pyrosequencing? 33
3.2 The 454 Sequencing Process 36
3.2.1 Library Preparation 36
3.2.2 Emulsion PCR Amplification 38
3.2.3 Sequencing 39
3.2.4 Image and Signal Processing 41
3.2.5 Whole-Read Filtering and Quality-Trimming 43
3.2.6 Base-Calling 46
3.2.7 Quality Score Calculation 47
3.3 Information Extraction Tools 50
3.4 Read Lengths and Throughput 51

4 454 Sequencing – Characteristics and Artifacts 55
4.1 Accuracy . . . 56
4.1.1 Per-Base Error Rates . . . 57
4.1.2 Consensus Accuracy . . . 58
4.2 Known Error Patterns . . . 59
4.2.1 Contamination . . . 61
4.2.2 PCR Bias . . . 61
4.2.3 Chimeras . . . 62
4.2.4 PCR and emPCR Errors . . . 63
4.2.5 Artificially Duplicated Reads . . . 64
4.2.6 Well Cross-Talk Type I and II . . . 67
4.2.7 Homopolymer Errors and Loss of Synchronism . . . 67
4.3 Data Cleaning – Tools and Strategies . . . 69
4.3.1 Additional Filtering and Trimming . . . 70
4.3.2 Adapter, Tag and Contaminant Removal . . . 71
4.3.3 Chimera Removal . . . 71
4.3.4 Duplicate Removal . . . 72
4.3.5 Base-Calling and Quality Score Calculation . . . 73
4.3.6 Multi-Purpose Tools and Pipelines . . . 76
4.3.7 Approaches in Flow Space . . . 76
4.3.8 Data Cleaning in Microbial Diversity Studies . . . 77

5 Contributions and Discussion 85
5.1 Modeling and Simulation . . . 86
5.2 Error Sources . . . 93
5.3 Duplicate Read Removal . . . 94
5.4 Closing Remarks . . . 97

Bibliography 101

II Contributions 123

Part I

Introduction

1

DNA Sequencing in the Post-Sanger Era

"I think there is a world market for maybe five computers."

Thomas Watson, president of IBM, 1943

The year 1977 marked the beginning of modern DNA sequencing. Frederick Sanger published his gel-based enzymatic chain termination method [1] and, three years later, received the Nobel Prize in Chemistry together with Paul Berg and Walter Gilbert. The same year, Gilbert had published an alternative sequencing method, Maxam-Gilbert sequencing [2]. While Maxam-Gilbert sequencing never achieved wide adoption, Sanger sequencing was further developed by a number of researchers, and eventually automated for higher throughput on capillaries which made the gels dispensable [3–6]. Today, Sanger sequencing is often referred to as first-generation sequencing and builds on capillary sequencing. It is commercialized by Applied Biosystems[1].

Sanger's method enabled a number of breakthroughs in the understanding of biological processes. One of the most important achievements was made in 2001 when two competing projects reported a draft sequence of large parts of the human genome [7, 8]. Sequencing of the initial draft cost around $300 million – it became

[1] with the currently distributed sequencing platform 3730XL

clear that there was a great demand for a considerably faster, cheaper and more robust sequencing method.

In 2003, the J. Craig Venter Science Foundation promised an award of \$500,000 to the first group that would present a technology capable of sequencing a human genome for \$1,000 [9]. This incentive and the funding for a series of projects through the US National Human Genome Research Institute (NHGRI) encouraged researchers to come up with new approaches for high-throughput sequencing technologies. The term throughput generally refers to the number of base pairs sequenced in a single run and is influenced by the number of templates sequenced in parallel and their read length.[2]

However, most of the newly developed technologies struggled with short read lengths and did not represent a serious alternative to Sanger sequencing.

1.1 Next-Generation Sequencing

In 1997, the company Pyrosequencing AB was founded in Uppsala, Sweden. It was already as early as 1999 that the first pyrosequencing platform became commercially available, but it only allowed for sequencing short stretches of DNA. In 2003, Pyrosequencing AB was renamed to Biotage and further licensed its pyrosequencing technology to 454 Life Sciences, a company founded in 2000. Eventually, in 2005, a promising new platform was presented. With the Genome Sequencer 20 instrument (hereafter "GS 20"), 454 Life Sciences (purchased by Roche Diagnostics in 2007) introduced a highly parallel, array-based pyrosequencing technology that produces massive amounts of data [10, 11]. The main achievement was an approximately 100-fold increase in throughput over Sanger sequencing at a cost-

[2] Reads are the main output from a sequencing instrument, composed of a sequence of the nucleotide bases adenine (A), cytosine (C), guanine (G) and thymine (T) and determined from segments of sample input. Apart from the four nucleotides, reads can also include undetermined bases (N's).

reduction of up to 25% [12]. Soon, the technology became popular under the name 454 sequencing, and a second platform, GS FLX, was unveiled in 2008. GS FLX Titanium followed in 2009, GS FLX+ in 2011. These platforms are referred to as GS 20, FLX, Titanium and FLX+ in the rest of the thesis.

Other platforms such as the Illumina (formerly Solexa) Genome Analyzer and the Applied Biosystems/SOLiD System (hereafter "Illumina" and "SOLiD") followed. 454 pyrosequencing, Illumina and SOLiD are often referred to as next-generation sequencing (NGS), second-generation sequencing, high-throughput next-generation sequencing (HT-NGS) or ultra-high-throughput sequencing (UHTS) platforms.

454 pyrophosphate-based sequencing (thus the name pyrosequencing) builds on a sequencing-by-synthesis approach. The latter involves determining the sequence of a DNA template by synthesizing the complementary DNA. A single-stranded DNA fragment is made double-stranded by the use of an enzyme (polymerase) that works its way along the fragment, starting at one end. This results in the release of inorganic pyrophosphate which – through a series of enzymatic reactions – produces visible light signals. The amount of light is recorded by a camera, and it is proportional to the number of nucleotides incorporated [10, 13–15]. Consecutive runs of the same nucleotide are referred to as homopolymer runs.

Similarly to 454, also Illumina uses sequencing-by-synthesis, and a camera captures the fluorescently labeled nucleotides. DNA extensions occur one nucleotide at a time (as opposed to 454 sequencing where all nucleotides of a homopolymer run are represented by one light signal). Current read lengths are around 100-150 bp. A detailed description of the technology can be found in Bentley *et al.* [16].

The SOLiD platform differs from 454 and Illumina in that it does not rely on sequencing-by-synthesis, but uses DNA ligase and complementary probes to sequence the amplified fragments [15]. It reaches read lengths of around 75 bp. The technology is presented in Valouev *et al.* [17].

All the technologies mentioned above have the advantage of being highly parallel and are therefore faster (and cheaper) than Sanger sequencing, and they yield significantly higher throughput (see Table 1.1). These massive amounts of data pose a challenge to the infrastructure of existing information technology systems, especially in terms of data transfer, storage, quality control, and computational analysis [18]. Roche's most successfully used 454 platform (GS FLX Titanium) yields about 450 Mbp (450,000,000 bp) of raw sequence, the latest platform GS FLX+ yields 700 Mbp, Illumina's HiSeq 2000 yields 600 Gbp (600,000,000,000 bp), and SOLiD's 5500xl system yields 100-160 Gbp (see Table 1.1). However, the rapid pace of NGS technology development suggests that these numbers will soon be outdated. Current Sanger sequencing, where readily produced sequencing products (96 at a time) are separated and detected from the capillary instrument, yields a throughput of 115 Kbp (115,000 bp) from one run [19]. Also, Ion Torrent semiconductor sequencing is usually counted as second-generation sequencing platform [20, 21]. Another, less common sequencing platform from this generation of sequencers is the multiplex polony technology, an open source platform with freely available software and protocols [22].

The sequencing process of each technology involves a number of methods that can be grouped into template preparation, sequencing and imaging, and data analysis. Metzker [18] provides a technical review of these stages for most of the platforms mentioned above, including graphical descriptions of template immobilization strategies, modified nucleotides used as reagents, and sequencing reactions. In addition, he discusses genome alignment and assembly approaches and gives an outline of NGS application areas. Similarly, Hutchison [23] reviews first-generation sequencing and discusses landmarks and application areas that can benefit from second-generation sequencing. Several papers provide useful overviews of second-generation sequencing techniques, their intrinsic characteristics, bioinformatic challenges, suitability for different applications and impacts on research [15, 19, 24–29].

When HeliScope launched the first single-molecule sequencing technology in 2008, a third generation of sequencers was born [30], and the term NGS was no longer referring to second-generation sequencers only. Schadt *et al.* [31] and Blow [32] provide an overview of HeliScope and other third-generation sequencers such as PacBio [33, 34] and nanopore [35, 36] sequencing. However, there is no consensus on what distinguishes second- from third-generation sequencing platforms. Throughout this thesis, all single-molecule sequencing platforms will be referred to as third-generation sequencing.

In line with this definition, one main difference between second- and third-generation techniques is that the latter do not require amplification of templates. Many artifacts and error patterns in 454 sequencing have to be seen in connection with emulsion PCR (emPCR) amplification (see Section 3.2.2) and the synchronized flowing of the amplified templates with reagents during the sequencing step. Consequently, the methods and algorithms developed in the context of this thesis are not directly applicable to single-molecule sequencing data. As described above, the sequencing step of other amplification-based technologies such as Illumina and SOLiD differs from the methodology used in 454 sequencing. In addition, the Illumina and SOLiD platforms use a different data output format than 454. Only Ion Torrent produces sequencing data similar to that of 454 and shares the 454-typical combination of emPCR amplification, sequencing-by-synthesis and expressing homopolymer runs in one number (rather than in one number per base) [20]. Unfortunately, this does not necessarily mean that all algorithms and tools targeted to 454 sequencing are applicable to Ion Torrent data, but further investigation strongly suggests itself.

1.2 Spoilt for Choice - Which Platform to Use?

When 454 sequencing was launched, limitations with respect to per-base costs, labor-intensiveness and speed were overcome, enabling large-scale and routine sequencing projects (e.g. for human genomes) [11, 37, 38]. Furthermore, templates could be handled in bulk within the emulsions which allows for massively parallel sequencing. Until then, large-scale sequencing projects had usually required the cloning of DNA fragments into bacterial vectors. Amplification and purification of individual templates was then followed by Sanger sequencing using fluorescent chain-terminating nucleotide analogues and either slab gel or capillary electrophoresis [10]. In both technologies, Sanger and 454, the target DNA is mechanically sheared into fragments of a few thousand base pairs (a few hundred for early 454 platforms). While Sanger requires subcloning into bacterial cells, most commonly *Escherichia coli (E. coli)*, in order to amplify the fragments, 454 can use the fragments directly. Consequently, 454 was the first technology that made subcloning in bacterial vectors superfluous, reducing putative contamination sources to a high degree [11]. In addition, the lack of a bacterial cloning step leads to a substantially more even coverage (see Section 2.3) in 454 sequencing data when compared to Sanger sequencing [15]. This was confirmed in several studies [10, 39, 40].

Several research groups have evaluated the extent to which 454 can outperform Sanger sequencing in different application areas and biological research fields [37, 41–46]. In projects involving *de novo* sequencing of complex genomes (which requires long reads in order to resolve repetitive regions, see Sections 2.4 and 3.4), short read lengths or high error rates compared to Sanger sequencing are still the main challenge, especially if no previously sequenced reference genome or draft assembly is available [47]. Nevertheless, Sanger sequencing is gradually being displaced by 454 sequencing and other NGS technologies.

However, researchers often face the decision of whether to use one of the short-read[3] platforms Illumina and SOLiD (producing a higher number of short sequences) or 454 sequencing (producing a lower number of longer sequences) for their projects. Third-generation sequencing is still less common, mostly due to the high costs of purchasing sequencing platforms and the relatively high error rates of these technologies (see Table 1.1).

Platform	**Read length (bp)**	**Run time**	**Throughput (Gbp)**	**Per-base error rate**
454 GS FLX Titanium	500	10 hours	0.45	see Sect. 4.1
454 GS FLX+	700	23 hours	0.7	see Sect. 4.1
SOLiD 5500 XL	75	7 days	100-160	< 1%
Illumina HiSeq 2000	150	11 days	600	< 1%
Illumina MiSeq	250	40 hours	8	< 1%
Ion Torrent PGM	250	2 hours	1	< 2%
PacBio RS	5,000	2 hours	0.1	10-20%

Table 1.1: Comparison of the most common NGS platforms [13, 18, 34, 48–50]. Read lengths are average estimates.

One of the most relevant applications for NGS is the resequencing of human genomes. Such high data volume applications require the detection of a large number of targets within one run (e.g. the sequencing of all genes for a single or even several individuals in parallel), providing a better understanding of how ge-

[3] The term short-read sequencing is, nowadays, mostly related to Illumina and SOLiD, but was often used when referring to the first 454 sequencing platform, GS 20. However, after 16 months on the market, 454 read lengths had increased from 100 bp to 250 bp [14]. The development of reads lengths is sketched in Section 3.4.

netic differences affect health and disease. Since almost all disease-causing genes of the human can be found in the exome, which only represents approximately 1% of the whole genome, exome sequencing at high coverage rather than whole genome sequencing (WGS) has evolved as a cost-efficient strategy in the context of genetic diseases or predispositions in humans [18, 51, 52]. Furthermore, analyses previously carried out using microarrays are more often being replaced with NGS-based techniques (e.g. in chromatin immuno-precipitation sequencing (ChIP-seq), DNase-seq, methyl-seq and ribonucleic acid sequencing (RNA-seq), see Section 2.1).

In 2008, the 1,000 Genomes Project was launched as an international collaboration between China, Germany, the UK and the USA. This collaboration represents an effort to sequence the genomes of at least 1,000 people from around the world ("a deep catalog of human genetic variation") [53]. By analyzing genetic variation and determining unobserved genetic variants, researchers can deepen our knowledge of evolutionary processes – e.g. to investigate the relationship between genotype (internally coded, inheritable information) and phenotype (observable characteristics). Another main goal is the identification of disease-causing genes, which allows for future clinical applications such as the prediction of disease susceptibility and drug response. The genomes of approximately 2000 individuals from different continents were collected, in some cases from both parents and an adult child. Using these samples, different strategies and platforms (454 sequencing, Illumina and SOLiD) for WGS were applied and compared [54–56].

Also, systematic benchmark studies of the different NGS platforms have been carried out, targeting SNP identification [57], variant detection [58–60], microbial diversity [61, 62], and transcriptome sequencing [63]. The contrasting features of newer technologies make it likely that NGS platforms will not only coexist and be applied in their respective strongest application areas, but will also be combined in a way that yields more accurate results than would be possible when relying on one technology alone. For example, the relatively long reads obtained from 454 sequencing can be complemented with the relatively cheap reads generated on Illumina or SOLiD platforms.

In this context, it is worth mentioning recent developments in PacBio single-molecule sequencing, which have demonstrated unprecedented read lengths of up to 20,000 bp at an average of 5,000 bp [34]. Despite the high per-base error rate (1̃0-20%), PacBio's strength clearly lies in its extraordinarily long reads and random error distribution. This enables the resolution of genetic complexity in applications such as finishing of draft genomes or in resolving genomic variation over long distances [34, 64]. Bashir *et al.* [65] present a hybrid assembly approach, which combines sequencing data from second-generation platforms and PacBio in genome assembly of a cholera strain responsible for the 2010 Haitian outbreak. Other research groups have aligned PacBio reads to previously published draft assemblies [47] or to high-fidelity sequences from other NGS platforms such as Illumina and 454 [66].[4] All these approaches are compromises to address the low quality of the PacBio reads in order to take advantage of their length. In draft assemblies that have been created using other platforms, accuracy is usually high, such that PacBio reads can close gaps in the original assembly [47].

Last, it is worth mentioning that much of the Sanger data have been deposited into databases and archives over the decades [67]. The co-existence of sequencing methods from all three generations and the large amount of data accessible in public databases make it possible to assess both the accuracy of newer data and the correctness of reference sequences in databases [14, 58].

[4] This latter strategy may, however, fail in repeat regions. In contrast, the approach is assumed to be useful in assemblies with polymorphic input data. Both factors are major causes of gaps in *de novo* genome assemblies (see Section 2.4) [47].

2

454 Sequencing - Milestones and Applications

"How many species inhabit our immediate surroundings? A straightforward collection technique suitable for answering this question is known to anyone who has ever driven a car at highway speeds."

Kosakovsky Pond *et al.*: Windshield splatter analysis (...) [68]

In the beginnings of 454 sequencing, the technology was mainly used for resequencing known whole genomes or DNA target regions (see Section 2.4.1) and for complementing Sanger sequencing projects [14]. Longer reads, higher accuracy and the launch of the paired end feature (see Section 2.2) have since paved the way for 454 sequencing to supplant Sanger sequencing in some application areas such as *de novo* WGS, metagenomics and RNA analysis [38].

2.1 An Overview of Applications and Fields

To date, there are thousands of scientific papers describing the application of 454 sequencing in different biological fields (see Figure 2.1). Massively parallel sequencing allows for large-scale SNP discovery, e.g. in the context of disease-associated SNPs [19]. Ever since the first human genome was sequenced (see Chapter 1), researchers from many laboratories have tried to map haplotype diversity in the human

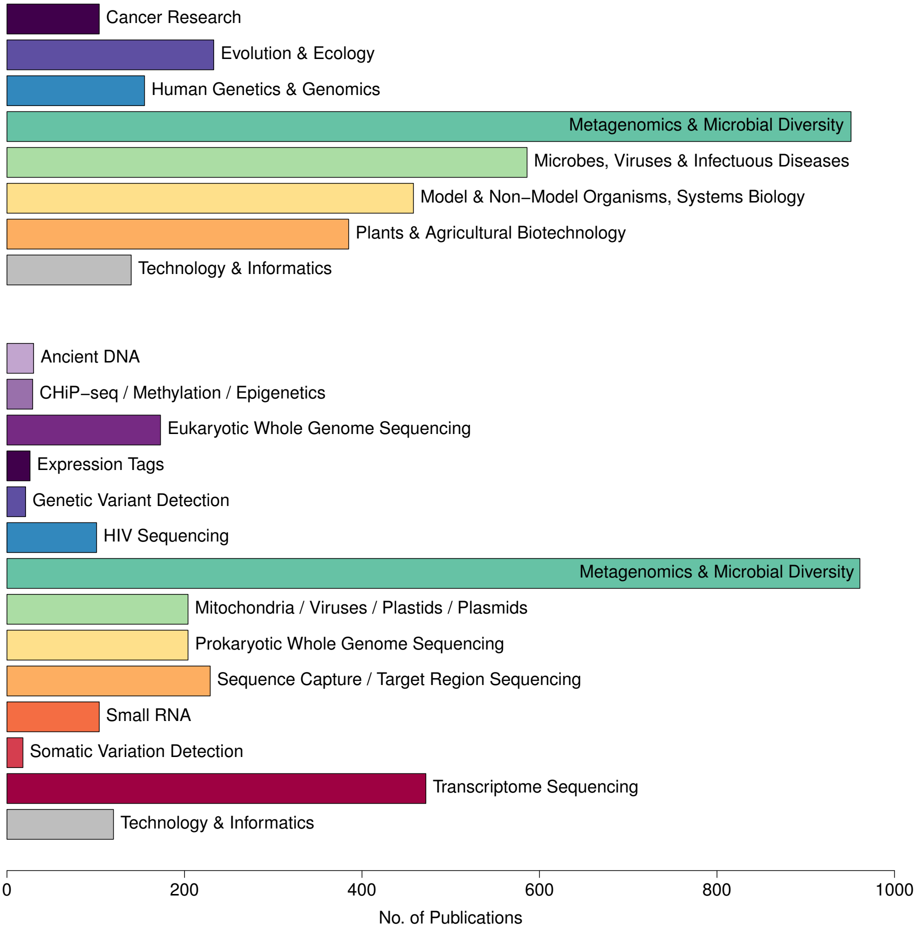

Figure 2.1: Publications enabled by 454 Sequencing technology (until January 2013). **Top:** By biological field. **Bottom:** By application. One publication can be assigned to several biological fields or applications. Numbers taken from the 454 website [13].

genome and have to date identified almost 40 million[5].

Beyond SNP detection, variant analyses further include structural variants. These are defined as all variants other than SNPs or small insertions/deletions, namely larger, often kbp- to Mbp-sized deletions, insertions (novel genome content) or inversions (changes in the orientation of segments), substitutions, segmental duplications and complex combinations of rearrangements.[6] Such variants represent the molecular basis for genomic variations [18, 19, 71].

ChIP-seq comprises methods for measuring genome-wide profiles of DNA-protein complexes and in the past was performed by microarray hybridization (ChIP-chip) [72]. Interactions between DNA and proteins play a key role in regulating gene expression and controlling transcription, replication etc. DNase-seq is used to identify DNase I hypersensitivity sites (open chromatin) [15, 18]. Methyl-seq is a method of great importance to epigenetics, a field that comprises the analysis of heritable gene regulation not encoded in the DNA sequence. One task in this context is to detect patterns of abnormal methylation (a biochemical process that influences gene-expression), associated with diseases such as cancer [73]. RNA-seq helps to determine gene expression and entails the sequencing of RNA templates converted to complementary DNA (cDNA) since second-generation sequencing technologies are incapable of sequencing RNA directly. RNA-seq commonly comprises messenger RNAs (mRNAs), non-coding RNAs and small RNAs. Several papers discuss NGS techniques for non-genomic applications including transcriptomics (aiming to determine the set of all mRNA molecules – and their abundances – in a sample) [73–76].[7]

[5] number of validates SNP clusters from dbSNP summary build 137 [69, 70] common SNP positions

[6] Sharp *et al.* [71] define fine-scale structural variations as variations spanning a size range of 50 bp to 5 kbp. Intermediate-scale variations reach from 5 kbp to 50 kbp, large-scale variations from 50 kbp to 5 Mbp. Even larger variations are defined as chromosomal variation.

[7] In third-generation sequencing, however, RNA can be sequenced directly, which promises higher accuracy since the conversion of cDNA to RNA can be omitted [31].

454 sequencing has also been successfully used in palaeogenomics (the study of ancient DNA, e.g. neanderthals and mammoths), reviewed by Millar *et al.* [77] and Pruefer *et al.* [78].

2.2 Shotgun vs. Paired End Reads

Shotgun sequencing describes the process of randomly breaking up DNA into numerous small segments of approximately the same size. The use of the term paired ends, on the other hand, varies across sequencing technologies. In Illumina sequencing, paired ends come from the same contiguous DNA molecule which is sequenced from both ends, the distance between the ends being user-definable (100-500 bp). In 454 sequencing, the terms “mate pairs” and “paired ends” are used interchangeably, but they are targeted to large insert sizes and follow another scheme (similar to what Illumina calls mate pairs).[8]

Paired end reads[9] were introduced with the FLX platform and are generated as follows (see Figure 2.2): As with shotgun sequencing, genomic DNA is sheared into fragments which follow a tight fragment size distribution according to the chosen insert size. Commonly used insert sizes are 3 kbp, 8-10 kbp or 20 kbp. After fragmentation, a linker sequence is ligated to the end of each fragment. The DNA is then circularized, and the ends are attached to each other by means of the linker. This allows for sequencing of the ends of the original molecule, producing paired reads originating from the same molecule and separated by a known distance, with a linker sequence (44 bp in Titanium) in between. As an example, for 3 kbp paired end reads, the flanking sequences to both sides of the linker are DNA segments that were originally located approximately 3 kbp apart in the genome of interest [79, 81, 82].

[8] The small distances of Illumina paired ends provide tighter insert-size distributions, and thus higher resolution, when compared to 454 paired ends (or mate-pairs in Illumina) which have the advantage of larger insert sizes and thus the ability to bridge long repetitive sequences [79].

[9] sometimes also referred to as “jumping library”

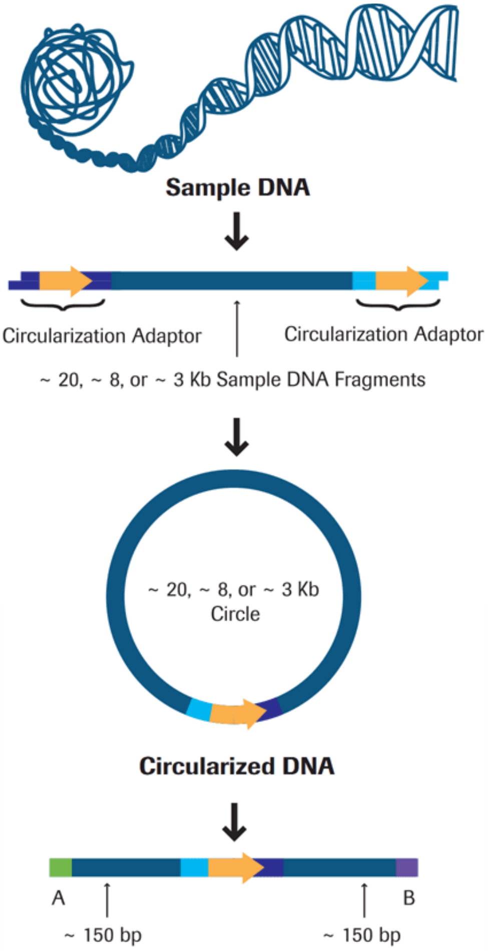

Figure 2.2: The GS FLX Titanium Series Paired End Protocol, taken from the 454 website [80].

The circularized fragment is then randomly sheared, and segments containing the linker are purified. Finally, paired end reads are generated by sequencing through the linker. When producing a sequencing run according to this 454 paired end protocol, a certain percentage (often not less than 50 %) of reads turn out to be missing the linker sequence ("linker-negative"). Only linker-positive reads (input DNA – linker – input DNA) that contain at least 15 bp of input DNA on each side of the linker sequence can be used as paired end reads, the remaining reads can be treated as ordinary shotgun reads [81, 83, 84].

In *de novo* genome assembly (see Section 2.4), paired ends make it possible to determine the relative positions and orientation of contigs that have been created during the assembly process of shotgun reads, but also to bridge repetitive sequence stretches. Another application is the identification of large-sized structural variants (see Section 2.1) by mapping paired end reads onto a reference genome. Korbel *et al.* [85, 86] use this technique in a human diversity study, where structural variation is presumably responsible for a considerable amount of phenotypic variation. Fullwood *et al.* [84] provide an extensive retrospective of applications making use of this so-called paired end tag (PET) sequencing strategy, emphasizing its broad application area.

2.3 Sequencing Coverage

The common use of the term coverage[10] indicates the average number of reads covering each base in the reconstructed sequence (e.g. 40 X coverage). In contrast, one can sometimes find a percentage coverage in literature. This refers to how well the genome or reference sequence is covered after a mapping or assembly process.[11] Throughout this thesis, sequencing coverage is defined according to the first description.

One crucial part of the study design in WGS with respect to time and budget considerations consists in deciding the minimum amount of sequence information that is required for an assembly of a certain quality, i.e. for obtaining an accurate assembly that represents the target genome to a high degree [10]. For a low-quality draft, such as a comparison with a readily finished reference genome or in a rapid

[10] often also referred to as depth, depth coverage or per-base (sequencing) coverage

[11] As an example, the human genome has a size of 2.85 Gbp, where 99% of the genome could be assembled leaving 341 gaps. The consensus error rate is 1 per 100,000 bp (which equals to a consensus accuracy of 99.999%) [18, 87].

response scenario [88], low coverage may be sufficient. Finished-grade genomes are of higher quality than draft-grade genomes because higher base coverage leads to higher consensus accuracy and often fewer gaps. In brief, a saturating level of sequence coverage implies that further increasing of coverage would have minimal, if any, effect on data quality and downstream analyses. Strong variation in coverage is not only wasteful for the overall sequencing yield, but also decreases the expected average coverage of a sequencing project. In other words, a more uniform coverage results in higher performance at lower coverage [58].

Wendl [89] proposes a method for modeling coverage distributions in WGS projects. Other parametric approaches include calculating the redundancy required to detect (a certain percentage of) sequence variations [58, 90]. Estimating the required sequencing coverage is even more challenging in metagenomics, where genomes from multiple species are simultaneously sequenced, such that obtaining large numbers of reads per genome is unlikely. In addition, species do not have uniform abundance in a community. However, there are approximations and rules of thumb that have been verified in simulation studies and metagenomic experiments [91–94].

Some DNA products (see Sections 2.5 and 3.2.1) require amplification via PCR prior to the actual sequencing process. This can cause a strong coverage bias (see Section 4.2.2), putatively leading to incorrect conclusions in downstream analysis. Also, the presence of artificial duplicates (see Section 4.2.5) can generate uneven coverage if those are not removed by data cleaning tools. The impact of coverage bias on sequencing analysis is extensively discussed in Chapter 4.

2.4 Genome Assembly

The problem of "genome assembly" arises from the fact that genomes often contain millions of base pairs but current genome sequencers only produce relatively short reads (under 1,000 bp).[12] The process of WGS includes fragmenting the genome (see Section 3.2.1), sequencing the fragments (see Section 3.2.3) and re-assembling them in order to obtain the full genome sequence. It is not uncommon that several billion reads are required for a genome assembly. Nevertheless, most published assemblies still contain gaps, i.e. undetermined regions.

In genome assembly, algorithms are used to align overlapping reads based on sequence similarity, so that the original genome is represented by sets of contiguous (i.e. gap-less) sequences, so-called contigs [31, 82]. The consensus sequence of a contig is determined either by the highest-quality base or based on majority rule (the most frequently encountered base) at each position [91].

Contigs can be ordered, oriented and placed in larger structures called scaffolds with the help of paired end reads (see Section 2.2) that are present in two different contigs. Hence, a scaffold is a sequence of contigs in the (presumably) correct order, where the size of the gaps between the contigs ("intercontig gap size") is unknown but can be estimated from the insert size of the paired ends [96, 97].

Factors that influence the feasibility and quality of an assembly and the number of remaining gaps are, amongst others, read lengths (too short reads provide too little sequencing context), the type of library (shotgun only / both shotgun and paired ends), sequencing depth (i.e. sequencing coverage, see Section 2.3), contamination with foreign and adapter sequences (see Section 4.2.1), a high level of polymorphism and, most importantly, the repeat content of the organism. Repeat content refers to duplications within the genome, i.e. large regions that are highly similar

[12] PacBio produces longer reads up to several tens of thousands of bp, but at a high error rate (see Section 1.2).

Figure 2.3: “I think I found a corner piece.” [95]

to other regions as they occur in almost every organism, but also to low-complexity regions.[13]

Longer reads are more likely to be uniquely placed onto a genome making assembly more straightforward. Highly repetitive genomes, however, are harder to assemble since repeats confuse the assembly process [98].[14] They can often be detected by looking for regions of unusually high sequencing coverage. A common strategy for improving the overall quality of an assembly is to increase read coverage (see Section 2.3) [18], but this often proves ineffective in repetitive regions. The key parameter in enhancing the efficiency to sequence and assemble stretches of repetitive DNA is to reduce the number of identical reads by increasing read length to better reach through repeats, or by sequencing smaller parts of the genome with

[13] The human genome, as an example, has a repeat content of approximately 45% [58].

[14] In transcriptome sequencing (see Section 2.1), the reduced amount of repetitive DNA compared to non-coding regions facilitates *de novo* assembly of 454 reads [99].

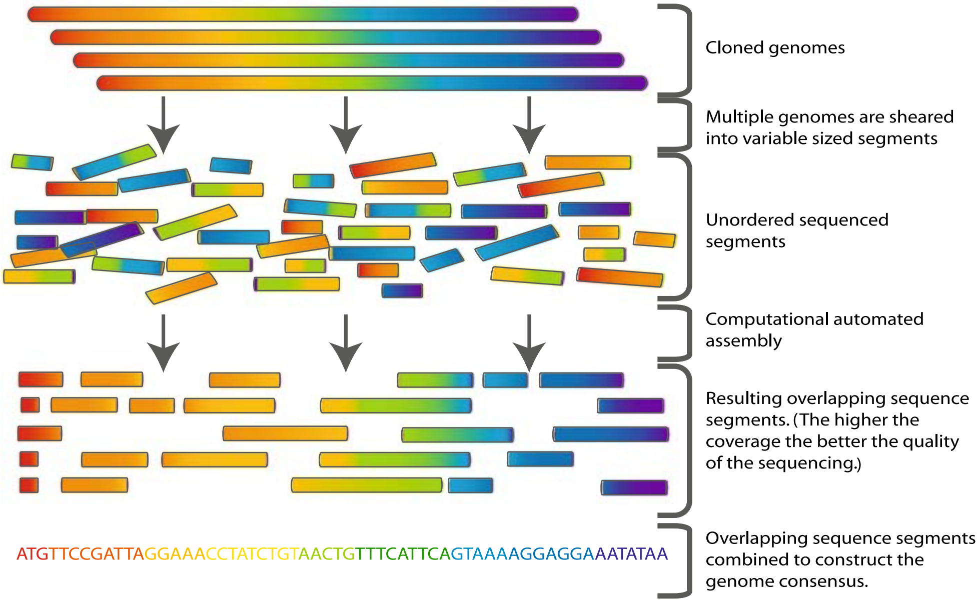
Cloned genomes
Multiple genomes are sheared into variable sized segments
Unordered sequenced segments
Computational automated assembly
Resulting overlapping sequence segments. (The higher the coverage the better the quality of the sequencing.)
ATGTTCCGATTAGGAAACCTATCTGTAACTGTTTCATTCAGTAAAAGGAGGAAATATAA
Overlapping sequence segments combined to construct the genome consensus.

Figure 2.4: Whole genome assembly: The genome is sheared into small approximately equally sized fragments which are subsequently small enough to be sequenced. The resulting reads are then fed to an assembler. Taken from Commins *et al.* [101].

hierarchical template sizes, e.g. plasmids, fosmids or bacterial artificial chromosomes (BACs, see Section 2.4.1 and Figure 2.5) separately [11, 82, 100].

2.4.1 Approaches and Issues

There are several different approaches to the assembly of large genomes, most of which were developed during the Sanger era and adapted to newer technologies. The decision of which approach to choose is based on the biological application as well as on cost, effort and time considerations [18]. Similarly, the decision whether or not to include paired end reads (see Section 2.2) and if so, which insert sizes to use, depends on the size and complexity of the genome, but also on the purpose of the project. For a quick overview of a genome, e.g. for identifying which genes are present, a shotgun-only draft assembly may be sufficient, while a high-quality draft or finished-grade assembly will require a combination of shotgun and paired end reads [81].

Today, the fastest and most cost-effective and therefore most common sequencing strategy is the so-called “whole genome assembly” (see Figure 2.4). Unfortunately, it is also the most error-prone strategy since the genome is assembled blindly to any data beyond the sequence reads (“*de novo*”).

Another common approach is a hierarchical strategy where the global problem of assembling the whole genome is reduced to many local assemblies (see Figure 2.5). This approach is often referred to as “clone-based” because it involves splitting up the genome into BACs of approximately 80-200 Kbp size each [96, 102]. Together, the clones can be used to calculate a path through the genome. The clones themselves are sequenced by shotgun sequencing. This approach has the advantage of limiting assembly errors to local assemblies [31, 96].

In case there is a reference genome, a comparative, reference-based assembly can be carried out, where a reference genome of a preferably very closely related organism is used to guide the assembly. Obvious issues arise when the reference genome is not closely related enough or in regions of high structural variation [82]. This assembly approach can be seen as one of the applications of resequencing (see Section 2.1) and requires much less coverage than *de novo* whole genome assemblies [14].

2.4.2 Assembly Tools for 454 Reads

There are a number of assemblers that are capable of either assembling 454 reads or, in a hybrid approach, combining 454 data with those from other technologies (i.e. with other NGS data and/or Sanger reads). Examples of such hybrid genome assemblies can be found in literature [65, 103–106].

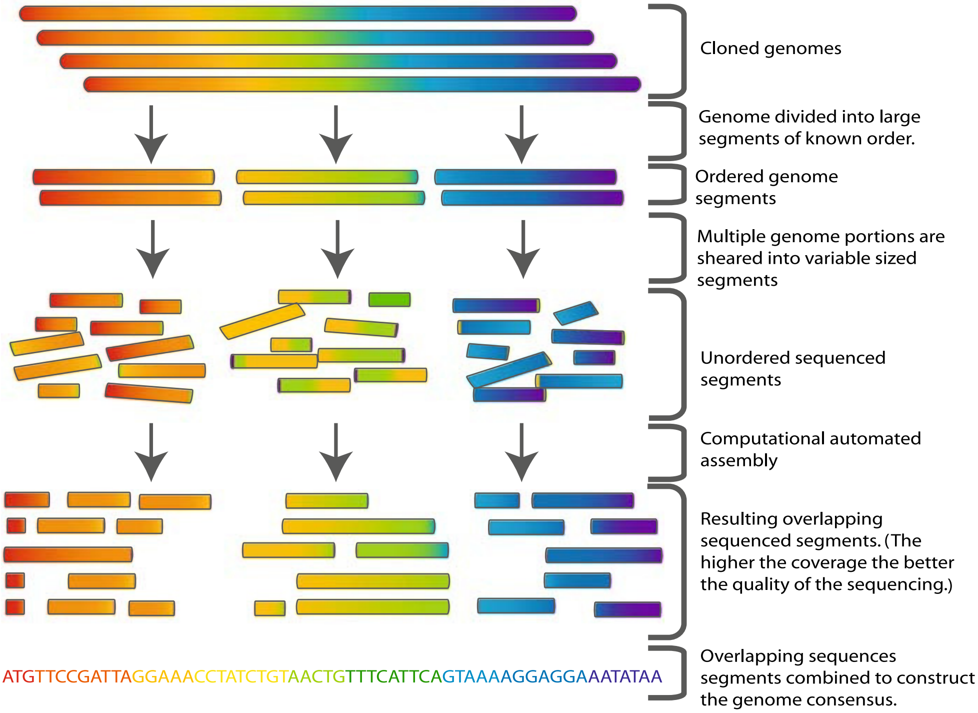
Cloned genomes
Genome divided into large segments of known order.
Ordered genome segments
Multiple genome portions are sheared into variable sized segments
Unordered sequenced segments
Computational automated assembly
Resulting overlapping sequenced segments. (The higher the coverage the better the quality of the sequencing.)
ATGTTCCGATTAGGAAACCTATCTGTAACTGTTTCATTCAGTAAAAGGAGGAAATATAA
Overlapping sequences segments combined to construct the genome consensus.

Figure 2.5: Hierarchical assembly: The genome is broken into a series of approximately equally sized, large segments of known order which are then subject to shotgun sequencing. This makes the assembly process simpler and less computationally expensive. Taken from Commins *et al.* [101].

Assemblers can be divided into two major classes: Those that use a so-called overlap/layout/consensus (OLC) approach, and those that make use of De Bruijn graphs (DBG). Assemblers using the OLC approach are optimized for assembling large genomes and follow three phases: overlap, layout, and consensus. First, the overlap between all sequences is calculated, then, the reads are arranged according to their overlap (layout step). In the consensus step, a contig is calculated from the consensus bases at each position. If the sequencing library also contains paired end reads (see Section 2.2), these allow the contigs to be placed into scaffolds. Reference-based assemblies (see Section 2.4.1) omit the overlap step, and scaffold building is not necessary since the reference genome is assumed to have the same genome structure.

Both Newbler [10], the assembler sold with the 454 platform, and CABOG [83], an extended pipeline of the previously published Celera assembler [12, 107], are OLC assemblers that are capable of combining Sanger and 454 reads and, in addition, allow the inclusion of paired ends. Also, PAVE [108] and iAssembler [109], building on the CAP3 assembler [110] can combine Sanger and 454 data. MIRA[111–113] can even create hybrid *de novo* assemblies from Sanger, 454, Illumina, Ion Torrent and PacBio data.

DBG assemblers are specifically targeted to short-read technologies that do not require aligning all reads against all [114, 115]. Examples are Euler-SR [116] and Velvet [117, 118].

A number of research groups have carried out comparisons and benchmarks on the performance of different assemblers and algorithms [83, 119–123]. Schatz *et al.* [124] review assembly algorithms and genomes assembled with NGS data and discuss the tradeoff between read length, coverage and expected contig length.

2.4.3 Assembly Quality

Common ways of evaluating the quality of an assembly in an assembler-independent way focus on assembly size and fragmentation, pursuing a “the bigger the better” approach. Such metrics take into account the size of the assembly, the sizes and numbers of contigs and scaffolds, the size and number of gaps and often the N50 statistic.

The contig N50 is the length of the smallest contig in the set that contains the largest contigs whose combined length represents at least 50% of the assembly, which means that using equal or longer contigs produces half the bases of the assembly. The contig N50 thus provides a measure of connectivity [114].[15]

However, judging assemblies only by size is misleading since large contigs can be the result of any arbitrary assembly. In consequence, several research groups have developed more sophisticated sets of metrics and software pipelines for measuring assembly accuracy and detecting mis-assemblies [100, 125–127]. Although most of these strategies require a reference genome, they are extremely useful for selecting a sequencing strategy and tuning assembly parameters when resequencing a finished reference genome. Also, sequencing the first human genome (see Chapter 1) has revealed the high cost of genome finishing. Assessing assembly quality and detecting mis-assemblies is a step towards sequencing genomes to more than a draft level in a more automated way than before [100].

Phillippy *et al.* [100] define two categories as the source of most mis-assemblies: Repeat collapse and expansion, and sequence rearrangement and inversion. Similarly, Haiminen *et al.* [126] introduce a scoring system that – through realignment with a reference genome – captures to what extent an assembly is correct or erroneous. The five independent characteristics that are integrated into the overall assembly score are: relocation (incorrect order), inversion (incorrect orientation), redundancy (insertions/duplications), match (reward for long matches and penalty for gaps) and the percentage of the reference sequence covered. Interestingly, Haiminen *et al.* found paired end reads to improve size statistics, but not necessarily correctness of assemblies.

Furthermore, Phillippy *et al.* [100] point out how collapsing or expanding reads during genome assembly can inflate or deflate the density of reads and thus directly influence coverage (see Section 2.3). Since these peaks or valleys in coverage strongly deviate from the coverage expected from a random shotgun process, they

[15] Often, the N50 is calculated based on large contigs (spanning at least 500 bp) and therefore biased. A more meaningful measure that permits fair comparisons between assemblies is the NG50 that uses genome size instead of assembly size [120].

can be used to identify mis-assemblies. The reliability of this method depends on a couple of factors such as the evenness of coverage (see Section 2.3).

2.4.4 First Genome Assemblies with 454 Sequencing

The first genomes sequenced with 454 technology were the 600,000 bp genome of the bacterium *Mycoplasma genitalium* and the 2.1 Mbp genome of *Streptococcus pneumoniae*, published with the launch of the first 454 sequencing platform in 2005. Margulies *et al.* [10] demonstrated the efficiency of their newly developed platform by *de novo* sequencing the genomes and comparing their assemblies to the previously published reference.

Wicker *et al.* [42] were the first to perform a study on the technological challenges posed by sequencing complex (i.e. large and highly repetitive) genomes with 454 when compared to Sanger sequencing. Earlier, such studies had only been carried out on compact microbial genomes with low repeat content [128, 129]. For sequencing the barley genome Wicker *et al.* chose a hierarchical sequencing approach (see Section 2.4.1) using BAC clones. They concluded that 454 sequencing allows for high-quality and cost-effective sequence assembly while providing a more even coverage than Sanger sequencing (see Section 1.2). Consensus accuracy (see Section 4.1) was found to be comparable in 454 and Sanger sequencing. Problems arose in repetitive DNA regions, but one has to keep in mind that the study was performed on GS 20 data, i.e. with read lengths of 100 bp on average, while Sanger yields read lengths of 800 bp. A similar study, also on barley, was later on carried out on FLX data [130].

In 2008, one of the two research groups who had earlier published the first human genome – the genome of J. Craig Venter [7, 8] – presented the complete genome of a second individual, James D. Watson [131]. The genome, approximately 3 Gbp large, was sequenced with the 454 technology [38].

The same year, Quinn *et al.* [37] performed a feasibility study on *de novo* sequencing pooled BACs of Atlantic salmon (*Salmo salar*) with 454 data only. Atlantic salmon is of high importance in aquaculture and can be seen as a model organism for studying evolutionary processes. However, no closely related fish had been sequenced before. The genome – estimated to approximately 3 Gbp of size – contains 30-35% of repeat content and, in addition, whole genome duplication [132], making sequencing and assembly extremely challenging. Quinn *et al.* used both shotgun and paired end reads (see Section 2.2), the latter enhancing assembly quality tremendously. Although they used FLX reads, Sanger sequencing was found to be superior to 454. The project highlighted the utility of 454 shotgun sequencing for gene discovery and identified read length as the main factor limiting assembly quality, especially in repeat regions. Even after the release of the Titanium technology and a new feasibility evaluation, 454 reads were not found to be sufficiently long for a *de novo* assembly of this complexity (especially with respect to the repeat content of the genome), which is why the sequencing project was further carried out using Sanger technology, supplemented by Illumina and PacBio reads [132].

In 2009, the 0.83 Gbp genome of Atlantic cod (*Gadus morhua*) was sequenced exclusively with 454 data, both Titanium shotgun reads and paired end reads of four different insert sizes, assembled with a WGS approach. Both the Newbler and the Celera assembler produced assemblies with scaffolds of comparable size with Sanger assemblies. Within the project, different assembly strategies and assemblies were tested and benchmarked [97, 133].

2.5 Metagenomics

Microbial diversity on the Earth is largely unexplored [92]. Unlike traditional microbial sequencing, metagenomics study microbial communities or genomic content of a sample of organisms that has been obtained *directly* from their natural environment, bypassing the requirement for prior culturing. This enables the study of the more than 99% of microorganisms which cannot be isolated or are difficult to grow in a lab [134, 135]. Sequencing a “metagenome” involves the direct determination of the whole collection of genomes within an environmental sample as well as studying biochemical activities and interactions between community members [135, 136]. In brief, characterizing the organisms present in a sample and quantifying the taxonomic composition of environmental communities is an important indicator of their ecology, function and evolution. Together with metagenomic studies, also metatranscriptomics [137] and metaproteomics help to explore the organization and function of microbial communities [135].

The study of metagenomics is applicable to many fields including ecology and environmental sciences, chemical industry, and human health (e.g. the human gut) and comprise a large range of analyses: assembly and gene prediction, characterization and quantification of microbial diversity, function prediction, comparative metagenomics, modeling interactions between microbes and their environment etc. Wooley and Ye [135] extensively discuss these topics, including a review of computational and statistical tools for metagenomic analysis and an overview of known artifacts caused by limitations in the experimental protocol.

In a typical metagenomic project, workflow steps involve sample and metadata collection, DNA extraction, library construction, sequencing and read preprocessing before moving on to finding answers to the questions “Who is out there?” (taxonomical binning), “How many are there?” (quantitative analysis) and “What are they doing?” (functional binning) [138].

The extraction and purification of sufficient quantities of DNA is often difficult because it must be acquired from low-biomass samples, an issue that is overcome by PCR-amplification. However, PCR often introduces bias (see Section 4.2.2), which in consequence means that the relative representation of DNA fragments is likely to be biased, especially if the amount of starting material is small [91].

In addition, detecting highly abundant organisms requires considerably less sampling than the identification of rare organisms. Despite this fact, sampling effort is often influenced by research budgets and technologies rather than by significance aspects regarding diversity or the ability to detect organisms. With the help of taxa-abundance distributions and statistical methods, one can calculate the sampling material and sequencing effort required to obtain a given fraction of the diversity present in a sample [92].

Common approaches for obtaining a metagenomic or microbial library for sequencing are the large-scale shotgun technique (e.g. for sequencing a metagenome) or phylogenetic marker genes such as the small subunit (16S) rRNA gene. This gene is widely used in community analysis because it is present in all organisms and, in addition, has both slow- and fast-evolving regions [139]. It allows for reliable reconstruction of phylogeny and provides measures of richness and relative abundance of species in microbial communities [140]. Examples for the early use of 454 pyrosequencing in metagenomics are studies of viral [141] and bacterial [128] communities and the use of PCR-amplified 16S rRNA genes to evaluate community composition [140, 142, 143].

Pre-processing of reads from metagenomic data sets prior to metagenome assembly, gene prediction and annotation is similar to pre-processing of reads in a *de novo* genome sequencing project (see Section 2.4). It usually comprises adapter and contaminant removal, quality-trimming to remove low-quality bases and further quality-filtering steps as described in detail in Chapter 4.

“Who is out there?”

Taxonomical binning involves clustering metagenomic sequences into different bins, also referred to as operational taxonomic units (OTUs) that correspond to species/ organisms or taxa (populations of organisms). Binning can be carried out from assembled contigs, single reads, or both [91].

The assembly of fragments from highly diverse ecological systems to obtain a metagenome (i.e. a mixture of multiple genomes) is challenging [144], both because the arrangement of reads into contigs fails and because contigs are created that contain reads from many different genomes (interspecies chimeras, see Section 4.2.3) [138, 145]. None of the assemblers presented in Section 2.4.2 address these problems.

There are a variety of methods and computational tools that infer species information directly from reads without the need for assembling them first [135], e.g. DOTUR [146] or MEGAN [136, 147, 148]. Such tools calculate and explore the taxonomical content of a data set, either by using BLAST and other comparisons against databases in order to assign reads to known taxa/species or by building clusters of sequences that do not differ by more than a certain percentage. Often, a threshold of 3% is chosen, i.e. a sequence identity of 97%. This threshold corresponds to what has earlier been observed to produce OTUs that are representatives of taxa [92, 140]. From the observed frequencies of OTUs[16] or from species abundance curves, one can then predict the number of different microbial taxa in a sample [135, 140]. This reveals that taxonomic binning strongly depends on the chosen similarity threshold and, in addition, can be compromised by poor data quality.

In 2006, Sogin *et al.* [140] reported that microbial diversity in the deep sea is one to two orders of magnitude more complex than previously assumed. They found thousands of low-abundance taxa to account for a high percentage of the observed

[16] often based on differences in regions of the 16S rRNA gene

phylogenetic diversity (the “underexplored rare biosphere”). This study triggered a vivid discussion on the impact of 454 sequencing errors on diversity estimates, which will be further discussed in Section 4.3.8.

“How many are there?”

Once the diversity in a microbial community has been identified, a common task is to quantify the relative abundances of taxa and estimate the amount of sequence information for which no species have yet been described [136]. However, biases introduced through artificial duplicates often lead to incorrect conclusions about the abundance of species in microbial communities. This issue is extensively discussed in Section 4.2.5. Also, PCR-induced bias (see Section 4.2.2) can skew estimates of community composition.

“What are they doing?”

Functional binning refers to identifying potential protein functions and metabolic pathways, the latter being important for growth and survival of organisms in any given environment [149]. Methods for metagenomic gene prediction and their robustness with respect to sequencing errors are extensively discussed in Johnson and Slatkin [150] and Hoff [151], concluding that the integration of error-compensating methods into such tools may significantly improve performance and annotation quality. It is worth mentioning that the intrinsic error pattern of 454 sequencing – indels representing a majority of base-calling errors – affects gene prediction to a higher degree than technologies where substitution errors are the main issue. This is due to statistical gene prediction tools utilizing codons to identify protein coding genes [151].[17]

[17] Substitution errors, in constrast to indels, do not cause shifts in the reading frames and only affect one codon, which means that they are less likely to accidentally introduce a stop codon. In consequence, their influence on gene prediction accuracy is considerably smaller.

3

454 Sequencing – The Basics

"One late afternoon in the beginning of January 1986, bicycling from the lab over the hill to the small village of Fullbourn, the idea for an alternative DNA sequencing technique came to my mind. It was late, dark, and rainy as I hurried home to tell my wife Maija about the new idea. She later told me that when I explained the new idea to her, she thought that I looked like Gyro Gearloose's little helper – the bright-headed assistant with a light bulb as a head. I had difficulty sleeping that night and was eager to go home to Sweden to test my new idea. What I could not expect that day was that 10 yr would pass before the method was fully developed."

Pål Nyrén: The history of pyrosequencing [152]

What Pål Nyrén had envisioned on that winter afternoon was the underlying mechanism of a method that would later become known as pyrosequencing. However, due to both funding and technological issues, it took more than ten years until the method was fully developed – and almost another decade until it was brought to market [10, 152].

3.1 What is Pyrosequencing?

While working with traditional Sanger sequencing during his post-doctoral period in Cambridge around 1986, Nyrén had felt the need for a more automated and efficient DNA sequencing method. As a newcomer to Sanger sequencing, he was struggling with the handling of the reagents (e.g. the thin acrylamide gels). Sequencing was, at that time, a time-consuming business, involving several steps that required weeks to learn. Nyrén was experienced with the modification and simplification of methods from his PhD. He had worked earlier with pyrophosphate detection in another context and came across the thought of using this method for indication of base incorporation during DNA synthesis. The basic idea was to detect the released pyrophosphate during the DNA polymerase reaction, which is followed by a cascade of enzymatic reactions, amongst those the conversion of pyrophosphate to adenosine triphosphate (ATP) by sulfurylase and the production of light by firefly luciferase [152, 153]. Then, the light intensity is recorded with the help of a camera device. When nucleotide reagents are added sequentially and in a fixed order, the sequence can then be deduced by making use of the Watson-Crick base pairing rules (A binds to T, G binds to C).

Much later, Nyrén discovered that other researchers had, at approximately the same time, developed and published a similar approach, which had been patented as *sequencing-by-synthesis* in 1985 but was too insensitive for DNA sequencing [154, 155]. In contrast, Nyrén found his firefly-luciferase-based method to work. Together with Mostafa Ronaghi and other researchers in the field, he started a long process of optimizing the method. The first success came with a three-enzyme solid-phase pyrosequencing system [156–158] – a technique where templates are attached to magnetic beads – which, at that time, gave them read lengths of 15 bp. For the first time, it seemed realistic to envision a cost-efficient, highly parallel and automated DNA sequencing process without the need for electrophoresis. However, one of the main drawbacks was the necessity of a washing step after each nucleotide addition in order to remove the excess reagent [152, 153, 158].

An apparent breakthrough came with the addition of a fourth enzyme, apyrase, to the enzyme mixture. Apyrase was chosen due to its ability to degrade nucleotides, which suggested its use instead of the washing step. The more efficient the apyrase, the less background signal there is. This not only eliminates the washing step, but also the need for solid support, thus called *liquid-phase pyrosequencing*. Unfortunately, by-product accumulation due to the lack of a washing step was found to limit read length [159]. Both systems, the three-enzyme solid-phase pyrosequencing and the four-enzyme liquid-phase pyrosequencing, were observed to have their strengths and weaknesses. These and other issues in sequencing-by-synthesis have been extensively discussed [10, 153, 159, 160]. Today's 454 pyrosequencing follows the principles of solid-phase pyrosequencing, i.e. it builds on a three-enzyme system and involves a washing step after each nucleotide addition – using apyrase. In other words, pyrosequencing can be defined as a "non-electrophoretic, bioluminescence method that measures the release of pyrophosphate by proportionally converting it into visible light using a series of enzymatic reactions" [18] (see Figure 3.1). The reactions can be modeled with the aid of mathematical/stochastic processes, which not only helps understanding but is also used for improving the pyrosequencing process in terms of substrate concentrations or enzyme choice [10, 153, 161].

The main challenge for pyrosequencing was and is to increase throughput – especially in terms of read lengths – while maintaining reliability and accuracy. The two central factors that still inhibit the system from performing longer reads accurately are uncertainty in homopolymeric regions and loss of synchronism (see Sections 3.2.4 and 4.2.7). In long homopolymeric regions, the number of nucleotides is hard to determine, resulting from a broadening of signal distributions (see Figure 3.5) [10, 18, 138, 161, 162]. Loss of synchronism occurs when some of the templates on each bead get ahead of (carry forward) or behind (incomplete extension) the templates during nucleotide addition [153]. These errors are commonly referred to as carry forward and incomplete extension (CAFIE). The cumulative ef-

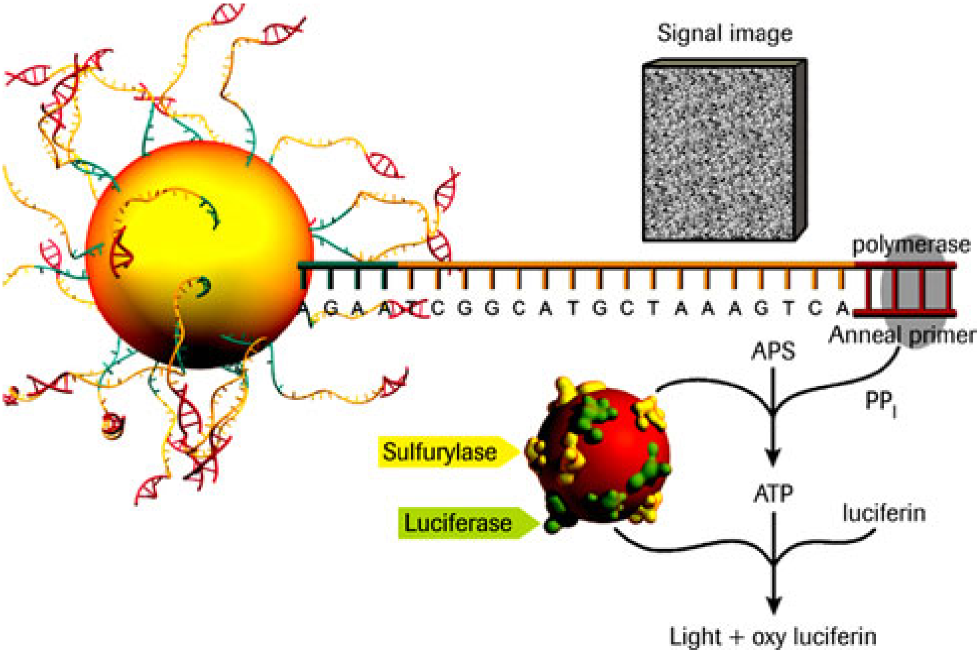

Figure 3.1: Pyrosequencing Chemistry, taken from the 454 website [13].

fect of CAFIE errors leads to the fact that quality decreases towards the end of a read [10].

3.2 The 454 Sequencing Process

This section outlines the complete 454 sequencing process from library preparation to data output. All steps contain potential error sources. In Chapter 4, error types and their putative sources in the process are explained in greater detail.

3.2.1 Library Preparation

The initial step in the 454 pyrosequencing process is the choice of sample input (from a subject or the environment) for library preparation. Sample input can be the DNA of a whole genome or targeted gene fragments of interest, but also PCR products (amplicons), bacterial artificial chromosomes (BACs) and cDNA. In a mechanical shearing process, the double-helix DNA ladder is broken into shorter double-stranded fragments of several hundred base pairs [163]. Samples consisting of smaller nucleotide molecules (e.g. small non-coding RNA) do not require fragmentation.

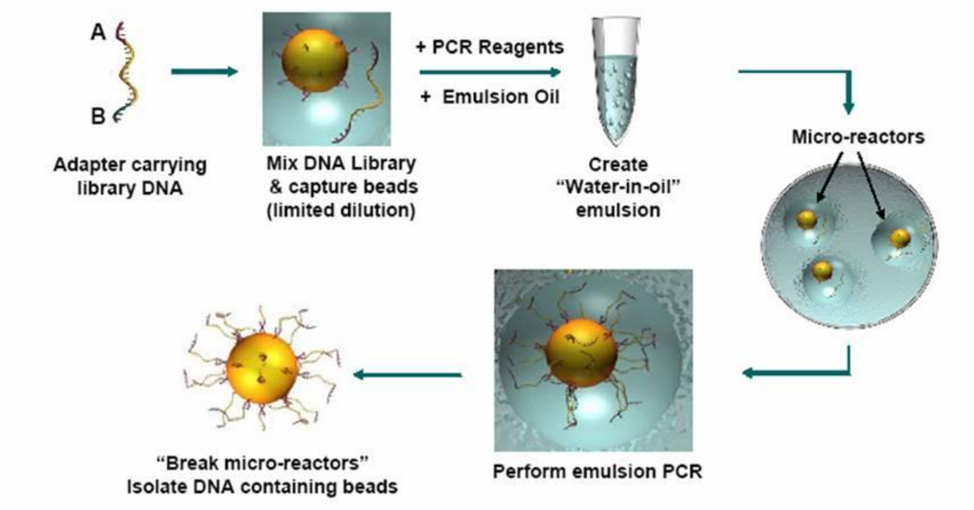
A
B
Adapter carrying library DNA
Mix DNA Library & capture beads (limited dilution)
+ PCR Reagents
+ Emulsion Oil
Create "Water-in-oil" emulsion
Micro-reactors
Perform emulsion PCR
"Break micro-reactors" Isolate DNA containing beads

Figure 3.2: Emulsion-based clonal amplification of the library, taken from the 454 website [13].

For purification, quantitation, amplification and sequencing, it is necessary to ligate shorts adapters (A and B) to the fragments. These contain universal priming sites, which allow the templates to be amplified with common PCR primers [10]. Finally, the fragments are separated into single strands (sstDNA), and one strand is discarded. The resulting templates represent the sequencing library.

All library preparation steps potentially introduce bias. Researchers inside and outside of Roche Diagnostics have since published attempts to further improve and simplify library preparation, e.g. by reducing the required amount of initial sample material, automating the library construction process, eliminating the titration step etc. [109, 164–168].

3.2.2 Emulsion PCR Amplification

Next, the DNA fragments are to be bound to beads under conditions that favor one fragment per bead. This process involves the following steps (see Figure 3.2): A water-in-oil-emulsion is created, containing the DNA library fragments along with capture beads and enzyme reagents, including polymerase and the firefly enzyme luciferase. This mixture is shaken so that droplets form around the beads, each bead being captured within its own so-called microreactor [169–172]. Typically, each droplet will only contain at most one sstDNA fragment. Now, the enzyme in the mixture causes the sstDNA fragment within the droplet to be amplified into around ten millions of copies by PCR. Since the reaction takes place in an emulsion, it is referred to as emPCR. Then, the beads are screened from the oil. Those that do not contain DNA are discarded; those that contain more than one DNA fragment are filtered out in a later step (see Section 3.2.5).

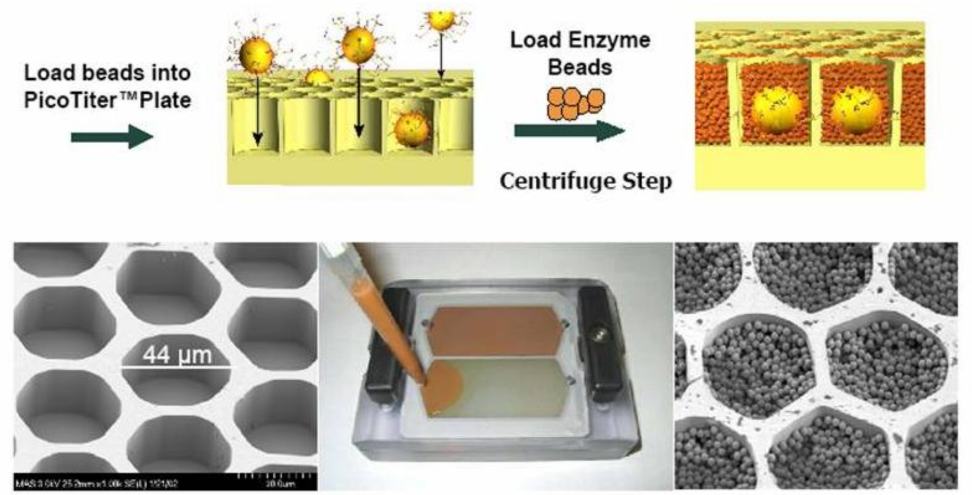

Figure 3.3: Depositing DNA beads into the PicoTiter plate, taken from the 454 website [13].

3.2.3 Sequencing

Now, the beads with the amplified sstDNA fragments are placed on a PicoTiter plate (see Figure 3.3), a device of 70 mm x 75 mm size containing 1.6 million hexagonal wells[18] [10, 173]. The beads are sized to ensure that only one bead fits into a well.[19]. Each well can be identified by an XY-coordinate on the plate that can then be placed in the 454 Genome Sequencer instrument for sequencing.

Amplification via emPCR (see Section 3.2.2) implies a population of identical templates in every well, and each template copy in a well undergoes the sequencing reaction independently. All beads carrying millions of copies of sstDNA templates are thus sequenced in parallel.

In the actual sequencing step, nucleotides are flowed sequentially in a fixed order across the plate.[20] If the flowed nucleotide is complementary to the nucleotide on the sstDNA template in a well, the polymerase extends the existing DNA strand by adding nucleotide(s). This addition results in a reaction that generates a chemiluminescent signal (see Figure 3.1), being recorded by a charge-coupled device (CCD) camera [10, 13, 38]. After the flow of each nucleotide reagent, the plate is washed with apyrase which ensures that no unattached nucleotides remain in the wells before the next nucleotide is flowed over the plate. This reduces the possibility of synchronism loss (see Section 4.2.7) [10].

[18] at a diameter of 44 μm and a volume of 75 picoliters each

[19] Still, a low percentage of wells contain more than one bead. Filtering mechanisms (see Section 3.2.5) take care of this problem.

[20] For GS 20, FLX and Titanium, the order equals to ATGC such that the signal translates by TACG.

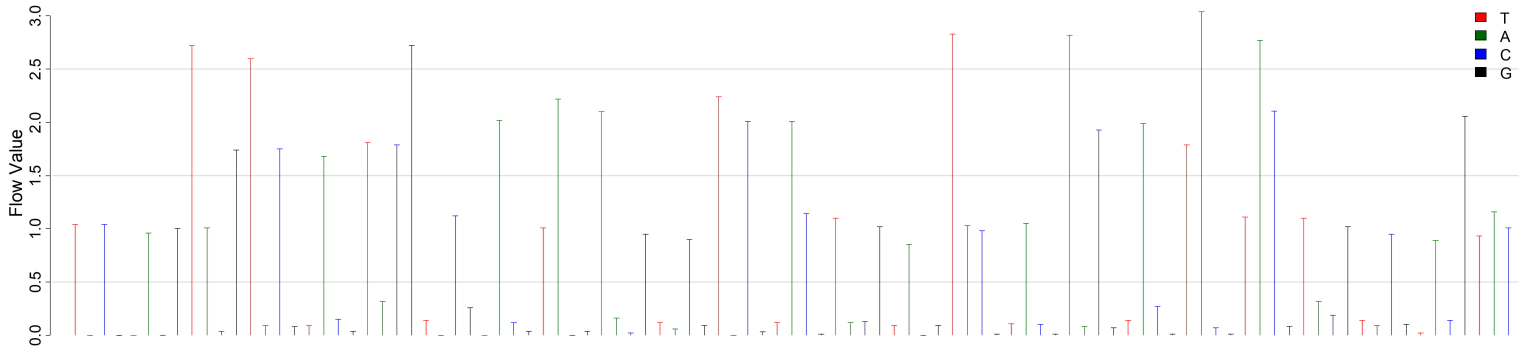

Figure 3.4: Flowgram for one read. The flowed nucleotides are displayed on the x axis. The bars on the y axis denote the flow values – i.e. the corrected light signals from the enzymatic reaction during the flowing process. Each pair of nucleotide and corresponding flow value indicates how many times the nucleotide is incorporated – or if it is not incorporated at all in case of a negative flow value. The DNA sequence can then be read from the left to the right. The flowgram was quality-trimmed after 99 flows, resulting in a 73 bp read. The first four positive flow values correspond to the key that is used to identify wells containing DNA-carrying beads and does not form part of the actual sequence [10].

3.2.4 Image and Signal Processing

– From Raw Data to Flow Data –

Technically speaking, raw data in the context of 454 sequencing are the imaging data. Light signal intensity is collected over the entire duration of a flow and proportional to the number of nucleotides incorporated, i.e. three consecutive As in the template would evoke a light signal at approximately three times the strength of a single A [10]. The observed signals of all template copies in a well are combined to obtain a consensus, raising a need for a highly efficient nucleotide addition process [18].

In order to determine the correct number of incorporated nucleotides for each flow and well, it is crucial to run correction algorithms on these data [153]. They are background-subtracted, normalized and corrected for well cross-talk (see Section 4.2.6) and other artifacts such as CAFIE errors (see Section 4.2.7). Based on the corrected light signal values – also referred to as flow values – the software creates a bar graph called a flowgram for each well on the plate (see Figure 3.4) [10]. Each flow value, expressed in a non-negative two-decimal float number, is proportional to the homopolymer length of the corresponding nucleotide. This corresponds to the incorporation of one, two or more nucleotides of the same kind (*positive* flow value) or no nucleotide incorporation (*negative* flow value). The term *negative* is somewhat misleading since those flow values that do not lead to a base-call are also (low-)positive (see Figure 3.4).[21]

[21] In some literature, negative flow values are therefore referred to as *noise* flow values. In most papers as well as in this thesis, the term *noise* is used to describe unwanted variations in quality. In Chapter 4, such quality variations are discussed in greater detail.

In principle, flow values directly indicate the number of incorporated nucleotides. They follow a series of statistical distributions (one distribution per homopolymer length, and an additional distribution for negative flow values). Optimally, each of these distributions would be a one-peak and one-value distribution on the integer value. More realistically, each distribution should be symmetrical, peak on the integer and have a small variance such that all values would lie within the interval of ±0.5 from the integer. In other words, if flow value distributions did not overlap, this would allow for an unambiguous translation of a sequence of flow values into a nucleotide sequence [10].

However, the reality is far from ideal. One can visualize the flow value distributions of a whole run by plotting flow values as a histogram (see Figure 3.5 left). Assigning each flow value to its true homopolymer length reveals a series of overlapping distributions (see Figure 3.5 right). The variance of the distributions increases with homopolymer length and also towards the end of a read.[22] The latter is due to CAFIE effects (see Section 4.2.7), revealing that the correction algorithms mentioned above only allow for partly removal of this error type.

Overlapping flow value distributions result in insertions (calling one or more additional bases than actually present in the genome) and deletions (omitting one or more bases relative to the underlying biological sequence) during base-calling. Insertions and deletions are collectively referred to as *indels*. Perceived substitution errors (miscalls, i.e. a wrong base is called) are significantly rarer.[25] During sequencing, they occur where an overcall follows an undercall or vice-versa.

All analyses and tools developed in the course of this PhD project have one thing in common: They build on flow data rather than on nucleotide sequences. Flow data contain more information than nucleotide data, which suggests that processing data

[22] This type of degradation is described in detail in the first paper, “Characteristics of 454 pyrosequencing data–enabling realistic simulation with flowsim”.

[25] In literature, the term *miscall* is sometimes used as a synonym of base-calling error, i.e. including indels.

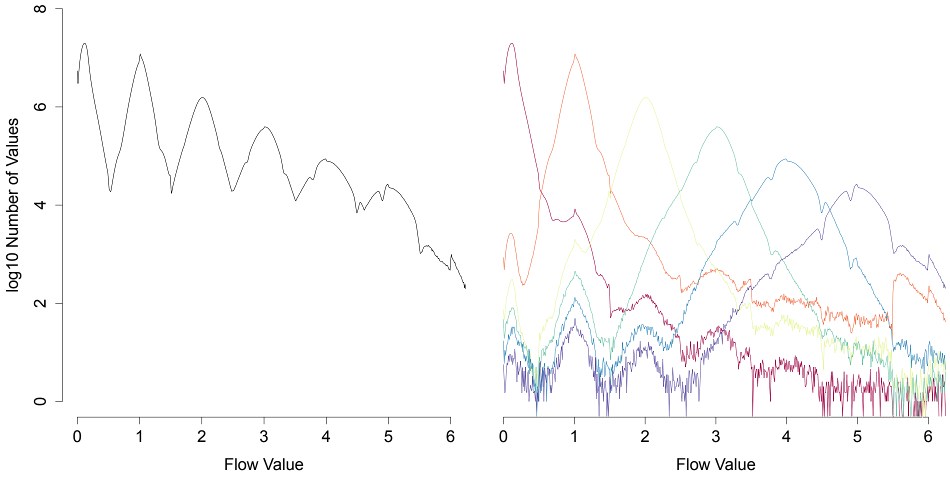

Figure 3.5: Histogram for flow values of a Titanium run (after quality-trimming, see Section 3.2.5). The y axis is plotted on a log scale in order to emphasize on the effect of overlapping distributions. **Left:** Overall histogram. **Right:** Histogram per homopolymer length, revealing the underlying flow value distributions.[24] Weaker, neighboring peaks in distributions can point towards other error sources than sequencing errors and are extensively discussed in Sections 4.2 and 5.2.

in *flow space* can generate more accurate results than taking into account nucleotide sequences only. It is therefore crucial to fully grasp the source and intrinsic characteristics of 454 data on their different aggregation levels (imaging data – flow data – nucleotide data).

3.2.5 Whole-Read Filtering and Quality-Trimming

Flow values are output in standard flowgram format (SFF) file format. Since this is a binary file format, there are a number of tools for text file output and data processing that either come with the sequencer or have been published (see Section 3.3). One SFF file usually corresponds to half a run/plate. Each read is characterized by a unique identifier (*read name*) and its X and Y position on the plate. Apart from the flowgram for each read, trimming information (a left and right trimpoint referring to the base-called nucleotide sequence) is also provided, indicating parts of each read that are either low-quality (commonly on the right end) or part of synthetic sequences such as adapters (see Section 3.2.1).[26]

Whole-read filtering and trimming are performed in flow space prior to base-calling. All filtering and trimming algorithms are described in detail in the 454 manual [174]. They aim to identify high-quality reads or sections of reads that can further be used in downstream data analysis. It is not uncommon that a high percentage (>50%) of template-carrying wells do not produce usable reads [143].

Obviously, flow values that lie close to integral values give more reliable estimates for homopolymer lengths than those that lie close to the valleys between the distributions (see Figure 3.5). This knowledge is used both in trimming algorithms and quality score calculation (see Section 3.2.7). In particular, reads containing a high percentage of flow values in the overlap region between negative and positive flows, roughly between 0.5 and 0.7, are often low-quality and can be used to identify such wells that accidentally carry more than one template. In contrast, high-quality reads have most of their signals close to the integral values equal to the number of incorporated nucleotides [10]. A low percentage of reads often accounts for a high percentage of errors within a run with a vast majority of reads being error-free [143, 175].

In a first step, the 454 software runs a series of whole-read filters on the sequencing data, in which failing any of the tests results in the rejection of the entire read. First, the *Keypass Filter* identifies wells that contain sequences with a valid key sequence.[27] The *Dots Filter* then rejects reads that are under 84 bp in length.

[26] The left trimpoint usually equals to 5, unless when tags for pooling of multiple samples are used (see Section 3.4). This corresponds to the first four nucleotides of the sequence being cut away and refers to the control key. Reads without this control key will not be contained in the final set of reads. The right trimpoint varies tremendously.

[27] The key sequence is a known four-nucleotide tag at the beginning of each read/flowgram, used to identify wells that contain template-carrying beads. It equals to TCAG for the GS 20, FLX and Titanium platforms and to GACT for Junior and FLX+.

Furthermore, all reads that contain a certain percentage of ambiguous flow cycles – reflected by three consecutive negative flows – are filtered out. This often happens when the signal intensity in a well is generally low. Lastly, the *Mixed Filter* aims to identify multi-template beads by calculating the percentage of positive, borderline positive and negative flows and a number of other metrics.

In a second step, all reads that have passed the three whole-read filters are run through a series of trimming algorithms. Trimming is performed from the right end of the read[28] and assesses the quality of flow values instead of single bases. This means that all bases of a homopolymer run are either included in- or excluded from the trimmed read, the trimpoint cannot lie between those bases.

The *Signal Intensity Filter* determines such reads that have a certain percentage of flows in the overlap region between 0.5 and 0.7 and iteratively trims a read until this percentage drops below a pre-defined threshold. The *Primer Filter* screens all processed reads for similarity to adapter sequences (see Sections 3.2.1 and 4.2) and trims all flows that are supposed to represent or partly represent the adapter. The *TrimBack Valley Filter* identifies the valleys between the flow value distributions, defines and calculates a percentage of low-quality flows and trims the read according to a set threshold. All reads that are no longer than 84 bp after trimming are discarded.

Certain parameters of the whole-read filtering and trimming algorithms can be changed in order to adjust stringency. Increasing stringency will lead to a higher average accuracy but also to a lower yield of reads from a run.

After filtering and trimming, quality scores are calculated based on flow values and assigned to each base after base-calling (see Sections 3.2.6 and 3.2.7). An additional trimming step, the *Quality Score Trimming Filter*, is run on nucleotide sequences after quality scores have been computed. The remaining sequences are

[28] also referred to as 3” or distal end, i.e. the end opposite the sequencing primer, represented by the later nucleotide flows of a run

considered high quality.

3.2.6 Base-Calling

– From Flow Data to Nucleotide Data –

Base-calling is the procedure of identifying DNA bases from the sequencer's output [176]. One flowgram (see Figure 3.4) corresponds to one read and contains a certain number of flow cycles depending on the platform's generation (see Section 3.4). One flow cycle encompasses four flows in fixed order. The nucleotide sequence is then derived from the pairs of the flowed nucleotides and the corresponding flow values. This procedure requires thresholds for determining whether a base was incorporated or not, and if yes, for calculating its homopolymer length. When the system fails to identify any base throughout an entire flow cycle (i.e. outputs at least three negative flows in a row), an N (undetermined/ambiguous base) is called.

From literature, one is sometimes led to believe that flow values are simply rounded to the closest integer in order to obtain the homopolymer length. In fact, this approach would be valid if the normalization and correction algorithms run on the imaging data (see Section 3.2.4) worked perfectly[29]. However, this is not the case. Instead, thresholds are determined by calculating the valleys that separate homopolymer distributions (see Figure 3.5 left). These can vary from run to run, emphasizing the extent of thresholding in base-calling as a putative error source.

Parts of the sequence that lie beyond the left and right trimpoints (see Section 3.2.5) are also base-called but, by convention, written to output in lower-case letters. This makes it possible to distinguish between high- and low-quality regions of a read in downstream analyses.

[29] Roger Winer, Roche Diagnostics, pers. comm., August 31st 2010

3.2.7 Quality Score Calculation

Quality scores as a measure of per-base confidence compress a variety of types of information into a single probability-of error value [177]. A number of analysis and data-cleaning tools [178, 179] and a large number of assemblers use quality scores in order to deliver accurate results. This also expresses a need for a score that is comparable across sequencing platforms, especially when comparing sequencing results from different technologies or laboratories or when carrying out hybrid assemblies from Sanger and NGS data (see Section 2.4.2).

In Sanger sequencing, the quality score is an estimate of the called base being erroneous. Sanger quality scores are also called *phred* scores, referring to the program that introduced their calculation [180, 181].

A first quality score algorithm for 454 sequencing was published with the GS 20 platform [10]. Making use of Bayesian statistics, the quality score for an individual base was determined by the probability that the measured flow value originates from a homopolymer of length at least equal to the called length (i.e. that the base in question is an overcall) [10]. The probability was then transformed into a phred-equivalent (see Formula 3.1). The lower the probability of an overcall, the higher the quality score.

$$Q_{\text{GS 20}} = -10 \cdot log_{10}(\text{probability of overcall}) \tag{3.1}$$

For calculating this probability, parametric distributions were fitted to the flow values. Negative flow values were supposed to be log-normally distributed, and positive flow values were fitted a Normal distribution, with mean and standard deviation proportional to the underlying homopolymer length.

However, quality scores calculated according to this algorithm were found to underestimate actual base accuracy [182]. They were especially criticized to only

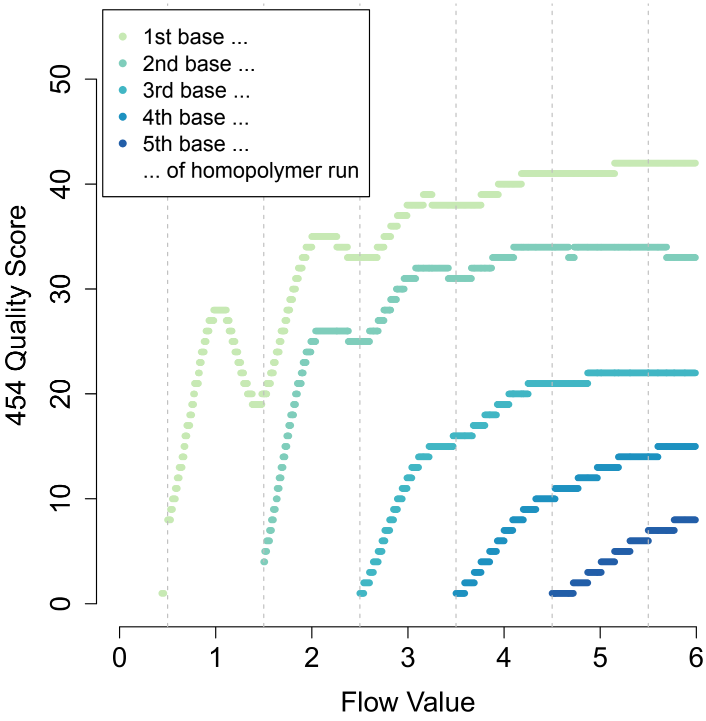

Figure 3.6: Quality Scores for a GS 20 run. The first base within a homopolymer run is assigned the highest quality score.

reflect the probability of an overcall but not the probability of undercall or miscall errors [143, 177]. Instead, the first base in a homopolymer run was always assigned the highest quality, and the last base the lowest (see Figure 3.6) – for both correctly and incorrectly called bases. A by-effect is that – regardless of error – the average quality score of those reads containing many and long homopolymer runs is lower than that of other reads.

Consequently, a new approach on defining quality scores was introduced after the release of the FLX platform, developed in cooperation with the Broad Institute. The new scores were designed to treat overcalls, undercalls, and miscalls evenhandedly. Thus, the new scores reflect the true error rate more accurately and identify a larger number of high-quality bases compared to the GS 20 algorithm (see Formula 3.2) [177]. The accurate prediction of undercalls is crucial since they comprise a high percentage of errors. Quality assessment of miscalls is especially important in the context of SNP discovery.

$$Q_{new} = -10 * log_{10}(\text{accuracy}) \tag{3.2}$$

While the old algorithm only used the flow value of the base in question, the new strategy compares the properties of each flow value against properties that have been found to correlate with high or low quality – involving all flow values of a read [174]. These properties are captured in six noise predictors that serve as input to the quality score algorithm, ranked by importance from high to low:

1. Observed noise in the neighborhood of the corresponding flow – providing an estimate of homopolymer accuracy
2. Observed noise in the whole read – measured as overall "separation" of the flowgram distributions
3. The corresponding flow value
4. Homopolymer length corresponding to the called base – higher homopolymer lengths yield more errors
5. Homopolymer length of the same base in the previous flow cycle – giving an indication of CAFIE effects
6. Base position in the read – later flows yield more errors

The GS 20 algorithm was replaced after the publication of the new algorithm. Each platform now uses its own lookup table for quality scores, generated from training data sets in order to account for the different error characteristics of the chemistries [174].[30]

Both the base-called sequence and the associated quality scores are reported in the SFF file (see Section 3.2.5) of a run.

3.3 Information Extraction Tools

Whenever it is desired to work in flow space instead of nucleotide space – whether for visual inspection of flow data or for using tools and pipelines in various application areas that build on flow data – one has to extract the information from the SFF file. Sfffile [13] is a command line tool that constructs a single SFF file from a list of SFF files, and reads can be filtered using inclusion and exclusion lists of read names (identifiers). This is useful when pooling results from multiple runs or regions to simplify further handling of the data. Sffinfo [13] extracts the whole or specified information from SFF files and writes to standard output in text form. For example, sffinfo can be used for generating the FASTA and associated quality score files (FASTA [183] and FASTQ [184] format) of the reads.[31] A majority of bioinformatic tools accept FASTA format (i.e. work in nucleotide space), many of them for historical reasons since they originally date from the Sanger era.

However, sffinfo and sfffile are not publically available since they are distributed with the 454 sequencing platform. As such they cannot be modified or redistributed. For this purpose, Flower [186] was written – a command line tool that provides textual output similar to sffinfo and writes to different output formats such as FASTA and FASTQ but can also generate easy-to-read tabular output and histogram data of

[30] Although Roche Diagnostics claim to use the algorithm described in Brockman *et al.* [177], the predictors described in the 454 manual differ slightly from those enlisted in the paper. The predictors mentioned here are taken from the paper.

[31] Lysholm *et al.* suggest the FFASTA (Flowgram-FASTA) format as an alternative to SFF, following a FASTA-like structure, but containing flowgrams instead of nucleotide sequences[185].

flow values. The latter is very useful for visualizing flow value distributions (see Figure 3.5) within a run.

Similarly to Flower but with less functions, sff_extract [187] is a simple command line application written in Python, targeted to extracting information from SFF files.

3.4 Read Lengths and Throughput

Throughout the years, 454 has made great refinements to both the sequencing chemistry and correction algorithms [13, 18, 38]. With the release of the Titanium technology in 2009, the plate was improved with a titanium-coated PTP design, reducing well cross-talk (see Section 4.2.6) to a minimum [18]. All those improvements have led to higher throughput with higher overall quality. Most notably, there has been a decrease in the per-base error rate [10, 13, 38, 143, 188–190] (see Section 4.1) and an increase in read length (see Figure 3.7). For their latest platform FLX+, Roche Diagnostics report read lengths of up to 1,000 bp with a mode value of 700 bp from a typical sequencing performance [13]. As a comparison, Sanger can yield read lengths of up to 1,000 bp at an average of 800 bp [18, 37].

Read lengths vary from run to run and depend on the generation-specific number of flow cycles (GS 20: 42 cycles, FLX: 100, Titanium: 200, FLX+: 400), but also on clone length, data quality and sequence complexity. Roughly, the average number of nucleotide bases gained within one flow cycle can be estimated to be 2.5 [15]. Furthermore, genome content that is more AT- or GC-rich typically yields longer reads as compared to AT-/GC-neutral genomes [38].[32] Read lengths reported in literature commonly refer to quality-trimmed sequences (see Section 3.2.5).

Long reads from the latest platforms Titanium and FLX+ are especially tailored

[32] GC-content is defined as the percentage of the bases cytosine and guanine in all bases of a sequence/genome.

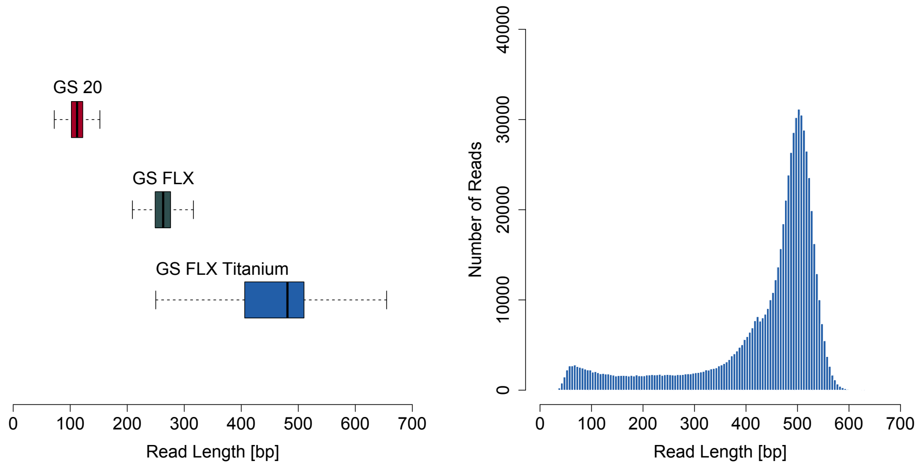

Figure 3.7: **Left:** Boxplots for read length in three runs from different platform generations. **Right:** Read length distribution for a Titanium run.

to improving *de novo* assemblies, yielding fewer gaps, longer contigs and scaffolds, and to overcoming issues when assembling repetitive regions. This makes the technology particularly useful for assembling complex genomes, but also for hybrid assemblies using FLX+ shotgun reads, paired end reads with different insert sizes (see Section 2.2), and short-read data (e.g. from Illumina or SOLiD). Such study designs reduce project costs and eliminate the need for additional Sanger sequencing. Furthermore, metagenomic studies (see Section 2.5) also benefit from longer reads due to an improved sensitivity and specificity of taxonomic assignments [138], i.e. longer reads lead to a higher probability of correctly identifying population members and hamper wrong classifications [19]. Whiteford *et al.* [191] analyze the level of genome sequencing possible as a function of read length.

The number of reads per run has greatly increased since the release of the first 454 sequencing platform in 2005. While a GS 20 run produces around 250,000 usable reads, FLX produces 350,000-400,000. Titanium and FLX+ yield around 1 million reads. However, the purchase cost and infrastructure still limit the use of 454 sequencing. GS Junior, released in 2010, is 454's answer to this need: a benchtop solution ("no bigger than a typical laser printer") that is particularly fitted to the needs of small- or medium-sized laboratories, producing around 100,000 reads. Since the GS Junior Titanium chemistry uses 200 flow cycles as introduced with Titanium, it reaches comparable read lengths (500 bp on average) [192].

In order to fully exploit the high throughput of 454 sequencing, a plate can be split up into several projects. This allows for efficient pooling of multiple samples that require less sequence data (such as BACs or amplicons), but also for application development and feasibility testing. One way to achieve such a partitioning is to physically divide the plate into smaller regions by the use of gaskets [13]. Alternative solutions are molecular barcoding techniques that rely on attaching sample-specific adapters to DNA samples, such as parallel tagged sequencing (PTS) [167, 168] or multiplex identifiers (MIDs) [13, 38]. Using the tag sequences, the source of each DNA sequence can be traced.

4

454 Sequencing – Characteristics and Artifacts

Due to the inherently unpredictable nature of biological data, there is always some distance between the theoretical design of a bioinformatic solution and the successful implementation of the solution in a working program that can handle real-world data reliably. The only way to shorten such distance to perfection (...) is to form a close collaboration between computer scientists and biologists. This allows the wisdom and experience of biologists to be slowly translated into functional program code.

Chou and Holmes: DNA sequence quality trimming and vector removal [178]

When deciding on which technology or platform to use for a certain sequencing project, researchers usually take into account at least three factors: Costs, sequence statistics (e.g. read lengths, see Sections 1.2 and 3.4) and error rates. Notably, there is a variety of approaches and tools for enhancing data quality from sequencing platforms. In which fashion these should be used depends on a thorough understanding of the underlying data. Just as every other sequencing technology, 454 has its intrinsic characteristics, error patterns and artifacts.

This chapter is divided into three sections. The first section reports per-base and consensus accuracy statistics for 454 sequencing. The second section aims to give a detailed overview of the most important error patterns, artifacts and issues that are relevant to downstream analysis, in the order they may occur during a sequencing project. The third section discusses strategies and tools for data cleaning in 454 sequencing.

4.1 Accuracy

The per-base error rate is commonly defined as the number of errors (insertions, deletions, substitutions) divided by the number of bases in a data set[33], e.g. over a whole run or plate [143]. Error calculations are based on aligning reads to their reference sequence.[34]

Most applications require a low per-base error rate, while others (such as *de novo* WGS, see Section 2.4) allow for mitigating errors through redundancy in sequencing (i.e. through increasing coverage, see Section 2.3). By aligning/overlapping reads to each other, one can then generate consensus sequences. These typically have a greater accuracy than single reads, referred to as consensus accuracy. Both per-base and consensus accuracy can be further improved by additional quality-filtering and -trimming.

On the one hand, the pyrosequencing technology provides such a large number of reads that the data loss through eliminating erroneous sequences is usually by far offset by an overall quality increase [143]. On the other hand, consensus-based projects would also benefit from a lower error rate since this would decrease the required coverage for building a reliable consensus (see Section 2.3).

It is common practice among researchers to report per-base error rates (the lower the better) and consensus accuracy (the higher the better). In an analogous manner,

[33] usually referring to sequences that have undergone default quality-filtering and -trimming

[34] Commonly, when calculating error rates, researchers assume that the reference they have obtained from a database or from another source is correct. However, this assumption is not always true [58].

it is possible to calculate per-base accuracy or a consensus error rate.

4.1.1 Per-Base Error Rates

Traditional Sanger sequencing has a per-base error rate of of 0.4-0.7% [38]. In contrast, when the first 454 platform GS 20 was launched in 2005, a per-base error rate of 4% was reported [10]. Overall sequence quality has increased over the years from GS 20 to FLX and FLX Titanium [38, 193].[35] However, sequence quality varies tremendously from lab to lab, project to project, and run to run [175]. Reported average per-base error rates vary from approximately 0.39% to 0.5% for GS 20 [143, 175, 182], 0.12% to 0.4% for FLX [175, 182], and 0.12% to 1.07% for Titanium [189, 190].[36] It is thus fair to say that 454 sequencing has reached the accuracy of traditional Sanger sequencing.

It is a well-known characteristic of 454 sequencing that errors are mostly indels, accounting for a high percentage of errors at around or over 90% [182, 195]. Substitution errors occur at a substantially lower rate [38, 182]. Many research groups have reported insertions to be the most common error type, followed by deletions [10, 18, 143, 175, 177, 182, 189]. Furthermore, it has been observed that indels most frequently occur in homopolymeric regions [10, 189, 196–199].[37] One main reason for this phenomenon can be found in the broadening of the flow value distributions (see Section 3.2.4).

Nucleotide-dependent effects [143, 182] have also been observed. Transitions

[35] To date, no per-base accuracy has been reported for FLX+, and researchers complain about short reads and low throughput compared to the numbers promised by Roche Diagnostics [194].

[36] Enhancements to both the chemistry protocol and to the built-in software (e.g. correction and base-calling algorithms), also disconnected from new platform releases, putatively account for the change in error rate when compared to the numbers published by Roche Diagnostics.

[37] Some researchers report error rates separately for homopolymeric and non-homopolymeric regions. For example, Droege and Hill [38] report a per-base error rate of <0.5% over the first 200 bp of a FLX read, where a majority of errors occur in homopolymer stretches. Excluding these, the error rate is lowered to <0.1%. Margulies *et al.* [10] and De Schrijver *et al.* [200] report error rates as a function of homopolymer length.

between nucleotides – both in indels and in substitutions – have been found to be biased towards certain nucleotide combinations [143, 162, 197, 199]. The pattern of substitutions was found to be similar to that observed in studies on PCR fidelity (see Section 4.2.2), suggesting that polymerase errors are the cause of most of the observed substitutions in amplicon sequencing [197]. PCR errors are further discussed in Section 4.2.4.

4.1.2 Consensus Accuracy

Errors in homopolymer stretches can often be detected and corrected by building a consensus sequence from several reads. This strategy is based on the knowledge that errors are not randomly distributed across all reads [10, 143, 201]. A vast majority of reads are completely or almost error-free, while those reads that contain errors contribute disproportionately to the overall error rate.[38]

In particular, genome assembly (see Section 2.4) strongly benefits from consensus building. Margulies *et al.* [10] report a consensus accuracy of 99.94% for GS 20.[39] Nevertheless, repeat identification in genome assembly requires a certain level of accuracy such that almost-identical repeats can be correctly assigned to their respective positions in the genome.

Enhancing accuracy by building consensus sequences is not possible in studies that seek information about natural variation from each read [143] (e.g. in microbial diversity studies). In such projects, it is crucial to, as far as possible, identify and correct or remove errors and artifacts (see Sections 2.5 and 4.3.8).

[38] In strong contrast to these findings, Gilles *et al.* [189] report almost 90% erroneous sequences at a relatively high overall error rate of around 1%.

[39] Similarly, Moore *et al.* [202] achieve a consensus accuracy of 99.96% and 99.97% for two plastid genomes at approximately half the coverage, resulting from improvements in the assembly software. Using an older version of the software resulted in much lower consensus accuracy for both genomes, 99.93% and 99.86%.

4.2 Known Error Patterns

Errors can arise at different stages during the library generation and sequencing process (see Figure 4.1), and there can be several sources for an error pattern. As suggested in the 454 sequencing protocol, the most relevant source of error may vary from experiment to experiment. A detailed empirical analysis of 454 error patterns has been carried out by Huse *et al.* [143] for the GS 20 platform. Gilles *et al.* [189] have published a follow-up study on Titanium data.

Library preparation (see Section 3.2.1) sometimes requires PCR amplification. One situation where this is almost always true is in the generation of microbial diversity sequencing libraries. This means that the limitations and biases introduced by PCR amplification have to be taken into account when interpreting results in downstream analysis. Furthermore, the 454 sequencing procedure involves an emPCR amplification step so that the enzymatic reactions produce sufficient signal for detection by the camera device (see section 3.2.2). In brief, every 454 sequencing project involves at least one PCR amplification step, namely emPCR. Sequencing of amplicons involves an additional PCR step (multi-template PCR first, emPCR later). In the rest of the thesis, these procedures are referred to as PCR and emPCR.

Using PCR when amplifying regions of interest can both lead to PCR bias and cause miscalls due to polymerase errors. In addition, chimeric sequences can be generated during the PCR process [203]. In contrast, sequencing libraries that do not involve any amplification for library preparation should be free of chimeras and PCR bias. In the emPCR strategy employed in 454 sequencing (see Section 3.2.2), each template is entrapped in its own microreactor. This implies that there is no competition between multiple templates for a limited number of PCR reactions, leading to bias-free amplification [14]. However, the emPCR step is where artificial duplicates arise, and these can account for a large percentage of sequences (see

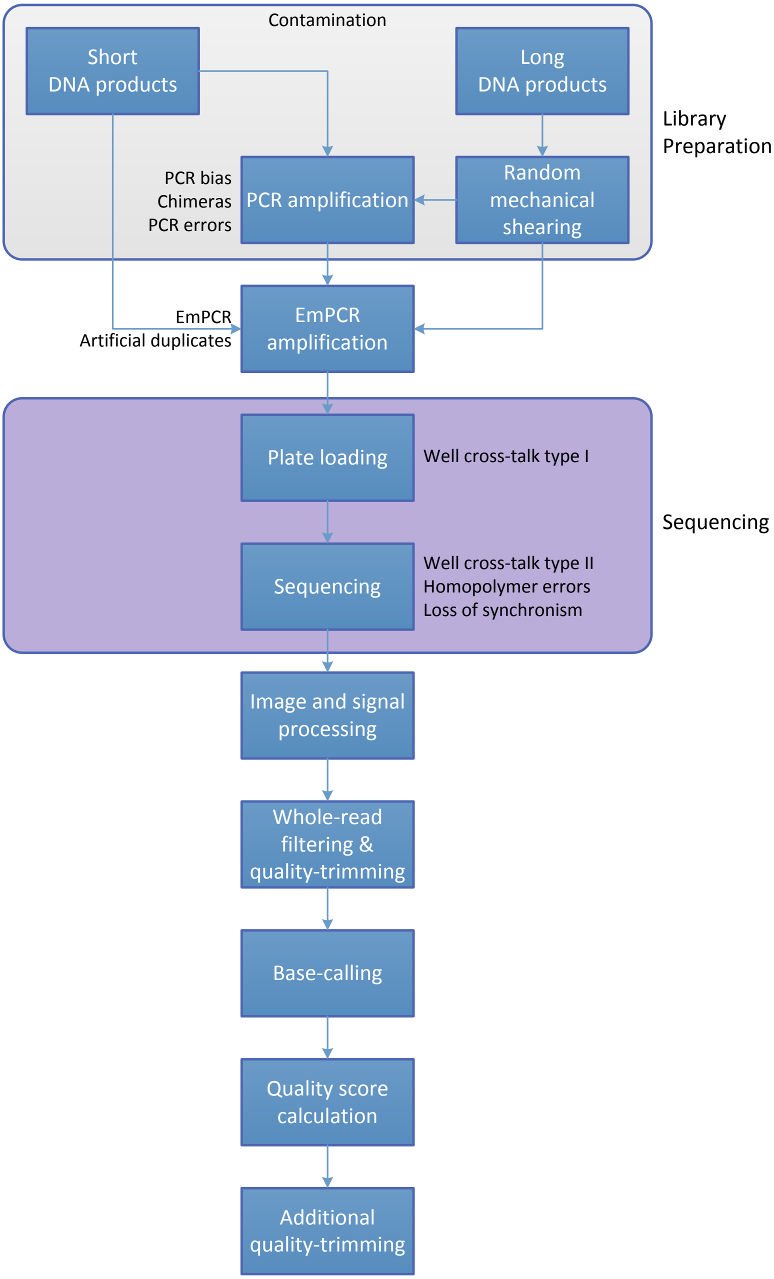

Figure 4.1: Error sources in the 454 sequencing process.

Section 4.2.5).[40] In other words, 454 sequencing of DNA products that have *not* been amplified for library preparation can be expected to deliver unbiased coverage *if* filtered for artificial duplicates prior to downstream analysis.[41]

The following sections explain known artifacts in their order of appearance according to Figure 4.1.

4.2.1 Contamination

Contaminants in a sequencing library can come from many sources, amongst others *E. coli*, cell plasmids, organelles, viruses, yeast or human (the latter due to handling during the experimental process) [204]. In the Sanger era, when *E. coli* was commonly used as a cloning host, the most common contamination was induced by the cloning vectors themselves. Since 454 sequencing involves no cloning step, one should expect less contamination, but it cannot be completely excluded [91].

4.2.2 PCR Bias

In amplicon sequencing, the comparison of template and product ratios often reveals considerable and reproducible discrepancies in amplification of specific templates [203, 205–207]. This is referred to as PCR bias. Polz and Cavanaugh [207] explore potential causes and the extent of bias in PCR amplification, finding different primer binding energies to be the primary cause for overamplification. Bias can be considerably reduced by using high template concentrations and performing fewer PCR cycles, but this approach is often unrealistic given the small sample amounts.

[40] Ratan *et al.* [40] astonishingly found 454 sequencing, but not SOLiD sequencing to be immune to emPCR bias although the platforms share the same emPCR approach. Notably, the research group removed all artificial duplicates before calculating coverage.

[41] In contrast with this theory, Harismendy *et al.* [58] state that only a small part of coverage bias can be explained by amplicon-specific bias [58]. However, the study was published before the research community became aware of artificial duplicates as a biasing factor.

Sequencing coverage of amplified DNA products can vary tremendously across a genome since PCR is less effective for some genomic regions than for others [58, 207–209]. For example, sequence composition (e.g. GC-content) has been identified as a main factor causing such cloning biases.

4.2.3 Chimeras

Chimeras are sequences that are composed of two or more true sequences, with a discrete break point where the transition from one sequence to the other occurs [135, 162]. The percentage of chimeric sequences varies widely, from few up to 45% [210, 211]. One of the factors influencing chimera formation is high sequence similarity.[42] Unfortunately, this is exactly the situation in microbial diversity studies that use 16S gene sequencing (see Section 2.5) [188], and undetected chimeras may be misinterpreted as novel species [210]. This in turn causes inflated estimates of diversity, which is why the detection and removal of chimeras is crucial in such studies.

Chimeras occur due to experimental errors during PCR amplification. Beside sequence similarity, other factors that have been shown to favor chimera formation are the number of PCR cycles and relative abundance of gene-specific PCR templates [211]. The most common scenario involves annealing of an incompletely extended template, where a partially extended sequence from one template reanneals to another parent during the next PCR cycle [210, 212]. Several factors have been found to reduce chimera formation experimentally. Since artifact formation occurs at a higher rate during the last few cycles of the PCR reaction [212, 213], lowering the number of amplification cycles. Unfortunately, as with reducing PCR bias, this is rarely possible in practice.

[42] In contrast, highly diverse amplicon libraries that do not contain conserved regions will only produce few chimeric reads [201].

This situation creates a need for computational methods that identify chimeric sequences (see Section 4.3.3). Algorithms either target databases of chimera-free sequences or detect chimeras by exploiting abundances [162, 203, 210, 211].

4.2.4 PCR and emPCR Errors

Polymerases are never 100% accurate, and errors arising during PCR have been extensively discussed and modeled in the past [212, 214–223]. Similarly, the detrimental effect of PCR errors in 454 amplicon sequencing has been described [162, 177, 197, 224].

PCR errors can occur both during library preparation and during emPCR, but the latter will only have minor implications.[43] These two error types are referred to as PCR errors and emPCR errors below.

Whenever the observed percentage of miscalls is higher than the reported or expected substitution error rate for a certain platform (see Section 4.1.1), this suggests that PCR amplification is the main error source. PCR errors are not necessarily associated with homopolymer tracts and often occur at a low but rather even rate across amplicon reads [197]. Most notably, the corresponding flow values are often high-quality, i.e. close to integers.

In emPCR, each sstDNA fragment is amplified into around ten millions of copies on the same bead (see Section 3.2.2). This requires at least 24 PCR cycles, ideally yielding approximately 17 million (2^{24}) copies through the branching process at maximum efficiency. If the (em)PCR process fails during the very first PCR cycle, all templates will be affected. If the PCR process fails for one of the templates in the second PCR cycle, half of the templates will be affected, one fourth in the third PCR cycle, and so on. When the consensus flowgram is calculated (see

[43] Zagordi *et al.* [225] report a significantly higher substitution error rate for PCR-amplified samples when compared to non-amplified samples (0.25% vs. 0.05%).

Section 3.2.4), emPCR errors will most likely be leveled out by the large number of templates copies on a bead. In addition, unless they occur in the first PCR cycle, they would appear as low-quality bases: A certain percentage suggests the incorporation of a base at a certain flow, but the rest would not incorporate the base. Averaging these flowgrams would lead to flow values that lie somewhere in the low-quality region between integers. Conversely, miscalls where the corresponding flow values peak on the integers – suggesting that all molecules on the bead have the same substitution – most likely reveal PCR errors from the library preparation step.

EmPCR errors can, especially if they occur in an early PCR cycle, have implications that appear as CAFIE errors (see Section 4.2.7) and thus influence quality degradation towards the end of a read. PCR errors would not show this pattern.

4.2.5 Artificially Duplicated Reads

Gomez-Alvarez *et al.* [226] were the first to point out that 454 sequencing data can suffer from a high percentage of artificially duplicated reads, identifying between 11% and 35% of sequences to be exact or almost-exact duplicates of other sequences. High percentages have also been reported by other research groups [78, 227, 228], but it remains unclear which factors cause one run to suffer from a substantially higher duplicate rate than another run.

Potential biases introduced through the presence of artificial duplicates are especially harmful in quantitative analyses (such as microbial diversity studies) and transcriptome profiling, where the amount of reads is used as an abundance measure [175]. This kind of bias is also problematic in variant detection, where empirical or parametric distributions of substitution error rates can be used to distinguish sequence errors from true variants at various thresholds [196].

The source of artificial duplicates was first suspected to lie in signal bleeding

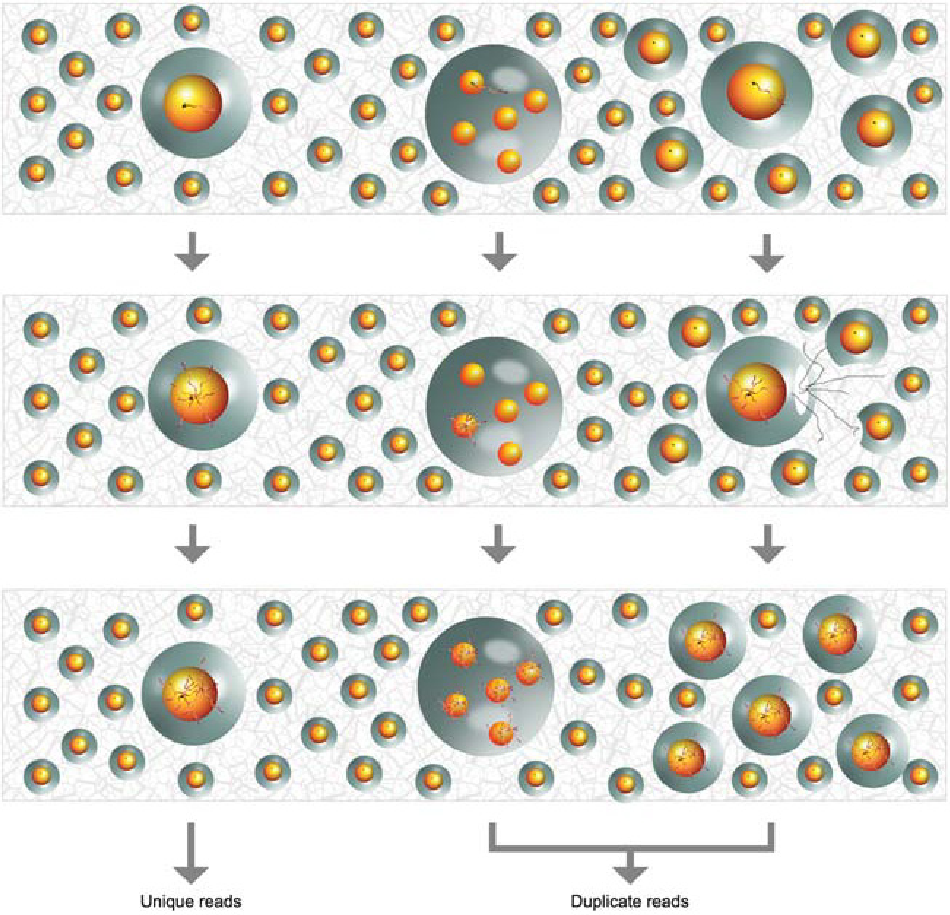

Figure 4.2: Hypothesis of duplicate reads generation during emulsion PCR, taken from Dong *et al.* [228]. **Left:** Amplification of sstDNA fragments during emulsion PCR (see Section 3.2.2), resulting in unique reads. Each droplet contains at most one bead. **Middle:** A sstDNA fragment and multiple beads may be contained in one emPCR droplet. After several cycles of PCR, DNA templates could bind to other beads in the same droplet and are further amplified during following cycles, generating artificially duplicated reads. **Right:** Some droplets may be broken during PCR and release multiple copies of DNA templates, be amplified on empty beads and generate artificial duplicate reads.

from neighboring wells [229] (see Section 4.2.6), but this cause can be neglected since the release of the Titanium platform [226, 230]. In our own analyses, we were unable to see any location effects when visualizing duplicate clusters in relation to read positions on a plate. Also, pre-amplification of DNA products for library preparation cannot be the source of artificial duplicates since high rates of artificial duplicates are not only detected in amplicon sequencing runs. Furthermore, Dong *et al.* [228] showed that several runs generated from the same sequencing library did not reveal overlaps between the members of duplicate clusters.

It is therefore highly probable that artificial duplicates arise during the emPCR step (see Figure 4.2). This is extensively discussed in our third paper.

One caveat when removing duplicates from a data set is that artificial duplicates cannot be distinguished from natural duplicates. Natural duplicates are reads from the same origin that start at the same genomic position by chance [175]. Simply removing all duplicates may therefore lead to an *under*estimation of abundances in quantitative analyses. Gomez-Alvarez *et al.* [226] and Niu *et al.* [175] provide formulas and tools for estimating the number of natural duplicates from genomic or metagenomic data sets. Intuitively, the number of natural duplicates highly correlates with sequencing coverage [175].

Another problem is that the removal of artificial duplicates is not possible for amplicons since it is impossible to discriminate between an amplicon fragment that was intentionally duplicated during PCR for library preparation or accidentally duplicated during emPCR.

Researchers have, throughout the last years, become aware of the fact that 454 data often contain a considerable number of duplicated reads that, depending on the application, can have a rather big influence on analysis results. However, it is to suspect that many projects that were finished before awareness about duplicates was raised suffer from a strongly biased interpretation [228].

4.2.6 Well Cross-Talk Type I and II

For GS 20 and FLX, it has been observed that the diffusion of ATP (see Section 3.1) sometimes induces a background signal in a neighboring well, a phenomenon also referred to as “ghost well”. Ghost wells are easy to identify computationally as they surround wells with identical signals but are characterized by low signal strength (a background signal of 10% or less) [10]. In order to avoid ghost wells, bead occupancy was limited to approximately 35% of all wells in GS 20 [10].

With the release of the Titanium technology, the plate was enhanced with a titanium-coated PTP design (see Section 3.4). The metallic coating using smaller DNA capture beads permits a higher density of wells, and makes improvements in both the number and length of reads possible [13].

A second kind of crosstalk between wells occurs due to “optical bleeding”, i.e. during image processing (see Section 3.2.4), due to the cladding of the camera not being completely opaque. By the use of an algorithm that was built on empirically determined data, the images are corrected for optical bleeding effects before being translated into flow data [10].

4.2.7 Homopolymer Errors and Loss of Synchronism

In pyrosequencing, a homopolymer is represented by a single flow value. This can lead to ambiguity of homopolymer length, especially in long homopolymers [143]. Although linearity in flow signals is preserved up to a homopolymer length of eight [10], the increase of signal intensity attenuates at higher homopolymer lengths [231]. This makes it harder to discriminate between flow value distributions, leading to indels and thus to higher error rates for longer homopolymers.[44]

As sketched in Section 3.1, one requirement for an accurate pyrosequencing

[44] De Beuf *et al.* [231] report an error probability of around 0.06 for homopolymers of length 4, and almost 0.1 for those of length 5.

system is that the parallelized flowing is and stays synchronous. However, one of the inherent problems of sequencing-by-synthesis is that, during strand extension, one or more strands get ahead or behind the other strands on a bead. This is referred to as CAFIE (carry-forward and incomplete extension).

Incomplete extension (also referred to as *lagging-strand dephasing*) occurs due to insufficient exposure of nucleotides to reagents, especially in homopolymeric regions.[45] Some DNA strands on a bead fail to incorporate during the flow, and must await another flow cycle for sequencing to continue, which means that they are incorporated out-of-phase with the other strands [174]. Incomplete extension can cause deletions and – assuming that multi-template beads have been successfully filtered out by the 454 software – is also the main source for undetermined bases (Ns).

Carry-forward (also referred to as leading-strand dephasing) is usually caused by leftover nucleotides in a well. This happens due to inefficient nucleotide degradation by the apyrase during the washing step [10, 18, 143, 153]. In particular, long homopolymer runs can partially transfer their strong signal to the subsequent flow cycle [177]. This will cause insertions.

CAFIE effects are – due to their cumulative effect – the main reason for quality degradation towards the end of a read [10]. This makes it essential to correct for CAFIE errors in order to obtain long reads with high quality.[46] The application of correction algorithms during the transformation from imaging to flow data (see Section 3.2.4) aims to reduce this type of error. However, some level of CAFIE noise remains, and several research groups have made attempts to quantify the proportion of CAFIE errors of the overall error rate [143, 197].

CAFIE effects can – apart from inaccuracies in the sequencing chemistry – also

[45] In theory, complete incorporation can be controlled by a delay in washing [153], but this would make the whole sequencing process a lot more time-consuming.

[46] Although quality degradation is a known characteristic of 454 sequencing, no noteworthy degradation was observed in single studies [143, 190].

be caused by emPCR errors (see Section 4.2.4. This has not been much explored, but can be modeled or simulated and will be further explored in Section 5.2.

Both in homopolymer length inaccuracies and in CAFIE errors, the combination of insertions and deletions can cause miscalls.

4.3 Data Cleaning – Tools and Strategies

Performing a data cleaning step prior to downstream analysis is crucial for the success of a sequencing project. However, the strategy for error correction and/or data cleaning strongly depends on the application. It includes amongst others the order in which cleaning steps are executed, and the stringency of cleaning. Consequently, there is a large and growing number of freely available bioinformatic tools and software programs for processing genomic data. The overall goal of all cleaning tools and pipelines is to enhance raw data from sequencing platforms to a more reliable level such that later stages of the processing can use the data without concern about base quality [178]. This scenario, however, is unrealistic. In addition, sequence cleaning can result in considerable data loss.

Approaches reach from very specific algorithms tailored to only one sequencing technology (e.g. 454) to hybrid tools that can deal with all NGS data. Some are relicts from the Sanger era that also work – more or less well – on 454 data. In other words, the versatility of a tool or pipeline is both boon and bane since the technologies have very different error patterns. For example, typical errors in 454 sequencing are over- and undercalls while Sanger sequencing mainly suffers from substitution errors [177]. In addition, Sanger sequences are of rather poor quality at the beginning of a sequence and gradually improve, while this is not the case for 454 sequencing [178, 189, 190]. Similarly, NGS technologies vary widely in their characteristic error patterns, which is why some hybrid tools employ separate

– often parametric – error models for each platform. Nevertheless, hybrid tools are rarely sufficiently tailored to the particular error characteristics of a sequencing technology and will therefore not always give satisfactory results.

Some tools require installation and/or configuration and may only work in a specific environment, but deliver accurate results, while other have shiny GUIs, but permit few parameter choices. As sketched above, the variety of application areas requires customizable algorithms and tools.

This section intends to give an insight, but not a complete overview, into existing approaches and tools for making the most out of 454 sequencing data.

4.3.1 Additional Filtering and Trimming

In some applications, correcting or trimming sequences can be seen as more useful than filtering out whole reads that contain errors but are otherwise usable [195]. However, some reads have to be filtered out prior to downstream analysis in order to avoid biases. Such reads include artificial duplicates, contaminated reads and chimeras.

In addition to the removal of artifacts, whole-read filtering and quality-trimming can be useful, e.g. for filtering out reads whose length is far below their expected length. This pattern may give a hint that the read has been sequentially trimmed by the sequencer's software, which again makes it more likely that the whole read is of low quality [143]. This also leads to higher observed indel rates for shorter sequences, and longer sequences tending to have lower error rates [189].

Additional filtering and trimming can either be achieved by changing the thresholds in the 454 filtering and trimming software [174] or by using specific tools. Some of these are mentioned below.

4.3.2 Adapter, Tag and Contaminant Removal

Whenever a DNA fragment from a sequencing library is shorter than read length, the machine sequences into the adapter. Longer reads from newer platforms will worsen this issue [179]. Unfortunately, adapters are not always trivial to detect and remove since they usually lie within the low-quality region of a sequence towards the end of a read, and may contain sequencing errors. The Newbler assembler has a built-in adapter-removal function that may, however, fail to detect adapters in low-quality sequences. Adapter removal should always be performed prior to quality-trimming since the trimming of low-quality bases may hamper the correct identification of the (remainder of the) adapter sequence [91].

One commonly used tool for adapter removal that takes into account quality scores is LUCY [178]. However, LUCY was originally developed for Sanger data and does not take the intrinsic characteristics of the 454 technology into consideration.

Cross_match [232] is targeted to masking and clipping of library-specific primers, adapter sequences as well as screening and elimination of possible contaminants, such as e.g. *E. coli*, phage and yeasts. Cutadapt [233] is another stand-alone adapter-trimming tool. BLAST [234] or GAST [235] can be used to identify and remove contaminants. The identification of tags and MID codes (see Section 3.4) from sequencing pooled samples can be performed by algorithms that are similar to those used for adapter-finding [179].

4.3.3 Chimera Removal

There exist a number of tools targeted to the removal of chimeras.[47] Newer tools that are targeted to NGS data include ChimeraSlayer [211], UCHIME [210], and

[47] However, chimera removal tools dating from the Sanger sequencing era such as e.g. Bellerophon [236] work rather poorly on 454 data.

Perseus [162] that is part of the AmpliconNoise [162] pipeline. ChimeraSlayer requires a reference data set of non-chimeric sequences. Perseus, in contrast, exploits sequence abundances for detecting chimeras, building on the idea that either parent of any chimera must have experienced at least one more PCR cycle than the chimera. This strategy allows for reference-free chimera removal at high sensitivity [162].

4.3.4 Duplicate Removal

Today, most microbial diversity studies involving 454 pyrosequencing reads include a step where duplicates are removed, making use of cd-hit-454 [175], 454 Replicate Filter [226, 230] or similar tools with stringency settings defined by the user.[48] This is an attempt, but not a guarantee to avoid bias when carrying out further analysis on species abundance. Lower stringency allows for tolerating mismatches (substitutions or indels). However, some tools, e.g. MG-RAST [168] and TagCleaner [237], only allow for removal of exact duplicates.

cd-hit-454 [175] is an extension of CD-HIT and performs all-against-all sequence comparisons on 454 reads. Also, a consensus sequence for each group of duplicates is provided. CD-HIT was originally designed to perform clustering of protein sequences. The complexity of sequence analyses had created a need for tools that cluster groups of similar proteins based on their sequence similarity [238, 239]. The idea behind CD-HIT was to apply short word filtering instead of computationally expensive pairwise sequence alignment, and a greedy incremental clustering algorithm. The latter was further extended to, amongst others, nucleotide sequence clustering [240]. With CD-HIT Suite [241], a web server version of CD-

[48] Stringency mainly refers to sequence identity, calculated as the number of identical base pairs in the alignment divided by the full length of the shorter sequence. Often, also a length difference threshold can be set, quantifying which difference in read length is tolerated when assigning sequences to the same cluster [230].

HIT was published, allowing sequence clustering without any local installation and allowing for online visualizations, including a refinement of the original algorithm. A clear advantage of CD-HIT is its speed, further improved through a parallelization strategy applied in the latest version [242].

Dong *et al.* [228] and Pruefer *et al.* [78] use in-house developed scripts for removing duplicates from 454 reads. Another tool for cleaning 454 data from artificial duplicates is contained in the PyroCleaner [243] pipeline (see Section 4.3.6). Also, both Newbler and the Celera/CABOG assembling pipeline have a built-in algorithm to remove duplicate reads.

Obviously, pairwise comparisons between reads are computationally expensive and require sophisticated algorithms, especially when non-exact duplicates are to be detected. This can be seen as the main reason for the fact that – until the day when our JATAC tool was published (see Section 5.3) – duplicate filtering was exclusively performed in nucleotide space.

4.3.5 Base-Calling and Quality Score Calculation

Base-calling, i.e. inferring a DNA sequence from physical signals, is a crucial step of the sequencing process since it directly influences accuracy. Quality scores have to be seen in direct context with base-calling since a quality score expresses the confidence in the base.[49]

One common way of improving accuracy in sequencing projects is to increase coverage and build consensus sequences, leading to lower error rates (see Section 4.1.2). However, this is often associated with high costs and not possible in all application areas. A different strategy therefore consists in enhancing base-calling accuracy, which consequently leads to a reduction of the required coverage [244].

[49] As the authors of the 454 quality score algorithm point out, updates on sequencing platforms may require recalibration of the quality scoring algorithm so that accuracy is kept high. This also includes the choice of noise predictors (see Section 3.2.7) [177].

Furthermore, accurate base-calls and quality scores are crucial in applications where true variation must be distinguished from sequencing errors. For example, with its low substitution error rate, 454 sequencing is particularly suited for SNP discovery. Commonly, SNPs are called from an allele when the quality of the base in question is above a certain cutoff. Both substation base-calling errors and quality scores that over- or underestimate the true base confidence will thus lead to bias in analysis results.

Some research groups have proposed alternatives to the 454 base-caller. At the cost of a slightly higher overall per-base error rate, Quinlan *et al.* [182] reduce substitution errors in order to enhance SNP detection. Their tool PyroBayes makes use of Bayesian statistics in combination with flow value distributions, similarly to the original 454 algorithm for GS 20 quality scores (see Section 3.2.7). For base-calling, the tool calculates the most likely number of incorporated bases given a certain flow value. The quality score assigned to each base is the probability that the base in question is not an overcall, just as in the GS 20 quality score algorithm.[50] Consequently, PyroBayes suffers from the same weaknesses as the GS 20 quality score algorithm. Its quality scores do not reflect the full spectrum of error types, pointing out a need to re-calibrate the PyroBayes algorithm for those platforms that were launched after GS 20.

Another base-calling and quality score tool is HPCall by De Beuf *et al.* [231]. The method uses a probabilistic framework for calling homopolymer lengths. It calculates an estimate that a certain homopolymer length is present given the values of a collection of well-known 454 noise predictors. In addition, probabilities from HPCall are transformed to quality scores. This approach is similar to the Bayesian statistics used by Quince *et al.* [188] and Quinlan *et al.* [182] and by our Flowsim tool (see Section 5.1) and represents the most direct way to quantify base-calling

[50] Unlike the vendor's tools, PyroBayes uses non-central Student's t distributions for modeling flow values.

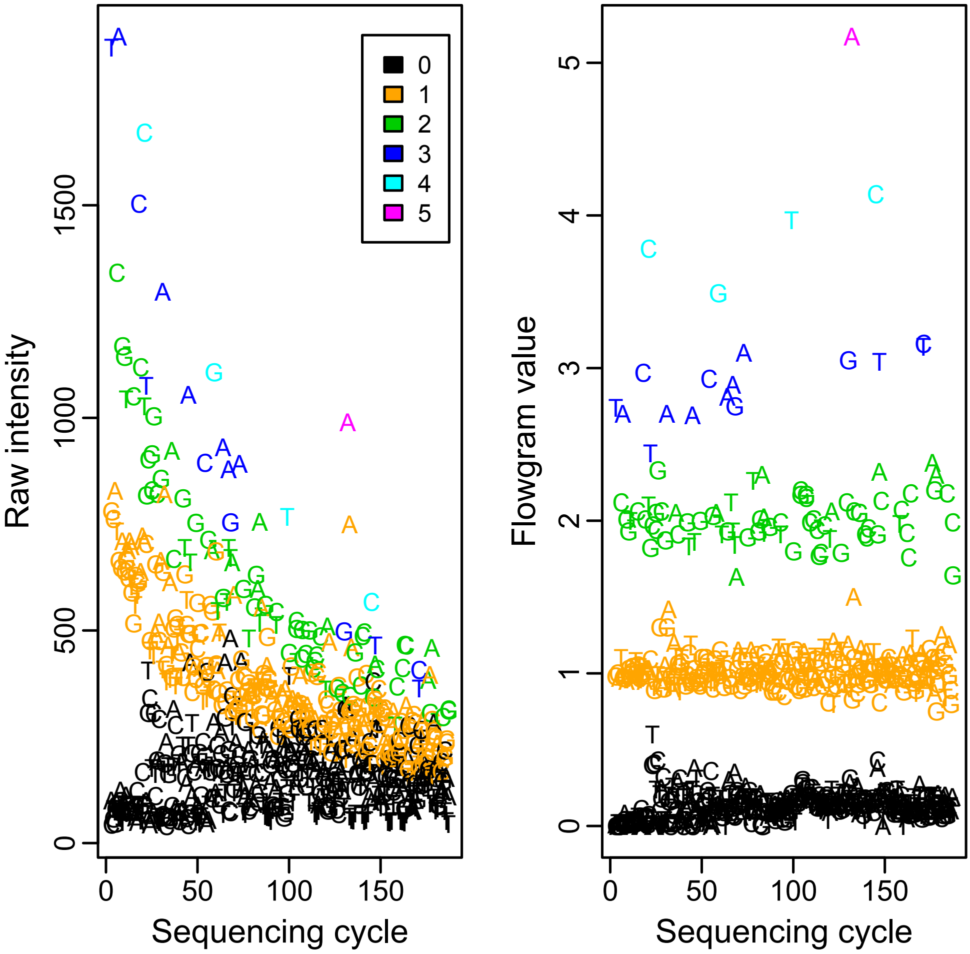

Figure 4.3: Raw intensities (left) and flow values (right) versus cycle number for one read. The colors represent the true homopolymer length. Taken from De Beuf *et al.* [231].

uncertainty. The tool is mentionable for two reasons. Firstly, HPCall quality scores give – in contrast to 454 quality scores – additional information about whether an undercall or an overcall is more likely. HPCall outperforms PyroBayes and accurately determines more high-quality bases than other base-callers including the native 454 base-caller. Secondly, the model that builds the basis for HPCall combines flowgrams and earlier-stage raw intensities (see Section 3.2.4). The authors sketch how the built-in correction algorithms in 454 sequencing remove noise but also otherwise useful information (see Figure 4.3). As opposed to the native 454 base-caller, HPCall employs the additional information from raw data both in base-calling and for quality score calculation.

4.3.6 Multi-Purpose Tools and Pipelines

It may sometimes come handy to have an all-in-one-tool that performs different filtering and trimming steps, and such tools have been published. SeqTrim [204] is a pipeline dedicated to preprocessing any type of sequence read including NGS data, being able to tackle diverse sequencing artifacts as well as chimeras and adapters. SeqClean [245] filters and trims reads by screening for various contaminants, low-quality and low-complexity sequences. The PyroCleaner [243] pipeline implements several filters using criteria such as read length, complexity, the number of Ns, per-base quality. Furthermore, it removes artificial duplicates and is able to filter paired-end reads.[51]

4.3.7 Approaches in Flow Space

Data processing pipelines stemming from the Sanger era usually include a data cleaning step *after* base-calling, i.e. in nucleotide space. To date, this is still common practice. The vast majority of data cleaning tools operate in nucleotide space, which is less computationally expensive than running algorithms in flow space and allows for hybrid use across platforms and technologies. Furthermore, some researchers have the rationale that flowgram data have distorted properties due to correction and normalization within the transition from light signals to flow data [195, 231].

[51] PyroCleaner allows for output in SFF format by using sfffile (see Section 3.3), but does not make use of flow values for filtering and trimming. The duplicate filtering algorithm uses megablast [246].

To the best of my knowledge, the only WGS assembler that takes into account flow value information (and uses it for mitigating sequencing errors) is Newbler, the assembler sold with the 454 platform. In nucleotide space assemblers, the consensus sequence of a contig is determined either by the highest-quality base or based on majority rule (the most frequently encountered base) at each position (see Section 2.4). Here it is determined by averaging flow values [10, 13]. Read similarity for alignments is assessed by directly comparing flowgrams.

Besides HPCall [231] and PyroBayes [182] (see Section 4.3.5), there are a couple of approaches dealing with flow data. Vacic *et al.* [247] suggest matching flowgrams against the target genome for improving results in small RNA discovery.[52] Small RNA discovery is an application field where mitigating errors through building consensus sequences cannot be applied. Lysholm *et al.* [185] present FAAST, an alignment algorithm that uses flowgram data in order to improve alignment accuracy by detecting homopolymer errors. Pruefer *et al.* [78] use an in-house developed flow space program for removing adapter sequences.

Most notably, tools for removing noise in amplicon data in microbial diversity studies have been successfully developed on the basis of flowgram data [162, 188, 248] (see Section 4.3.8). For example, the QIIME software pipeline [249] accepts flow data input and contains modules for a wide range of microbial community analyses and visualizations including OTU clustering and taxa-based diversity analysis within and between samples.

4.3.8 Data Cleaning in Microbial Diversity Studies

In studies on microbial diversity (see Section 2.5), it is common practice to extract DNA from an entire microbial community in environments such as marine, soil,

[52] Each nucleotide sequence can be translated into an "ideal" flowgram by assigning integral values to the flows.

the human hand or the human gut [188, 250]. Often, a particular target (such as a variable region of the 16S rRNA marker gene, see Section 2.5) is amplified by PCR prior to sequencing, which generates an amplicon library [188]. Sequencing such target regions is – unlike shotgun sequencing of genomic data – especially sensitive to errors. Firstly, such studies cannot rely on leveling out errors by consensus building. Secondly, the data may have large numbers of highly similar sequences [143].

In the early years of 454 sequencing, it was questioned if short reads lengths would provide enough accuracy for identifying species in a metagenomic sample.[53] Today, researchers are no longer struggling with putative under-, but over-prediction of diversity.

OTU Clustering – The Basic Ideas

One of the most common strategies in microbial diversity studies is to cluster the amplified sequences (e.g. 16S rRNA) obtained from an environmental DNA sample into a collection of OTUs. Each OTU serves as a proxy for the occurrence of a species or microbial genome [201]. A singleton OTU (i.e. an OTU containing only one sequence) thus represents a rare species. The best evidence for the existence of such a species is its appearance across several samples [250].

Assuming no sequencing errors, the number of OTUs when clustering at 100% identity should thus correspond to the actual number of species in the sample. There are techniques for extrapolating the total number of species from a sample, but the estimates can be heavily influenced by single-member OTUs. Differentiating between novel sequences (that are interpreted as a species) and sequence artifacts such as erroneous reads or chimeras is therefore crucial. Even at the low error rate of 454 sequencing where only a low percentage of reads contain one or more errors (sequencing errors, PCR errors, or chimeras), each erroneous read putatively

[53] Huson *et al.* [136] concluded that reads of 200 bp length would be enough to avoid under-prediction.

leads to the registration of a new species, leading to over-estimates of diversity by up to several orders of magnitude and creating a bias towards low abundances reported [162, 188, 201, 211, 250]. Increasing the size of the data set would further increase inflation. In other words, the extent of a long tail of rare species can reflect true biological diversity, where singleton OTUs represent valid rare phylotypes in diverse environmental samples [201] – or deep molecular sampling could amplify the detrimental effect of sequencing noise (and clustering methods). Whenever the majority of OTUs are supported only by a single read, removing these single reads obviously has great impact on the total number of OTUs [250].

Accurate OTU construction is only possible when sequence differences surpass the level of noise [188]. The distance threshold, i.e. clustering stringency, is sensitive to changes, making it challenging to compare the results of studies where different thresholds have been used. Furthermore, overly stringent clustering can artificially inflate the estimated diversity and composition of a microbial environment [251]. In practice, clustering of reads into OTUs is rarely performed at 100% stringency. Usually, an OTU clustering threshold of 97% is used, for reasons of robustness, i.e. to absorb sequencing errors. This means that sequences that differ by 3% are clustered into a single OTU [176, 201]. As identity thresholds are relaxed, the number of OTUs descreases exponentially.

Also, small differences in OTU methodologies can lead to significantly different OTU structures, thereby affecting ecological conclusions.

These three impact factors – errors, clustering stringency, and OTU methodologies are discussed below.

A first strategy for detecting low-quality reads

In a detailed study on the quality of 454 sequencing data, Huse *et al.* [143] suggested that multi-template beads are the main source of error, referring to GS 20 data. As a conclusion, they recommended to remove reads with one or more unresolved bases (Ns), with errors in the barcode or primer sequence, and atypically short or long read lengths, achieving a substantially lower error rate. The decision to remove reads with Ns resulted from their observation that the presence of even a single N in a read strongly correlates with the presence of further errors. They argued that those beads would frequently lead to undeterminate flows (Ns) since neither base has ample luminescence to clearly register.

The "rare biosphere"

A paper that triggered many reactions was the microbial diversity study by Sogin *et al.* [140] (see Section 2.5). Sogin's research group had been well aware of (at least some of the) 454-intrinsic error patterns. They followed the recommendations of Huse *et al.* (see above), retaining around 90% of the reads. In addition, they only used the first 100 bp after the PCR primer in order to account for quality degradation. Consequently, they concluded that an elevated rate of random sequencing errors was unlikely to explain the extremely high diversity in the sample that they had observed, manifested in an observed tail of highly diverse low-abundance species (the "rare biosphere").[54]

"Wrinkles in the rare biosphere"

In order to quantify the effect of quality-filtering (and OTU threshold choice) on diversity estimates, Kunin *et al.* [176] analyzed the impact of the data cleaning suggested by Huse *et al.* [143], an additional quality-trimming based on quality scores (with LUCY [178]), and different OTU clustering thresholds. They proved the read-filtering practice for GS 20 data [143] described above to be not strict

[54] The question of which percentage of a microbial data set is regarded as "rare", i.e. the cutoff threshold that divides abundant from rare, and which impact this has on downstream analysis, is discussed in Gobet *et al.* [252].

enough for microbial diversity analyses carried out on FLX data.[55] Substantial noise remained after this data cleaning process.

Even when lowering the OTU clustering threshold to the commonly used stringency of 97%, the previously suggested quality-filtering and -trimming was insufficient to ensure accurate diversity estimates [176]. Only when an additional quality-trimming was performed – using a per-base error probability of 0.2% as a cutoff in LUCY [178] – the artefactual inflation of diversity could be reduced. Further trimming at an even lower cutoff did not produce better results, but in a sharp decrease of usable reads.

It became obvious that and to what extent diversity estimates are sensitive to the abundance of rare members of a community and how easily they are confused by sequencing noise [251]. In other words, the "rare biosphere" observed by Sogin *et al.* [140] was probably not as large as previously assumed [250].

"Ironing out wrinkles in the rare biosphere"

Previous studies on diversity estimate biases either focused on the impact of pyrosequencing errors or on alignment methods used in clustering. However, also other sources than pyrosequencing errors can inflate OTU estimates, namely the applied clustering algorithm [201]. The common method of complete-linkage clustering was found to favor the inflation of OTU estimates due to sequencing noise (see Section 5.3).

A new strategy towards a more accurate characterization of microbial diversity was presented with PyroNoise [188]. PyroNoise uses a flowgram clustering algorithm, building on the knowledge that two sequences can substantially differ, but still have very similar flowgrams. Using flowgrams and distributions of flow values and thus modeling sequencing noise, Quince *et al.* define a probability that a flow-

[55] Data cleaning according to Huse *et al.* [143] resulted only in a marginal improvement (1%) in errorless reads as opposed to >15% of the reads containing one or more errors [176].

gram was generated by a given sequence. Noise removal, referred to as flowgram pre-clustering, predicts from this probability whether a read is noise or a genuinely novel sequence.[56] PyroNoise involves, following noise removal, screening for PCR chimeras[57], a measure that further reduces the number of incorrect OTU assignments during the clustering step.

Unfortunately for most research groups, PyroNoise's computational demands are beyond the capabilities of most individual laboratories [250]. In addition to operating in flow space (gold standard in terms of accuracy), PyroNoise performs all-on-all comparisons. The flowgram clustering approach used in PyroNoise accounts for two facts: that sequences with errors are likely to be rare, and that they should be similar to a true abundant sequence [162].

DeNoiser [248] exploits rank-abundance distributions, performing pre-clustering on read suffices and comparing unclustered reads to the most abundant clusters (represented by their centroids). This builds on the assumption that error-free sequences will occur more frequently than their error-induced variants [201], in compliance with the observation that a majority of reads are error-free (see Section 4.1). Those sequences that accurately represent the template pool will therefore preferentially seed the establishment of a new cluster rather than erroneous sequences that occur at lower frequency [201].[58] Huse *et al.* [201] pursue a similar approach in nucleotide space rather than in flow space, which makes the process even faster. They refer to their method as single-linkage pre-clustering (SLP) (followed by average-linkage clustering).

With an updated version of PyroNoise, called AmpliconNoise [162], Quince *et al.* made a sophisticated approach toward the accurate determination of microbial diversity. AmpliconNoise couples a flowgram clustering step without alignments,

[56] In addition, PyroNoise trims any read as soon as a single flow value between 0.5 and 0.7 is observed, and discards the whole read if the remaining sequence has less than 200 bp.

[57] using an adaption of the Mallard algorithm [253]

[58] However, there is a risk that high frequency chimeras are identified as cluster seed [162].

still called PyroNoise, followed by nucleotide space clustering with SeqNoise. The latter performs alignments and attempts to filter out PCR errors by calculating nucleotide transition probabilities. Splitting the removal of pyrosequencing noise from that of PCR error allows for the use of more appropriate models and consequently to a more sensitive artifact filtering. Furthermore, computational costs are reduced because the fast alignment-free flowgram clustering reduces the data set size for the slower sequence clustering. Both steps employ similar probabilistic models (see above).

When filtering errors with PyroNoise and SeqNoise, pyrosequencing errors were found to account for roughly half of the extra diversity. The majority of the remaining errors are due to PCR substitution errors.[59] However, some spurious OTUs remain, and these are usually caused by chimeras.[60] The latter can be removed with Perseus (see Section 4.3.3). The described strategy of removing sequencing errors, PCR errors and chimeras allows for an accurate OTU construction, outperforming previously published agglomerative clustering tools such as DeNoiser and SLP clustering (see above) both in terms of per-base error rates and OTU construction [162].[61]

[59] Similarly to the previous version of PyroNoise (see above), reads are truncated as soon as a single flow value between 0.5 and 0.7 or an undetermined base (N) is observed. Reads are discarded if this occurs before flow 360 both for FLX and Titanium. In order to account for quality degradation, the last 10% of flows are trimmed, i.e. at flow 360 for FLX and 720 for Titanium.

[60] AmpliconNoise was found capable to reduce noise by one-third to a half [162] in different data sets.

[61] AmpliconNoise shows significant improvements, both in OTU clustering and speed, over the original PyroNoise program [162].

5

Contributions and Discussion

Modeling is like vintage wine; it matures with time.

Unknown

A recurring theme throughout this thesis is that the key to effective use of sequencing data in downstream analysis lies in the identification of characteristics associated with noise. This includes modeling the flow values, calculating measures for data accuracy and applying filtering and trimming mechanisms to the reads. Many research groups have reported that the 454 default filtering and trimming is not sufficient for their purposes.

This chapter puts the three papers that contribute to this thesis into context. A discussion of the results of each paper provides a basis for further research. The papers are closely related to each other since all of them deal with error characteristics of 454 data. Another common key aspect of the contributions is that all operations and analyses are performed in flow space in order to provide a maximum level of accuracy with minimal information loss. In other words, flowgram data capture the varying levels of system noise and sequencing error better than nucleotide sequences [177].

5.1 Modeling and Simulation

In the first paper “Characteristics of 454 pyrosequencing data - enabling realistic simulation with Flowsim”, we provide a detailed numerical and visual analysis of the main error source in 454 data, namely homopolymer errors, also in the context of synchronism loss and quality degradation.

How to Model Flow Values

When modeling sequencing errors in flow space, it is essential to make assumptions about the underlying flow value distributions. The overlap character of these distributions is responsible for a large percentage of base-calling errors (see Section 3.2.4). Margulies *et al.* [10] had earlier modeled the data by a set of Normal distributions (see Section 3.2.7), Quinlan *et al.* [182] fitted non-central Student’s *t* distributions to the data. In contrast, we found all parametric distributions to fit poorly (data not shown). It seemed therefore a logical consequence to use empirical instead of parametric distributions. We calculated these by aligning flowgrams to the matching genomic region, assigning each flow value to the corresponding true homopolymer length as known from the reference (see Figure 3.5 right). The analysis was carried out on sequencing data from *E. coli* and sea bass (*Dicentrarchus labrax*) and provided us with a good basis to create a simulator that mimics characteristics of 454 sequencing data.

The fact that other NGS technologies have different error patterns [58, 254], for example substitution errors being the most abundant error type in Illumina sequencing [18, 255], emphasizes on the need for a tool that closely models the 454-intrinsic errors. We found our empirical distributions to reflect flow values in a considerably more accurate way than the parametric approaches mentioned above. This is one of the strengths of the Flowsim pipeline. However, one risk with using data from only two species for building empirical distributions that are later used in simulation is that the distributions may not be representative for other species. Due to the unfortunate lack of other data, our approach was the only possible way to go. Through smoothing of distributions and validating a separate *E. coli* model on *D. labrax* data and vice versa, we could at least avoid overfitting issues.

For calculating quality scores from simulated flow values, our Bayesian approach is not very different from that of Quinlan *et al.* (see Section 4.3.5). The main difference between the two algorithms is that we calculate the probability for a certain homopolymer length given a flow value, not the probability for an overcall. In our Bayesian approach, the posterior probabilities are calculated from the data likelihoods (the empirical flow value distributions, see above) and the priors. The latter reflect homopolymer probabilities and are calculated from the average homopolymer lengths of the *E. coli* and *D. labrax* data.[62] However, complex organisms contain longer homopolymers than bacterial genomes. Using an average can only be an approximation of the true homopolymer length distribution. It would be theoretically possible – but adds computational complexity to the Flowsim algorithm – to estimate homopolymer length distributions from the sequences that are used as input to the simulation tool. A possible bias through sequencing errors should be negligible.

In general, it is hard to assess the impact of an error source or artifact on an application. In most cases, it will be necessary to not only have sequencing data, but also a reliable reference. This can be e.g. a reference genome in the case of genome assembly, a known diversity in a metagenomic sample, or the control DNA sequences provided by the manufacturer.

[62] Using other priors that do not “fit” the data likelihoods would lead to a false application of the Bayesian formula.

Why Simulate?

In brief, producing simulated reads allows rapid generation of large numbers of sequencing libraries with controlled and predefined parameters [256]. Simulation facilitates the design of sequencing projects. For example, previous feasibility studies of *de novo* sequencing of large and complex genomes (see Section 2.4.4) would clearly have benefited from simulations for examining and quantifying the impact of read length, coverage, sequencing errors and quality degradation on assembly quality. Further questions raised by assembly projects are e.g. how well a known genome can be reconstructed from reads with certain characteristics, or how well large genomic rearrangements can be detected [190].

Testing new algorithms is another application area for simulators. Proper algorithm design and implementation require large amounts of sequence data, and such data is rarely available in the volume necessary for rigorous testing [256]. The construction of *in vitro* libraries in the laboratory is expensive and labor intensive. Simulation overcomes these limitations and, in addition, allows for optimization of default parameters.

Also, assessing and benchmarking existing methods and tools in a number of application areas such as read alignment, read correction, SNP identification and metagenomics largely profits from being able to create large amounts of data *in silico*. For example, the impact of a stricter whole-read filtering and read-trimming can be examined.

Lastly, but no less significant, simulations allow for assessing the potential of future generations or enhancements of the sequencing platform.

The Flowsim Suite (I)

Flowsim is a suite of tools or modules for simulating the 454 pyrosequencing process. It is based on the characteristics of real 454 data, and attempts to model the known aspects of the process. The tool was programmed in Haskell by my advisor, Ketil Malde, and is documented on the Flowsim website [257].

The original version described in the paper consists of two modules, Clonesim and Flowsim. Clonesim simulates the shearing step, breaking the input sequence(s) into random fragments. The distribution of read lengths can be specified by the user, choosing between a number of parametric distributions. Flowsim mimics the actual sequencing process, converting homopolymer lengths to flow values. Apart from the empirical distributions described above, also parametric flow value distributions can be chosen. The resulting flows are base-called, quality-filtered and -trimmed, and assigned quality scores.[63] Reads are output in SFF format, which allows for experimenting with software that operates in flow space. Public tools can be used to write to FASTA or FASTQ output (see Section 3.3).

Flowsim was used for the validation of a method for viral quasispecies spectrum reconstruction [258], a new strategy for complete prokaryotic genomic sequencing [259] and a new metagenomics gene prediction system [260]. A number of e-mails from Flowsim users have revealed that also other researchers are successfully using our simulation pipeline, and feedback is largely positive.

Other Simulators

MetaSIM [138], building on its unpublished pre-version ReadSIM, is a versatile read simulator targeted to designing metagenomic projects and to testing and benchmarking metagenomic or assembly software [138].[64] However, MetaSIM is neither targeted to NGS data, nor does it produce quality scores.

After the release of Flowsim, several other simulation tools were published (for

[63] Quality-filtering and -trimming was implemented in accordance with the algorithms described in the 454 manual [174]. However, their documentation is not clear enough to ensure that they were implemented correctly in Flowsim.

[64] Researchers have used MetaSIM e.g. for construction of a synthetic metagenome [261] and for testing a new metagenome clustering and annotation pipeline [262].

a comparison see Table 5.1). These are described below.

GemSIM [190] is targeted at simulation studies where a reference is available. An alignment of control data in SAM format[65] is used as input for calculating an error model, considering a sequence context of five bases (three before and one after the base in question) and the sequence position.[66]

MASON [264] uses a simple parametric model for simulating 454 reads, building on the Normal and log-normal distributions described in Margulies *et al.* [10]. Most notably, no quality degradation as described in Section 4.2.7 is provided, and neither has quality-trimming been implemented. Furthermore, the paired end model creates 2x450 bp reads, but the 454 paired end protocol produces both sequences in the same pair, joined by a linker (see Section 2.2). The only advantage of MASON over Flowsim (apart from considerably lower runtime) is the useful SAM format output (when a reference is provided) that enables the user to carry out further data analyses.

ART [265] is a read simulator for the three second-generation sequencers (454, Illumina, SOLiD) and was initially developed for read simulation in the context of the 1000 Genome Project [54]. It is one of the few simulators to date that enables simulation of 454 paired end reads.

454sim [266] is highly similar to the Flowsim tool, implemented in C++, and multi-thread capable, thus a lot faster than Flowsim. To date, 454sim is the only data simulator apart from Flowsim that provides flow data output in the form of SFF files. Lysholm *et al.* [266] adapted all degradation modeling, quality score calculation etc. from Flowsim, but fall back to a parametric model for flow value distributions. Instead of performing a run-time benchmark 454sim vs. Flowsim, it would be a lot more interesting to see how much more realistic one can simulate

[65] SAM is a generic format for storing large nucleotide sequence alignments [263].

[66] For those users who do not have access to control data, an error model from the study is provided, based on plasmid data from a Hepatitis C Virus study.

454 pyrosequencing data by using empirical flow value distributions.

Grinder [256] is perhaps the most sophisticated and versatile simulator currently available. It is the first tool to simulate amplicon datasets, but is also able to produce shotgun (genomic, metagenomic, transcriptomic and metatranscriptomic) datasets. Amplicon simulation involves creating sequences with a certain community structure and mimicking the PCR process including biases and errors.

Grinder is very suitable for use in combination with Flowsim, especially in Grinder's core strength – amplicon sequencing with PCR simulation. Microbial community data can be simulated with Grinder by using the species abundance models provided. PCR errors, chimeras, and PCR bias can be introduced. Subsequently, Flowsim could be run on the resulting FASTA sequences, introducing pyrosequencing noise.[67]

A feature that remains to be implemented into Flowsim is the option to create paired end reads. Analogously to Grinder, it would also be very helpful to include detailed information for each read in the output, including its location on the reference sequence and introduced errors, making reads traceable for downstream analysis and applications [256].

[67] For realistic 454 quality scores, the authors recommend to use Flowsim subsequent to read simulation with Grinder.

Tool	454 Error Model	Output Formats	Application Area	Special Features
MetaSIM	Parametric error model (in nucleotide space); incorporation of user-defined, position-specific error statistics possible	FASTA	Simulator for 454, Illumina, and Sanger, targeted to metagenomics	Genome evolution model; empirical error model for Illumina, adaptable to other NGS technologies; parametric read length distribution; paired ends
Flowsim	Empirical and parametric error models (in flow space); quality degradation	SFF	454 simulator	Embedding of user-defined empirical error models possible; artificial duplicates
MASON	Parametric error model (in flow space)	FASTA/FASTQ; alignments in SAM format	Allround simulator for 454, Illumina, and Sanger	Position-specific error rates; quality scores
454Sim	Parametric error model (in flow space); quality degradation	FASTA/FASTQ and SFF	454 simulator	Multi-thread capable, thus very fast with 1̃0,000 Titanium reads per second; quality scores analogous to Flowsim
ART	Context-dependent error model with homopolymer-specific insertion/deletion error distributions (in nucleotide space)	FASTA/FASTQ; alignments in ALN, SAM or UCSC BED format	Allround simulator for 454, Illumina, and SOLiD	Empirical read length distribution; quality scores are modeled as homopolymer-specific first order Markov chains
GemSIM	Empirical error model (in nucleotide space) that relies on reference	FASTQ	Simulator for 454, Illumina, and Sanger, best suited for simulating resequencing or metagenomic data where a reference is available	Quality scores, but calculated from nucleotide space
Grinder	Parametric error model (in nucleotide space)	FASTA/FASTQ	Amplicon and shotgun simulator for 454, Illumina, and Sanger	PCR simulation (chimeras, PCR bias); species abundance/diversity models; parametric read length distribution; paired ends

Table 5.1: Comparison of existing sequencing read simulators.

5.2 Error Sources

After the first release of Flowsim, we became aware of the fact that the data produced by Flowsim contained too little noise, i.e. were of too high a quality. When assembling simulated data with Newbler, we obtained more correct assemblies than when assembling real data. This led to the conclusion that we were facing unknown error sources. Consequently, we decided to expand the Flowsim pipeline by adding further modules (see below).

Neighboring peaks – a mystery

While analyzing our data for the Flowsim paper, we observed peaks in neighboring flow value distributions (see Figure 3.5 right) which we, at that time, explained by true biological differences. When sequencing amplicon products, such peaks can be explained by PCR errors from the library preparation step (see Section 4.2.4). Consequently, we proposed and pursued the idea to analyze flowgrams of paired end linker sequences in order to further characterize error patterns in flow values. Since we used the 42 bp linker sequences only, we could minimize the risk of biological differences.

The data we built our analyses on were shotgun sequences from *G. morhua* that did not undergo any PCR step for library preparation, such that PCR errors could be excluded as a source of error. We therefore suspected emPCR errors to be responsible for the artifact, but there is no reason to assume that such emPCR errors would peak on the integers (see Section 4.2.4). In brief, the neighboring peaks remain a mystery. The presence of misaligned DNA pieces that almost match the linker sequence cannot be ruled out, although we made an attempt to exclude such alignments by rigorous filtering of the sequences included in our analysis.

It would be interesting to see if different data sets for other species showed the same pattern, in particular the same error variants. In addition, a deeper analysis of the observed erroneous linker sequences could reveal whether those mostly contain indels or also comprise substitution errors. The latter would suggest a comparison with nucleotide transitions that are typical for PCR errors (see Section 4.1.1). In addition, duplicate filtering should be applied prior to performing the same analysis again.

The Flowsim Suite (II)

The most important among the new modules are Kitsim, Mutator, and Duplicator. These are typically run after Clonesim and prior to transforming sequences into flow space with Flowsim.

Kitsim attaches adapter sequences to the reads.[68] Mutator is a utility for introducing random indels and substitution errors into the sequences. This takes account to the neighboring peaks and can further be used for mimicking PCR errors since the indel and the substitution rate can be specified separately. Finally, Duplicator creates artificially duplicated reads.

The various tools are designed in a modular way, and each module uses the FASTA format for input and output (apart from Flowsim that produces SFF output). This makes it possible to replace individual steps with other programs.

5.3 Duplicate Read Removal

The presence of artificial duplicates (see Section 4.2.5) is mainly an issue in microbial diversity studies, but also in a couple of other research areas such as SNP discovery [193] and structural variation detection [267]. At low coverage, an already low number of duplicates can have a marked impact.

[68] Kitsim was earlier included in the Flowsim module. The separation from Flowsim allows for introducing duplicates and random errors prior to transforming sequences into flow space.

However, a general problem inherent to duplicate filtering is that artifical duplicates cannot be distinguished from natural duplicates. The risk of unwanted removal of natural duplicates leads to the fact that some research groups omit any duplicate filtering step, especially when a project includes both amplified and non-amplified samples [91, 268].

In contrast to microbial diversity, the effect of duplicates on *de novo* genome assembly has been (too) little examined. It is obvious that artificially duplicated reads are a waste of coverage and do not add value to an assembly.[69]

454 sequencing is free from cloning bias (see Section 4.2), but can still suffer from a substantial coverage bias due to artificial duplicates and other factors. The extent to which duplicate filtering evens out coverage would therefore be worth analyzing.[70]

The main weakness of JATAC is that the additional computational costs when compared to tools that operate in nucleotide space (see Section 4.3.4) may – depending on the application area – not be outweighed by its enhanced accuracy in duplicate filtering. This shows a need for fine-tuning the algorithm and testing it rigorously on a large number of data sets with a reference available. Optimally, JATAC could be integrated into the 454 quality-filtering pipeline by Roche Diagnostics. A good e-mail contact with Roche Diagnostics has revealed their strong interest in enhancing the accuracy of the algorithms built into the sequencing pipeline.

Another issue is the lack of further datasets for testing the algorithm. The fact that flow space-based tools such as JATAC perform more or less well on different

[69] During the assembly of the Atlantic cod genome (see Section 2.4.4) some runs were observed to worsen overall assembly quality when they were added (Ketil Malde, Institute of Marine Research, pers. comm., November 1st 2010). Although Newbler, the assembler mainly used for assembly in that project, involves a duplicate filtering step, it cannot be excluded that the low overall quality of those runs can be seen in context with artificial duplicates.

[70] Since duplicates are likely to arise during emPCR, all SFF files (usually two) representing one run have to be combined before filtering is carried out. This is also true for FASTA files when using tools that operate in nucleotide space.

datasets points towards the variation in the extent to which datasets suffer from different error types.

Improved clustering methods

Huse *et al.* [201] and Quince *et al.* [162, 188] have extensively discussed the impact of the choice of clustering algorithm on (OTU) clustering accuracy, comparing complete-linkage, average-linkage, and single-linkage (see Section 4.3.8). The difference between these algorithms is, in brief, that they employ different rules to determine whether a new sequence is added to an existing cluster or forms a new cluster. At an identity threshold of 97%, complete-linkage requires that a new sequence is less than 3% different from *each* sequence that is already present in the cluster. Average-linkage requires that the *average* difference between the new sequence and each sequence in the cluster is no more than 3%. Single-linkage requires only that the new sequence has less than 3% difference from *at least one* sequence already present in the cluster.

One of the problems in clustering sequences or flowgrams is that noise confuses the clustering process, making the latter more sensitive to methodological differences. Complete-linkage was found to inflate the number of estimated OTUs because, with an increasing level of noise, it is decreasingly likely that a sequence will meet the requirement that it is less than 3% different from *each* sequence in an existing cluster, and it is thus more likely that the sequence will form a new cluster [201]. Both in OTU and in duplicate clustering, it is important to not only get the right number of clusters but also the correct assignment of sequences to clusters [162].[71]

We can greatly profit from the lessons learned in OTU clustering (see Sec-

[71] White *et al.* [251] found complete-linkage clustering – despite its sensitivity to sequencing noise – form OTUs with a closer correspondence to true composition when compared to average-linkage clustering.

tion 4.3.8). The characteristics of the different clustering approaches reveal a necessity to put more effort into the JATAC algorithm, yielding maximum similarity of clusters to true duplicate clusters and minimum sensitivity to sequencing noise. Visual and numerical analyses of true flowgram clusters have revealed that flowgram clustering in duplicate removal has great potential, and the refinement of the algorithm can be supposed to reach unprecedented accuracy.

5.4 Closing Remarks

Pyrosequencing allows for reliable high-resolution sequence detection and quantification and provides a high level of accuracy. It is relatively cost-efficient, and the error-prone and time-consuming cloning step required for Sanger sequencing is avoided. However, the future of 454 sequencing is uncertain. Only few research groups are currently using the latest platform, FLX+, and even longer reads are improbable due to known issues related to CAFIE effects. With Illumina and Ion Torrent read lengths slowly approaching FLX at low error rates and PacBio generating substantially longer reads than 454 (although at a low signal-to-noise ratio, with error rates around 15%, see Section 1.2), 454 sequencing has decreasing importance among the NGS platforms.

One particularly promising characteristic of third-generation sequencing technologies such as PacBio is that they require neither PCR nor emPCR amplification, which in consequence reduces errors and biases. Synchronization becomes unnecessary such that dephasing is no longer an issue, likewise artificial duplicates. PCR amplification can be omitted because the preparation of single-molecule templates requires less starting material. This avoids both PCR bias (see Section 4.2.2) and PCR errors. Quantitative applications such as diversity studies and RNA-seq perform more accurately and effectively with non-amplified template sources such that the original representational abundance of molecules is retained. In brief, third-generation sequencers follow promising approaches to reduce the time, error and cost currently associated with template preparation, PCR amplification and the actual sequencing associated with wash-and-scan techniques.

However, the future of 454 is not that bleak. Laboratories in possession of a 454 sequencing machine will continue to use 454. With respect to this PhD thesis, it is worth exploring which techniques are applicable to other platforms, both existing and future ones. For example, Ion Torrent uses a similar flow approach that is sensitive to homopolymer errors. We have run a couple of tests of JATAC on Ion Torrent data, without convincing success. Other tools that build on more general probabilistic frameworks have, however, revealed promising results [269].

The effect of emPCR errors on sequencing quality

One of the main issues in 454 sequencing and limiting factors for yielding longer reads are errors introduced through loss of synchronism. Besides from CAFIE errors occurring during the actual sequencing process, emPCR errors can also lead to CAFIE effects (see Section 4.2.4).

Both PCR bias (see Section 4.2.2) and PCR errors (see Section 4.2.4) have been extensively discussed in the research community. The PCR error rate has been found to determine the fundamental limit of the ability of deep resequencing to detect non-artifactual single-base substitutions in PCR amplicons [197]. However, it is a common misbelief that emPCR errors do not cause much noise in the resulting flowgrams since millions of templates are combined to obtain a consensus. There are several ways of providing an estimated upper bound for the impact of emPCR errors, e.g. parametric approaches (using the binomial equation) or simulations. The relevant question to ask is how many of the copies can be expected to be error-free. This depends largely on the assumed substitution error rate, the number of PCR cycles, and on sequence length.[72] A simulation framework would be able to perform *in silico* PCR amplification of a given template sequence with a certain parametrization, generating a collection of sequences each differing from their template by zero or more substitution errors. Changing parametrization allows for assessing the impact on possible read lengths. Results from simulations reveal that emPCR errors will result in highly inaccurate sequences as read length increases.[73] In brief, this shows that pyrosequencing beyond 1,000 bp will not be possible at a low error rate comparable to that observed for the Titanium platform.

In addition, emPCR errors contribute to quality degradation. Quantifying this impact involves calculating similarity to the template along the sequence. Similarly, it would be interesting to analyze to what degree emPCR errors account for CAFIE errors.

Last, but not least, it should be mentioned that each analysis result is the combined effect of the laboratory methods recommended by the manufacturer, read alignment tools, base-calling algorithms and a number of other components [58]. These components partially contribute to quality problems and therefore need to be simultaneously optimized. For example, experimental issues (e.g. determining the optimal ratio of DNA to beads) that can be controlled by the user can account for a high error rate [195]. Data processing methods can introduce further error, and users are often uncertain how their choice of methods and tools will affect the interpretation of their data [269]. In addition, all methods putatively involve unexpected effects that may not become evident before comparisons across methods are carried out. However, methodological impacts on analysis results are often carried into publications – while few datasets are ever re-evaluated with updated methodologies. One example where such re-evaluation would, in fact, be strongly recommended,

[72] The Taq polymerase used in the 454 sequencing protocol has an error rate of one substitution in 9,000 bases [214].

[73] Inge Jonassen, University of Bergen, unpublished results, 2010

are microbial diversity studies.

Bibliography

[1] Sanger, F., Nicklen, S., and Coulson, A. R. “DNA sequencing with chain-terminating inhibitors”. *Proceedings of the National Academy of Sciences of the USA* 74(12) (1977), pp. 5463–7 (cited on page 3).

[2] Maxam, A. M. and Gilbert, W. “A new method for sequencing DNA”. *Proceedings of the National Academy of Sciences of the USA* 74(2) (1977), pp. 560–4 (cited on page 3).

[3] Smith, L. M. et al. “Fluorescence detection in automated DNA sequence analysis”. *Nature* 321(6071) (1986), pp. 674–9 (cited on page 3).

[4] Prober, J. M. et al. “A system for rapid DNA sequencing with fluorescent chain-terminating dideoxynucleotides”. *Science* 238(4825) (1987), pp. 336–41 (cited on page 3).

[5] Adams, M. D. et al. “A model for high-throughput automated DNA sequencing and analysis core facilities”. *Nature* 368(6470) (1994), pp. 474–5 (cited on page 3).

[6] Karger, A. E. “Separation of DNA sequencing fragments using an automated capillary electrophoresis instrument”. *Electrophoresis* 17(1) (1996), pp. 144–51 (cited on page 3).

[7] Lander, E. S. et al. “Initial sequencing and analysis of the human genome”. *Nature* 409(6822) (2001), pp. 860–921 (cited on pages 3, 27).

[8] Venter, J. C. et al. “The sequence of the human genome”. *Science* 291(5507) (2001), pp. 1304–51 (cited on pages 3, 27).

[9] Service, R. F. “Gene sequencing. The race for the 1000*genome*”. *Science* 311(5767) (2006), pp. 1544–6 (cited on page 4).

[10] Margulies, M. et al. “Genome sequencing in microfabricated high-density picolitre reactors”. *Nature* 437(7057) (2005), pp. 376–80 (cited on pages 4, 5, 8, 18, 25, 27, 33, 35–37, 39–42, 44, 47, 51, 57, 58, 67, 68, 77, 86, 90).

[11] Rothberg, J. M. and Leamon, J. H. “The development and impact of 454 sequencing”. *Nature Biotechnology* 26(10) (2008), pp. 1117–24 (cited on pages 4, 8, 22).

[12] Goldberg, S. M. et al. “A Sanger/pyrosequencing hybrid approach for the generation of high-quality draft assemblies of marine microbial genomes”. *Proceedings of the National Academy of Sciences of the USA* 103(30) (2006), pp. 11240–5 (cited on pages 5, 25).

[13] “Roche 454 Sequencing”. http://www.454.com/products/technology.asp. (accessed February 2009) (cited on pages 5, 9, 14, 36–39, 50, 51, 53, 67, 77).

[14] Schuster, S. C. “Next-generation sequencing transforms today’s biology”. *Nature Methods* 5(1) (2008), pp. 16–8 (cited on pages 5, 9, 11, 13, 23, 59).

[15] Mardis, E. R. “Next-generation DNA sequencing methods”. *Annual Review of Genomics and Human Genetics* 9 (2008), pp. 387–402 (cited on pages 5, 7, 8, 15, 51).

[16] Bentley, D. R. et al. “Accurate whole human genome sequencing using reversible terminator chemistry”. *Nature* 456(7218) (2008), pp. 53–9 (cited on page 5).

[17] Valouev, A. et al. “A high-resolution, nucleosome position map of C. elegans reveals a lack of universal sequence-dictated positioning”. *Genome Research* 18(7) (2008), pp. 1051–63 (cited on page 6).

[18] Metzker, M. L. “Sequencing technologies - the next generation”. *Nat Rev Genet* 11(1) (2010), pp. 31–46 (cited on pages 6, 9, 10, 15, 18, 21, 22, 35, 41, 51, 57, 68, 86).

[19] Mardis, E. R. “A decade’s perspective on DNA sequencing technology”. *Nature* 470(7333) (2011), pp. 198–203 (cited on pages 6, 7, 13, 15, 52).

[20] “Ion Torrent semiconductor sequencing”. http://www.invitrogen.com/site/us/en/home/brands/Ion-Torrent.html. (accessed February 2013) (cited on pages 6, 7).

[21] Elliott, A. M. et al. “Rapid Detection of the ACMG/ACOG-Recommended 23 CFTR Disease-Causing Mutations Using Ion Torrent Semiconductor Sequencing”. *Journal of Biomolecular Techniques* 23(1) (2012), pp. 24–30 (cited on page 6).

[22] Shendure, J. et al. “Accurate multiplex polony sequencing of an evolved bacterial genome”. *Science* 309(5741) (2005), pp. 1728–32 (cited on page 6).

[23] Hutchison, C. A. “DNA sequencing: bench to bedside and beyond”. *Nucleic Acids Research* 35(18) (2007), pp. 6227–37 (cited on page 6).

[24] Pop, M. and Salzberg, S. L. “Bioinformatics challenges of new sequencing technology”. *Trends in Genetics* 24(3) (2008), pp. 142–9 (cited on page 7).

[25] Shendure, J. and Ji, H. “Next-generation DNA sequencing”. *Nature Biotechnology* 26(10) (2008), pp. 1135–45 (cited on page 7).

[26] Mardis, E. R. “The impact of next-generation sequencing technology on genetics”. *Trends in Genetics* 24(3) (2008), pp. 133–41 (cited on page 7).

[27] Pettersson, E., Lundeberg, J., and Ahmadian, A. “Generations of sequencing technologies”. *Genomics* 93(2) (2009), pp. 105–11 (cited on page 7).

[28] Zhou, X. et al. “The next-generation sequencing technology and application”. *Protein Cell* 1(6) (2010), pp. 520–36 (cited on page 7).

[29] Kahvejian, A., Quackenbush, J., and Thompson, J. F. “What would you do if you could sequence everything?” *Nature Biotechnology* 26(10) (2008), pp. 1125–33 (cited on page 7).

[30] Harris, T. D. et al. “Single-molecule DNA sequencing of a viral genome”. *Science* 320(5872) (2008), pp. 106–9 (cited on page 7).

[31] Schadt, E. E., Turner, S., and Kasarskis, A. “A window into third-generation sequencing”. *Human Molecular Genetics* 19(R2) (2010), R227–40 (cited on pages 7, 15, 20, 23).

[32] Blow, N. “DNA sequencing: generation next-next”. *Nature Methods* 5(3) (2008), pp. 267–72 (cited on page 7).

[33] Eid, J. et al. “Real-time DNA sequencing from single polymerase molecules”. *Science* 323(5910) (2009), pp. 133–8 (cited on page 7).

[34] “PacBio single molecule, real-time sequencing”. http://www.pacb.com/brochure. (accessed January 2013) (cited on pages 7, 9, 11).

[35] Branton, D. et al. “The potential and challenges of nanopore sequencing”. *Nature Biotechnology* 26(10) (2008), pp. 1146–53 (cited on page 7).

[36] Venkatesan, B. M. and Bashir, R. “Nanopore sensors for nucleic acid analysis”. *Nature Nanotechnology* 6(10) (2011), pp. 615–24 (cited on page 7).

[37] Quinn, N. L. et al. “Assessing the feasibility of GS FLX Pyrosequencing for sequencing the Atlantic salmon genome”. *BMC Genomics* 9 (2008), p. 404 (cited on pages 8, 28, 51).

[38] Droege, M. and Hill, B. “The Genome Sequencer FLX System–longer reads, more applications, straight forward bioinformatics and more complete data sets”. *Journal of Biotechnology* 136(1-2) (2008), pp. 3–10 (cited on pages 8, 13, 27, 39, 51, 53, 57).

[39] Wicker, T. et al. “Low-pass shotgun sequencing of the barley genome facilitates rapid identification of genes, conserved non-coding sequences and novel repeats”. *BMC Genomics* 9 (2008), p. 518 (cited on page 8).

[40] Ratan, A. et al. “Comparison of sequencing platforms for single nucleotide variant calls in a human sample”. *PLoS One* 8(2) (2013), e55089 (cited on pages 8, 61).

[41] Gharizadeh, B. et al. “Large-scale pyrosequencing of synthetic DNA: a comparison with results from Sanger dideoxy sequencing”. *Electrophoresis* 27(15) (2006), pp. 3042–7 (cited on page 8).

[42] Wicker, T. et al. “454 sequencing put to the test using the complex genome of barley”. *BMC Genomics* 7 (2006), p. 275 (cited on pages 8, 27).

[43] Tedersoo, L. et al. “454 Pyrosequencing and Sanger sequencing of tropical mycorrhizal fungi provide similar results but reveal substantial methodological biases”. *New Phytologist* 188(1) (2010), pp. 291–301 (cited on page 8).

[44] Zaragoza, M. V. et al. “Mitochondrial DNA variant discovery and evaluation in human Cardiomyopathies through next-generation sequencing”. *PLoS One* 5(8) (2010), e12295 (cited on page 8).

[45] Liang, B. et al. “A comparison of parallel pyrosequencing and sanger clone-based sequencing and its impact on the characterization of the genetic diversity of HIV-1”. *PLoS One* 6(10) (2011), e26745 (cited on page 8).

[46] Puritz, J. B., Addison, J. A., and Toonen, R. J. “Next-generation phylogeography: a targeted approach for multilocus sequencing of non-model organisms”. *PLoS One* 7(3) (2012), e34241 (cited on page 8).

[47] English, A. C. et al. “Mind the gap: upgrading genomes with Pacific Biosciences RS long-read sequencing technology”. *PLoS One* 7(11) (2012), e47768 (cited on pages 8, 11).

[48] “SOLiD system accuracy”. http://www3.appliedbiosystems.com/cms/groups/mcb_marketing/documents/generaldocuments/cms_057511.pdf. (accessed November 2012) (cited on page 9).

[49] Quail, M. A. et al. “A tale of three next generation sequencing platforms: comparison of Ion Torrent, Pacific Biosciences and Illumina MiSeq sequencers”. *BMC Genomics* 13 (2012), p. 341 (cited on page 9).

[50] Liu, L. et al. “Comparison of next-generation sequencing systems”. *Journal of Biomedicine and Biotechnology* 2012 (2012), p. 251364 (cited on page 9).

[51] Ng, S. B. et al. “Exome sequencing identifies the cause of a mendelian disorder”. *Nature Genetics* 42(1) (2010), pp. 30–5 (cited on page 10).

[52] Pareek, C. S., Smoczynski, R., and Tretyn, A. “Sequencing technologies and genome sequencing”. *Journal of Applied Genetics* 52(4) (2011), pp. 413–35 (cited on page 10).

[53] "1000 genomes – A deep catalog of human genetic variation". http://www.1000genomes.org. (accessed December 2012) (cited on page 10).

[54] Durbin, R.M. et al. "A map of human genome variation from population-scale sequencing". *Nature* 467(7319) (2010), pp. 1061–73 (cited on pages 10, 90).

[55] Via, M., Gignoux, C., and Burchard, E. G. "The 1000 Genomes Project: new opportunities for research and social challenges". *Genome Medicine* 2(1) (2010), p. 3 (cited on page 10).

[56] Sudmant, P. H. et al. "Diversity of human copy number variation and multicopy genes". *Science* 330(6004) (2010), pp. 641–6 (cited on page 10).

[57] Smith, D. R. et al. "Rapid whole-genome mutational profiling using next-generation sequencing technologies". *Genome Research* 18(10) (2008), pp. 1638–42 (cited on page 10).

[58] Harismendy, O. et al. "Evaluation of next generation sequencing platforms for population targeted sequencing studies". *Genome Biology* 10(3) (2009), R32 (cited on pages 10, 11, 19, 21, 56, 61, 62, 86, 99).

[59] Benaglio, P. and Rivolta, C. "Ultra high throughput sequencing in human DNA variation detection: a comparative study on the NDUFA3-PRPF31 region". *PLoS One* 5(9) (2010) (cited on page 10).

[60] Archer, J. et al. "Analysis of high-depth sequence data for studying viral diversity: a comparison of next generation sequencing platforms using Segminator II". *BMC Bioinformatics* 13 (2012), p. 47 (cited on page 10).

[61] Claesson, M. J. et al. "Comparison of two next-generation sequencing technologies for resolving highly complex microbiota composition using tandem variable 16S rRNA gene regions". *Nucleic Acids Research* 38(22) (2010), e200 (cited on page 10).

[62] Luo, C. et al. "Direct comparisons of Illumina vs. Roche 454 sequencing technologies on the same microbial community DNA sample". *PLoS One* 7(2) (2012), e30087 (cited on page 10).

[63] Wall, P. K. et al. "Comparison of next generation sequencing technologies for transcriptome characterization". *BMC Genomics* 10 (2009), p. 347 (cited on page 10).

[64] Carneiro, M. O. et al. "Pacific biosciences sequencing technology for genotyping and variation discovery in human data". *BMC Genomics* 13 (2012), p. 375 (cited on page 11).

[65] Bashir, A. et al. “A hybrid approach for the automated finishing of bacterial genomes”. *Nature Biotechnology* 30(7) (2012), pp. 701–7 (cited on pages 11, 23).

[66] Koren, S. et al. “Hybrid error correction and de novo assembly of single-molecule sequencing reads”. *Nature Biotechnology* 30(7) (2012), pp. 693–700 (cited on page 11).

[67] “NCBI – Trace Archive”. http://www.ncbi.nlm.nih.gov/Traces/trace.cgi. (accessed December 2012) (cited on page 11).

[68] Kosakovsky Pond, S. et al. “Windshield splatter analysis with the Galaxy metagenomic pipeline”. *Genome Research* 19(11) (2009), pp. 2144–53 (cited on page 13).

[69] “NCBI – dbSNP summary”. http://www.ncbi.nlm.nih.gov/projects/SNP/snp_summary.cgi. (accessed February 2012) (cited on page 15).

[70] “The single nucleotide polymorphism database (dbSNP) of nucleotide sequence variation”. http://www.ncbi.nlm.nih.gov/books/NBK21088/. (accessed February 2012) (cited on page 15).

[71] Sharp, A. J., Cheng, Z., and Eichler, E. E. “Structural variation of the human genome”. *Annual Review of Genomics and Human Genetics* 7 (2006), pp. 407–42 (cited on page 15).

[72] Ho, J. W. et al. “ChIP-chip versus ChIP-seq: lessons for experimental design and data analysis”. *BMC Genomics* 12 (2011), p. 134 (cited on page 15).

[73] Morozova, O. and Marra, M. A. “Applications of next-generation sequencing technologies in functional genomics”. *Genomics* 92(5) (2008), pp. 255–64 (cited on page 15).

[74] Marguerat, S., Wilhelm, B. T., and Bahler, J. “Next-generation sequencing: applications beyond genomes”. *Biochemical Society Transactions* 36(Pt 5) (2008), pp. 1091–6 (cited on page 15).

[75] Marguerat, S. and Bahler, J. “RNA-seq: from technology to biology”. *Cellular and Molecular Life Sciences* 67(4) (2010), pp. 569–79 (cited on page 15).

[76] Wang, Z., Gerstein, M., and Snyder, M. “RNA-Seq: a revolutionary tool for transcriptomics”. *Nature Reviews Genetics* 10(1) (2009), pp. 57–63 (cited on page 15).

[77] Millar, C. D. et al. “New developments in ancient genomics”. *Trends in Ecology and Evolution* 23(7) (2008), pp. 386–93 (cited on page 16).

[78] Prufer, K. et al. “Computational challenges in the analysis of ancient DNA”. *Genome Biology* 11(5) (2010), R47 (cited on pages 16, 64, 73, 77).

[79] Medvedev, P., Stanciu, M., and Brudno, M. “Computational methods for discovering structural variation with next-generation sequencing”. *Nature Methods* 6(11 Suppl) (2009), S13–20 (cited on page 16).

[80] “The 454 paired end protocol”. http://454.com//img/content/paired-end-library.gif. (accessed December 2012) (cited on page 17).

[81] Jarvie, T. and Harkins, T. “De novo assembly and genomic structural variation analysis with genome sequencer FLX 3K long-tag paired end reads”. *Biotechniques* 44(6) (2008), pp. 829–31 (cited on pages 16, 18, 23).

[82] “Norwegian Sequencing Centre FAQ”. http://www.sequencing.uio.no/services/faq. (accessed January 2013) (cited on pages 16, 20, 22, 23).

[83] Miller, J. R. et al. “Aggressive assembly of pyrosequencing reads with mates”. *Bioinformatics* 24(24) (2008), pp. 2818–2824 (cited on pages 18, 25).

[84] Fullwood, M. J. et al. “Next-generation DNA sequencing of paired-end tags (PET) for transcriptome and genome analyses”. *Genome Research* 19(4) (2009), pp. 521–32 (cited on page 18).

[85] Korbel, J. O. et al. “Paired-end mapping reveals extensive structural variation in the human genome”. *Science* 318(5849) (2007), pp. 420–6 (cited on page 18).

[86] Korbel, J. O. et al. “PEMer: a computational framework with simulation-based error models for inferring genomic structural variants from massive paired-end sequencing data”. *Genome Biology* 10(2) (2009), R23 (cited on page 18).

[87] International Human Genome Consortium. “Finishing the euchromatic sequence of the human genome”. *Nature* 431(7011) (2004), pp. 931–45 (cited on page 18).

[88] Chen, P. E. et al. “Rapid identification of genetic modifications in Bacillus anthracis using whole genome draft sequences generated by 454 pyrosequencing”. *PLoS One* 5(8) (2010), e12397 (cited on page 19).

[89] Wendl, M. C. “Occupancy modeling of coverage distribution for whole genome shotgun DNA sequencing”. *Bulletin of Mathematical Biology* 68(1) (2006), pp. 179–96 (cited on page 19).

[90] Wendl, M. C. and Wilson, R. K. “Aspects of coverage in medical DNA sequencing”. *BMC Bioinformatics* 9 (2008), p. 239 (cited on page 19).

[91] Kunin, V. et al. “A bioinformatician’s guide to metagenomics”. *Microbiology and Molecular Biology Reviews* 72(4) (2008), 557–78, Table of Contents (cited on pages 19, 20, 30, 31, 61, 71, 95).

[92] Quince, C., Curtis, T. P., and Sloan, W. T. “The rational exploration of microbial diversity”. *The ISME Journal* 2(10) (2008), pp. 997–1006 (cited on pages 19, 28, 30, 31).

[93] Stanhope, S. A. “Occupancy modeling, maximum contig size probabilities and designing metagenomics experiments”. *PLoS One* 5(7) (2010), e11652 (cited on page 19).

[94] Hooper, S. D. et al. “Estimating DNA coverage and abundance in metagenomes using a gamma approximation”. *Bioinformatics* 26(3) (2010), pp. 295–301 (cited on page 19).

[95] “I think I found a corner piece”. http://www.rednoblue.com/bob2/wp-content/uploads/2008/01/imagescornerpiece.gif. (accessed January 2013) (cited on page 21).

[96] “NCBI – Assembly Basics”. http://www.ncbi.nlm.nih.gov/projects/genome/assembly/assembly.shtml. (accessed November 2012) (cited on pages 20, 23).

[97] Johansen, S. D. et al. “Large-scale sequence analyses of Atlantic cod”. *New Biotechnology* 25(5) (2009), pp. 263–71 (cited on pages 20, 28).

[98] Treangen, T. J. and Salzberg, S. L. “Repetitive DNA and next-generation sequencing: computational challenges and solutions”. *Nature Reviews Genetics* 13(1) (2012), pp. 36–46 (cited on page 21).

[99] Parchman, T. L. et al. “Transcriptome sequencing in an ecologically important tree species: assembly, annotation, and marker discovery”. *BMC Genomics* 11 (2010), p. 180 (cited on page 21).

[100] Phillippy, A. M., Schatz, M. C., and Pop, M. “Genome assembly forensics: finding the elusive mis-assembly”. *Genome Biology* 9(3) (2008), R55 (cited on pages 22, 26).

[101] Commins, J., Toft, C., and Fares, M. A. “Computational biology methods and their application to the comparative genomics of endocellular symbiotic bacteria of insects”. *Biological Procedures Online* 11 (2009), pp. 52–78 (cited on pages 22, 24).

[102] Shizuya, H. and Kouros-Mehr, H. “The development and applications of the bacterial artificial chromosome cloning system”. *Keio Journal of Medicine* 50(1) (2001), pp. 26–30 (cited on page 23).

[103] Diguistini, S. et al. “De novo genome sequence assembly of a filamentous fungus using Sanger, 454 and Illumina sequence data”. *Genome Biology* 10(9) (2009), R94 (cited on page 23).

[104] Reinhardt, J. A. et al. “De novo assembly using low-coverage short read sequence data from the rice pathogen Pseudomonas syringae pv. oryzae”. *Genome Research* 19(2) (2009), pp. 294–305 (cited on page 23).

[105] Dalloul, R. A. et al. “Multi-platform next-generation sequencing of the domestic turkey (Meleagris gallopavo): genome assembly and analysis”. *PLoS Biology* 8(9) (2010) (cited on page 23).

[106] Salem, M. et al. “Characterization of the rainbow trout transcriptome using Sanger and 454-pyrosequencing approaches”. *BMC Genomics* 11 (2010), p. 564 (cited on page 23).

[107] Myers, E. W. et al. “A whole-genome assembly of Drosophila”. *Science* 287(5461) (2000), pp. 2196–204 (cited on page 25).

[108] Soderlund, C. et al. “PAVE: program for assembling and viewing ESTs”. *BMC Genomics* 10 (2009), p. 400 (cited on page 25).

[109] Zheng, Y. et al. “iAssembler: a package for de novo assembly of Roche-454/Sanger transcriptome sequences”. *BMC Bioinformatics* 12 (2011), p. 453 (cited on pages 25, 37).

[110] Huang, X. and Madan, A. “CAP3: A DNA sequence assembly program”. *Genome Research* 9(9) (1999), pp. 868–77 (cited on page 25).

[111] Chevreux, B., Wetter, T., and Suhai, S. “Genome Sequence Assembly Using Trace Signals and Additional Sequence Information”. *Computer Science and Biology: Proceedings of the German Conference on Bioinformatics (GCB)* 99 (1999), pp. 45–56 (cited on page 25).

[112] Chevreux, B. et al. “Using the miraEST assembler for reliable and automated mRNA transcript assembly and SNP detection in sequenced ESTs”. *Genome Research* 14(6) (2004), pp. 1147–59 (cited on page 25).

[113] “Mira-assembler”. http://sourceforge.net/apps/mediawiki/mira-assembler/index.php. (accessed December 2012) (cited on page 25).

[114] Miller, J. R., Koren, S., and Sutton, G. “Assembly algorithms for next-generation sequencing data”. *Genomics* 95(6) (2010), pp. 315–27 (cited on pages 25, 26).

[115] Compeau, P. E., Pevzner, P. A., and Tesler, G. “How to apply de Bruijn graphs to genome assembly”. *Nature Biotechnology* 29(11) (2011), pp. 987–91 (cited on page 25).

[116] Chaisson, M. J. and Pevzner, P. A. “Short read fragment assembly of bacterial genomes”. *Genome Research* 18(2) (2008), pp. 324–30 (cited on page 25).

[117] Zerbino, D. R. and Birney, E. “Velvet: algorithms for de novo short read assembly using de Bruijn graphs”. *Genome Research* 18(5) (2008), pp. 821–9 (cited on page 25).

[118] Zerbino, D. R. et al. “Pebble and rock band: heuristic resolution of repeats and scaffolding in the velvet short-read de novo assembler”. *PLoS One* 4(12) (2009), e8407 (cited on page 25).

[119] Kumar, S. and Blaxter, M. L. “Comparing de novo assemblers for 454 transcriptome data”. *BMC Genomics* 11 (2010), p. 571 (cited on page 25).

[120] Earl, D. et al. “Assemblathon 1: a competitive assessment of de novo short read assembly methods”. *Genome Research* 21(12) (2011), pp. 2224–41 (cited on pages 25, 26).

[121] Zhang, W. et al. “A practical comparison of de novo genome assembly software tools for next-generation sequencing technologies”. *PLoS One* 6(3) (2011), e17915 (cited on page 25).

[122] Mundry, M. et al. “Evaluating characteristics of de novo assembly software on 454 transcriptome data: a simulation approach”. *PLoS One* 7(2) (2012), e31410 (cited on page 25).

[123] Salzberg, S. L. et al. “GAGE: A critical evaluation of genome assemblies and assembly algorithms”. *Genome Research* 22(3) (2012), pp. 557–67 (cited on page 25).

[124] Schatz, M. C., Delcher, A. L., and Salzberg, S. L. “Assembly of large genomes using second-generation sequencing”. *Genome Research* 20(9) (2010), pp. 1165–73 (cited on page 25).

[125] Darling, A. E. et al. “Mauve assembly metrics”. *Bioinformatics* 27(19) (2011), pp. 2756–7 (cited on page 26).

[126] Haiminen, N., Feltus, F. A., and Parida, L. “Assessing pooled BAC and whole genome shotgun strategies for assembly of complex genomes”. *BMC Genomics* 12 (2011), p. 194 (cited on page 26).

[127] Narzisi, G. and Mishra, B. “Comparing de novo genome assembly: the long and short of it”. *PLoS One* 6(4) (2011), e19175 (cited on page 26).

[128] Edwards, R. A. et al. “Using pyrosequencing to shed light on deep mine microbial ecology”. *BMC Genomics* 7 (2006), p. 57 (cited on pages 27, 30).

[129] Pinard, R. et al. “Assessment of whole genome amplification-induced bias through high-throughput, massively parallel whole genome sequencing”. *BMC Genomics* 7 (2006), p. 216 (cited on page 27).

[130] Steuernagel, B. et al. “De novo 454 sequencing of barcoded BAC pools for comprehensive gene survey and genome analysis in the complex genome of barley”. *BMC Genomics* 10 (2009), p. 547 (cited on page 27).

[131] Wheeler, D. A. et al. “The complete genome of an individual by massively parallel DNA sequencing”. *Nature* 452(7189) (2008), pp. 872–6 (cited on page 27).

[132] Davidson, W. S. et al. “Sequencing the genome of the Atlantic salmon (Salmo salar)”. *Genome Biology* 11(9) (2010), p. 403 (cited on page 28).

[133] Star, B. et al. “The genome sequence of Atlantic cod reveals a unique immune system”. *Nature* 477(7363) (2011), pp. 207–10 (cited on page 28).

[134] Handelsman, J. et al. “The new science of metagenomics: Revealing the secrets of our microbial planet.” *The National Academies Press* (2007) (cited on page 29).

[135] Wooley, J. C. and Ye, Y. “Metagenomics: Facts and Artifacts, and Computational Challenges*”. *Journal of Computer Science and Technology* 25(1) (2009), pp. 71–81 (cited on pages 29, 31, 62).

[136] Huson, D. H. et al. “MEGAN analysis of metagenomic data”. *Genome Research* 17(3) (2007), pp. 377–386 (cited on pages 29, 31, 32, 78).

[137] Gilbert, J. A. et al. “Detection of large numbers of novel sequences in the metatranscriptomes of complex marine microbial communities”. *PLoS One* 3(8) (2008), e3042 (cited on page 29).

[138] Richter, D. C. et al. “MetaSim: a sequencing simulator for genomics and metagenomics”. *PLoS One* 3(10) (2008), e3373 (cited on pages 29, 31, 35, 52, 89).

[139] Tringe, S. G. and Hugenholtz, P. “A renaissance for the pioneering 16S rRNA gene”. *Current Opinion in Microbiology* 11(5) (2008), pp. 442–6 (cited on page 30).

[140] Sogin, M. L. et al. “Microbial diversity in the deep sea and the underexplored "rare biosphere"”. *Proceedings of the National Academy of Sciences of the USA* 103(32) (2006), pp. 12115–20 (cited on pages 30, 31, 80, 81).

[141] Angly, F. E. et al. “The marine viromes of four oceanic regions”. *PLoS Biology* 4(11) (2006), e368 (cited on page 30).

[142] Huber, J. A. et al. “Microbial population structures in the deep marine biosphere”. *Science* 318(5847) (2007), pp. 97–100 (cited on page 30).

[143] Huse, S. M. et al. “Accuracy and quality of massively parallel DNA pyrosequencing”. *Genome Biology* 8(7) (2007), R143 (cited on pages 30, 44, 48, 51, 56–59, 67, 68, 70, 78–81).

[144] Mavromatis, K. et al. “Use of simulated data sets to evaluate the fidelity of metagenomic processing methods”. *Nature Methods* 4(6) (2007), pp. 495–500 (cited on page 31).

[145] Wooley, J. C., Godzik, A., and Friedberg, I. “A primer on metagenomics”. *PLoS Computational Biology* 6(2) (2010), e1000667 (cited on page 31).

[146] Schloss, P. D. and Handelsman, J. “Introducing DOTUR, a computer program for defining operational taxonomic units and estimating species richness”. *Applied and Environmental Microbiology* 71(3) (2005), pp. 1501–6 (cited on page 31).

[147] Huson, D. H. et al. “Methods for comparative metagenomics”. *BMC Bioinformatics* 10 Suppl 1 (2009), S12 (cited on page 31).

[148] Huson, D. H. et al. “Integrative analysis of environmental sequences using MEGAN4”. *Genome Research* 21(9) (2011), pp. 1552–60 (cited on page 31).

[149] Dinsdale, E. A. et al. “Functional metagenomic profiling of nine biomes”. *Nature* 452(7187) (2008), pp. 629–32 (cited on page 32).

[150] Johnson, P. L. and Slatkin, M. “Accounting for bias from sequencing error in population genetic estimates”. *Molecular Biology and Evolution* 25(1) (2008), pp. 199–206 (cited on page 32).

[151] Hoff, K. J. “The effect of sequencing errors on metagenomic gene prediction”. *BMC Genomics* 10 (2009), p. 520 (cited on page 32).

[152] Nyren, P. “The history of pyrosequencing”. *Methods in Molecular Biology* 373 (2007), pp. 1–14 (cited on pages 33, 34).

[153] Ronaghi, M. “Pyrosequencing sheds light on DNA sequencing”. *Genome Research* 11(1) (2001), pp. 3–11 (cited on pages 34, 35, 41, 68).

[154] Melamede, R. J. “Automatable process for sequencing nucleotide”. *US Patent 4863849* (1985) (cited on page 34).

[155] Hyman, E. D. “A new method of sequencing DNA”. *Analytical Biochemistry* 174(2) (1988), pp. 423–36 (cited on page 34).

[156] Stahl, S. et al. “Solid phase DNA sequencing using the biotin-avidin system”. *Nucleic Acids Research* 16(7) (1988), pp. 3025–38 (cited on page 34).

[157] Nyren, P., Pettersson, B., and Uhlen, M. “Solid phase DNA minisequencing by an enzymatic luminometric inorganic pyrophosphate detection assay”. *Analytical Biochemistry* 208(1) (1993), pp. 171–5 (cited on page 34).

[158] Ronaghi, M. et al. “Real-time DNA sequencing using detection of pyrophosphate release”. *Analytical Biochemistry* 242(1) (1996), pp. 84–9 (cited on page 34).

[159] Mashayekhi, F. and Ronaghi, M. “Analysis of read length limiting factors in Pyrosequencing chemistry”. *Analytical Biochemistry* 363(2) (2007), pp. 275–87 (cited on page 35).

[160] Fuller, C. W. et al. “The challenges of sequencing by synthesis”. *Nature Biotechnology* 27(11) (2009), pp. 1013–23 (cited on page 35).

[161] Ronaghi, M., Uhlen, M., and Nyren, P. “A sequencing method based on real-time pyrophosphate”. *Science* 281(5375) (1998), pp. 363, 365 (cited on page 35).

[162] Quince, C. et al. “Removing noise from pyrosequenced amplicons”. *BMC Bioinformatics* 12(1) (2011), p. 38 (cited on pages 35, 58, 62, 63, 72, 77, 79, 82, 83, 96).

[163] Quail, M. A. “DNA: Mechanical Breakage”. *eLS* (cited on page 37).

[164] White, R. A. et al. “Digital PCR provides sensitive and absolute calibration for high throughput sequencing”. *BMC Genomics* 10 (2009), p. 116 (cited on page 37).

[165] Wiley, G. et al. “Methods for generating shotgun and mixed shotgun/paired-end libraries for the 454 DNA sequencer”. *Current Protocols in Human Genetics* Chapter 18 (2009), Unit18 1 (cited on page 37).

[166] Lennon, N. J. et al. “A scalable, fully automated process for construction of sequence-ready barcoded libraries for 454”. *Genome Biology* 11(2) (2010), R15 (cited on page 37).

[167] Meyer, M. et al. “Targeted high-throughput sequencing of tagged nucleic acid samples”. *Nucleic Acids Research* 35(15) (2007), e97 (cited on pages 37, 53).

[168] Meyer, F. et al. “The metagenomics RAST server - a public resource for the automatic phylogenetic and functional analysis of metagenomes”. *BMC Bioinformatics* 9 (2008), p. 386 (cited on pages 37, 53, 72).

[169] Tawfik, D. S. and Griffiths, A. D. “Man-made cell-like compartments for molecular evolution”. *Nature Biotechnology* 16(7) (1998), pp. 652–6 (cited on page 38).

[170] Williams, R. et al. “Amplification of complex gene libraries by emulsion PCR”. *Nature Methods* 3(7) (2006), pp. 545–50 (cited on page 38).

[171] Diehl, F. et al. “BEAMing: single-molecule PCR on microparticles in water-in-oil emulsions”. *Nature Methods* 3(7) (2006), pp. 551–9 (cited on page 38).

[172] Tiemann-Boege, I. et al. “Product length, dye choice, and detection chemistry in the bead-emulsion amplification of millions of single DNA molecules in parallel”. *Analytical Chemistry* 81(14) (2009), pp. 5770–6 (cited on page 38).

[173] Leamon, J. H. et al. “A massively parallel PicoTiterPlate based platform for discrete picoliter-scale polymerase chain reactions”. *Electrophoresis* 24(21) (2003), pp. 3769–77 (cited on page 39).

[174] Roche Diagnostics. “Genome Sequencer Data Analysis Software Manual”, Software Version 2.0.00. (2008) (cited on pages 44, 49, 50, 68, 70, 89).

[175] Niu, B. et al. “Artificial and natural duplicates in pyrosequencing reads of metagenomic data”. *BMC Bioinformatics* 11 (2010), p. 187 (cited on pages 44, 57, 64, 66, 72).

[176] Kunin, V. et al. “Wrinkles in the rare biosphere: pyrosequencing errors can lead to artificial inflation of diversity estimates”. *Environmental Microbiology* 12(1) (2009), pp. 118–23 (cited on pages 46, 79–81).

[177] Brockman, W. et al. “Quality scores and SNP detection in sequencing-by-synthesis systems”. *Genome Research* 18(5) (2008), pp. 763–770 (cited on pages 47–50, 57, 63, 68, 69, 73, 85).

[178] Chou, H. H. and Holmes, M. H. “DNA sequence quality trimming and vector removal”. *Bioinformatics* 17(12) (2001), pp. 1093–104 (cited on pages 47, 55, 69, 71, 80, 81).

[179] Kong, Y. “Btrim: a fast, lightweight adapter and quality trimming program for next-generation sequencing technologies”. *Genomics* 98(2) (2011), pp. 152–3 (cited on pages 47, 71).

[180] Ewing, B. et al. “Base-calling of automated sequencer traces using phred. I. Accuracy assessment”. *Genome Research* 8(3) (1998), pp. 175–85 (cited on page 47).

[181] Ewing, B. and Green, P. “Base-calling of automated sequencer traces using phred. II. Error probabilities”. *Genome Research* 8(3) (1998), pp. 186–94 (cited on page 47).

[182] Quinlan, A. R. et al. “Pyrobayes: an improved base caller for SNP discovery in pyrosequences”. *Nature Methods* 5(2) (2008), pp. 179–181 (cited on pages 47, 57, 74, 77, 86).

[183] Pearson, W. R. and Lipman, D. J. “Improved tools for biological sequence comparison”. *Proceedings of the National Academy of Sciences of the USA* 85(8) (1988), pp. 2444–8 (cited on page 50).

[184] Cock, P. J. et al. “The Sanger FASTQ file format for sequences with quality scores, and the Solexa/Illumina FASTQ variants”. *Nucleic Acids Research* 38(6) (2010), pp. 1767–71 (cited on page 50).

[185] Lysholm, F., Andersson, B., and Persson, B. “FAAST: Flow-space Assisted Alignment Search Tool”. *BMC Bioinformatics* 12 (2011), p. 293 (cited on pages 50, 77).

[186] Malde, K. “Flower: extracting information from pyrosequencing data”. *Bioinformatics* 27(7) (2011), pp. 1041–2 (cited on page 50).

[187] “Sff_extract”. http://bioinf.comav.upv.es/sff_extract/index.html. (accessed January 2013) (cited on page 51).

[188] Quince, C. et al. “Accurate determination of microbial diversity from 454 pyrosequencing data”. *Nature Methods* 6(9) (2009), pp. 639–41 (cited on pages 51, 62, 74, 77–79, 81, 96).

[189] Gilles, A. et al. “Accuracy and quality assessment of 454 GS-FLX Titanium pyrosequencing”. *BMC Genomics* 12 (2011), p. 245 (cited on pages 51, 57–59, 69, 70).

[190] McElroy, K. E., Luciani, F., and Thomas, T. “GemSIM: general, error-model based simulator of next-generation sequencing data”. *BMC Genomics* 13 (2012), p. 74 (cited on pages 51, 57, 68, 69, 88, 90).

[191] Whiteford, N. et al. “An analysis of the feasibility of short read sequencing”. *Nucleic Acids Research* 33(19) (2005), e171 (cited on page 52).

[192] “454 Life Sciences unveils new bench top sequencer”. http://www.roche.com/media/media_releases/med_dia_2009-11-19.htm. (accessed January 2013) (cited on page 53).

[193] Ueno, S. et al. “Bioinformatic analysis of ESTs collected by Sanger and pyrosequencing methods for a keystone forest tree species: oak”. *BMC Genomics* 11 (2010), p. 650 (cited on pages 57, 94).

[194] “454 customers report mixed results on FLX+”. http://www.genomeweb.com/sequencing/454-customers-report-mixed-results-flx-short-reads-and-low-throughput-top-proble. (accessed February 2013) (cited on page 57).

[195] Fischer, W. et al. “Transmission of single HIV-1 genomes and dynamics of early immune escape revealed by ultra-deep sequencing”. *PLoS One* 5(8) (2010), e12303 (cited on pages 57, 70, 76, 99).

[196] Wang, C. et al. “Characterization of mutation spectra with ultra-deep pyrosequencing: application to HIV-1 drug resistance”. *Genome Research* 17(8) (2007), pp. 1195–201 (cited on pages 57, 64).

[197] Campbell, P. J. et al. “Subclonal phylogenetic structures in cancer revealed by ultra-deep sequencing”. *Proceedings of the National Academy of Sciences of the USA* 105(35) (2008), pp. 13081–6 (cited on pages 57, 58, 63, 68, 98).

[198] De Grassi, A. et al. "Ultradeep sequencing of a human ultraconserved region reveals somatic and constitutional genomic instability". *PLoS Biology* 8(1) (2010), e1000275 (cited on page 57).

[199] Prabakaran, P. et al. "454 antibody sequencing - error characterization and correction". *BMC Research Notes* 4 (2011), p. 404 (cited on pages 57, 58).

[200] De Schrijver, J. M. et al. "Analysing 454 amplicon resequencing experiments using the modular and database oriented Variant Identification Pipeline". *BMC Bioinformatics* 11 (2010), p. 269 (cited on page 57).

[201] Huse, S. M. et al. "Ironing out the wrinkles in the rare biosphere through improved OTU clustering". *Environmental Microbiology* 12(7) (2010), pp. 1889–98 (cited on pages 58, 62, 78, 79, 81, 82, 96).

[202] Moore, M. J. et al. "Rapid and accurate pyrosequencing of angiosperm plastid genomes". *BMC Plant Biology* 6 (2006), p. 17 (cited on page 58).

[203] Stecher, B. et al. "Like will to like: abundances of closely related species can predict susceptibility to intestinal colonization by pathogenic and commensal bacteria". *PLoS Pathogens* 6(1) (2010), e1000711 (cited on pages 59, 61, 63).

[204] Falgueras, J. et al. "SeqTrim: a high-throughput pipeline for pre-processing any type of sequence read". *BMC Bioinformatics* 11 (2010), p. 38 (cited on pages 61, 76).

[205] Suzuki, M. T. and Giovannoni, S. J. "Bias caused by template annealing in the amplification of mixtures of 16S rRNA genes by PCR". *Applied and Environmental Microbiology* 62(2) (1996), pp. 625–30 (cited on page 61).

[206] Wintzingerode, F. von, Gobel, U. B., and Stackebrandt, E. "Determination of microbial diversity in environmental samples: pitfalls of PCR-based rRNA analysis". *FEMS Microbiology Reviews* 21(3) (1997), pp. 213–29 (cited on page 61).

[207] Polz, M. F. and Cavanaugh, C. M. "Bias in template-to-product ratios in multitemplate PCR". *Applied and Environmental Microbiology* 64(10) (1998), pp. 3724–30 (cited on pages 61, 62).

[208] Garber, M. et al. "Closing gaps in the human genome using sequencing by synthesis". *Genome Biology* 10(6) (2009), R60 (cited on page 62).

[209] Schwientek, P. et al. "Sequencing of high G+C microbial genomes using the ultrafast pyrosequencing technology". *Journal of Biotechnology* 155(1) (2011), pp. 68–77 (cited on page 62).

[210] Edgar, R. C. et al. “UCHIME improves sensitivity and speed of chimera detection”. *Bioinformatics* 27(16) (2011), pp. 2194–200 (cited on pages 62, 63, 71).

[211] Haas, B. J. et al. “Chimeric 16S rRNA sequence formation and detection in Sanger and 454-pyrosequenced PCR amplicons”. *Genome Research* 21(3) (2011), pp. 494–504 (cited on pages 62, 63, 71, 79).

[212] Kanagawa, T. “Bias and artifacts in multitemplate polymerase chain reactions (PCR)”. *Journal of Bioscience and Bioengineering* 96(4) (2003), pp. 317–23 (cited on pages 62, 63).

[213] Wang, G. C. and Wang, Y. “The frequency of chimeric molecules as a consequence of PCR co-amplification of 16S rRNA genes from different bacterial species”. *Microbiology* 142 (Pt 5) (1996), pp. 1107–14 (cited on page 62).

[214] Tindall, K. R. and Kunkel, T. A. “Fidelity of DNA synthesis by the Thermus aquaticus DNA polymerase”. *Biochemistry* 27(16) (1988), pp. 6008–13 (cited on pages 63, 99).

[215] Dunning, A. M., Talmud, P., and Humphries, S. E. “Errors in the polymerase chain reaction”. *Nucleic Acids Research* 16(21) (1988), p. 10393 (cited on page 63).

[216] Ennis, P. D. et al. “Rapid cloning of HLA-A,B cDNA by using the polymerase chain reaction: frequency and nature of errors produced in amplification”. *Proceedings of the National Academy of Sciences of the USA* 87(7) (1990), pp. 2833–7 (cited on page 63).

[217] Eckert, K. A. and Kunkel, T. A. “DNA polymerase fidelity and the polymerase chain reaction”. *PCR Methods and Applications* 1(1) (1991), pp. 17–24 (cited on page 63).

[218] Barnes, W. M. “The fidelity of Taq polymerase catalyzing PCR is improved by an N-terminal deletion”. *Gene* 112(1) (1992), pp. 29–35 (cited on page 63).

[219] Sun, F. “The polymerase chain reaction and branching processes”. *Journal of Computational Biology* 2(1) (1995), pp. 63–86 (cited on page 63).

[220] Weiss, G. and Haeseler, A. von. “Modeling the polymerase chain reaction”. *Journal of Computational Biology* 2(1) (1995), pp. 49–61 (cited on page 63).

[221] Hite, J. M., Eckert, K. A., and Cheng, K. C. “Factors affecting fidelity of DNA synthesis during PCR amplification of d(C-A)n.d(G-T)n microsatellite repeats”. *Nucleic Acids Research* 24(12) (1996), pp. 2429–34 (cited on page 63).

[222] Bracho, M. A., Moya, A., and Barrio, E. "Contribution of Taq polymerase-induced errors to the estimation of RNA virus diversity". *Journal of General Virology* 79 (Pt 12) (1998), pp. 2921–8 (cited on page 63).

[223] Sharifian, H. "Errors induced during PCR amplication". Master Thesis. http://e-collection.library.ethz.ch/eserv/eth:1397/eth-1397-01.pdf. (accessed February 2013). 2010 (cited on page 63).

[224] Vandenbroucke, I. et al. "Minor variant detection in amplicons using 454 massive parallel pyrosequencing: experiences and considerations for successful applications". *Biotechniques* 51(3) (2011), pp. 167–77 (cited on page 63).

[225] Zagordi, O. et al. "Error correction of next-generation sequencing data and reliable estimation of HIV quasispecies". *Nucleic Acids Research* 38(21) (2010), pp. 7400–9 (cited on page 63).

[226] Gomez-Alvarez, V., Teal, T. K., and Schmidt, T. M. "Systematic artifacts in metagenomes from complex microbial communities". *The ISME Journal* 3(11) (2009), pp. 1314–7 (cited on pages 64, 66, 72).

[227] Desgagne-Penix, I. et al. "Integration of deep transcriptome and proteome analyses reveals the components of alkaloid metabolism in opium poppy cell cultures". *BMC Plant Biology* 10 (2010), p. 252 (cited on page 64).

[228] Dong, H. et al. "Artificial duplicate reads in sequencing data of 454 Genome Sequencer FLX System". *Acta Biochimica et Biophysica Sinica (Shanghai)* 43(6) (2011), pp. 496–500 (cited on pages 64–66, 73).

[229] Briggs, A. W. et al. "Patterns of damage in genomic DNA sequences from a Neandertal". *Proceedings of the National Academy of Sciences of the USA* 104(37) (2007), pp. 14616–21 (cited on page 66).

[230] Teal, T. K. and Schmidt, T. M. "Identifying and removing artificial replicates from 454 pyrosequencing data". *Cold Spring Harbor Protocols* 2010(4) (2010), pdb prot5409 (cited on pages 66, 72).

[231] Beuf, K. D. et al. "Improved base-calling and quality scores for 454 sequencing based on a Hurdle Poisson model". *BMC Bioinformatics* 13 (2012), p. 303 (cited on pages 67, 74–77).

[232] "Phred, phrap, and consed". http://www.phrap.org/phredphrapconsed.html. (accessed December 2012) (cited on page 71).

[233] Martin, M. "Cutadapt removes adapter sequences from high-throughput sequencing reads". *EMBnet.journal* 17(1) (2011), pp. 10–12 (cited on page 71).

[234] Altschul, S. F. et al. "Basic local alignment search tool". *Journal of Molecular Biology* 215(3) (1990), pp. 403–10 (cited on page 71).

[235] Huse, S. M. et al. “Exploring microbial diversity and taxonomy using SSU rRNA hypervariable tag sequencing”. *PLoS Genetics* 4(11) (2008), e1000255 (cited on page 71).

[236] Huber, T., Faulkner, G., and Hugenholtz, P. “Bellerophon: a program to detect chimeric sequences in multiple sequence alignments”. *Bioinformatics* 20(14) (2004), pp. 2317–9 (cited on page 71).

[237] Schmieder, R. et al. “TagCleaner: Identification and removal of tag sequences from genomic and metagenomic datasets”. *BMC Bioinformatics* 11 (2010), p. 341 (cited on page 72).

[238] Li, W., Jaroszewski, L., and Godzik, A. “Clustering of highly homologous sequences to reduce the size of large protein databases”. *Bioinformatics* 17(3) (2001), pp. 282–3 (cited on page 72).

[239] Li, W., Jaroszewski, L., and Godzik, A. “Tolerating some redundancy significantly speeds up clustering of large protein databases”. *Bioinformatics* 18(1) (2002), pp. 77–82 (cited on page 72).

[240] Li, W. and Godzik, A. “Cd-hit: a fast program for clustering and comparing large sets of protein or nucleotide sequences”. *Bioinformatics* 22(13) (2006), pp. 1658–9 (cited on page 72).

[241] Huang, Y. et al. “CD-HIT Suite: a web server for clustering and comparing biological sequences”. *Bioinformatics* 26(5) (2010), pp. 680–2 (cited on page 72).

[242] Fu, L. et al. “CD-HIT: accelerated for clustering the next-generation sequencing data”. *Bioinformatics* 28(23) (2012), pp. 3150–2 (cited on page 73).

[243] Mariette, J., Noirot, C., and Klopp, C. “Assessment of replicate bias in 454 pyrosequencing and a multi-purpose read-filtering tool”. *BMC Research Notes* 4 (2011), p. 149 (cited on pages 73, 76).

[244] Ledergerber, C. and Dessimoz, C. “Base-calling for next-generation sequencing platforms”. *Briefings in Bioinformatics* 12(5) (2011), pp. 489–97 (cited on page 73).

[245] “SeqClean”. http://compbio.dfci.harvard.edu/tgi/software/. (accessed December 2012) (cited on page 76).

[246] Zhang, Z. et al. “A greedy algorithm for aligning DNA sequences”. *Journal of Computational Biology* 7(1-2) (2000), pp. 203–14 (cited on page 76).

[247] Vacic, V. et al. “A probabilistic method for small RNA flowgram matching”. *Pacific Symposium on Biocomputing* (2008), pp. 75–86 (cited on page 77).

[248] Reeder, J. and Knight, R. “Rapidly denoising pyrosequencing amplicon reads by exploiting rank-abundance distributions”. *Nature Methods* 7(9) (2010), pp. 668–9 (cited on pages 77, 82).

[249] Caporaso, J. G. et al. “QIIME allows analysis of high-throughput community sequencing data”. *Nature Methods* 7(5) (2010), pp. 335–6 (cited on page 77).

[250] Reeder, J. and Knight, R. “The ’rare biosphere’: a reality check”. *Nature Methods* 6(9) (2009), pp. 636–7 (cited on pages 78, 79, 81, 82).

[251] White, J. R. et al. “Alignment and clustering of phylogenetic markers–implications for microbial diversity studies”. *BMC Bioinformatics* 11 (2010), p. 152 (cited on pages 79, 81, 96).

[252] Gobet, A., Quince, C., and Ramette, A. “Multivariate Cutoff Level Analysis (MultiCoLA) of large community data sets”. *Nucleic Acids Research* 38(15) (2010), e155 (cited on page 80).

[253] Ashelford, K. E. et al. “New screening software shows that most recent large 16S rRNA gene clone libraries contain chimeras”. *Applied and Environmental Microbiology* 72(9) (2006), pp. 5734–41 (cited on page 82).

[254] Nakamura, K. et al. “Sequence-specific error profile of Illumina sequencers”. *Nucleic Acids Research* 39(13) (2011), e90 (cited on page 86).

[255] Dohm, J. C. et al. “Substantial biases in ultra-short read data sets from high-throughput DNA sequencing”. *Nucleic Acids Research* 36(16) (2008), e105 (cited on page 86).

[256] Angly, F. E. et al. “Grinder: a versatile amplicon and shotgun sequence simulator”. *Nucleic Acids Research* 40(12) (2012), e94 (cited on pages 88, 91).

[257] “Flowsim – A simulation pipeline for pyrosequencing data”. http://biohaskell.org/Applications/FlowSim. (accessed December 2012) (cited on page 89).

[258] Astrovskaya, I. et al. “Inferring viral quasispecies spectra from 454 pyrosequencing reads”. *BMC Bioinformatics* 12 Suppl 6 (2011), S1 (cited on page 89).

[259] Jiang, J. et al. “A cost-effective and universal strategy for complete prokaryotic genomic sequencing proposed by computer simulation”. *BMC Research Notes* 5 (2012), p. 80 (cited on page 89).

[260] Kelley, D. R. et al. “Gene prediction with Glimmer for metagenomic sequences augmented by classification and clustering”. *Nucleic Acids Research* 40(1) (2012), e9 (cited on page 89).

[261] Krause, L. et al. “Phylogenetic classification of short environmental DNA fragments”. *Nucleic Acids Research* 36(7) (2008), pp. 2230–9 (cited on page 89).

[262] Li, W. “Analysis and comparison of very large metagenomes with fast clustering and functional annotation”. *BMC Bioinformatics* 10 (2009), p. 359 (cited on page 89).

[263] Li, H. et al. “The Sequence Alignment/Map format and SAMtools”. *Bioinformatics* 25(16) (2009), pp. 2078–9 (cited on page 90).

[264] Holtgrewe, M. “Mason - a read simulator for second generation sequencing data”. *Technical Report TR-B-10-06, Institut für Mathematik und Informatik, Freie Universität Berlin.* http://publications.mi.fu-berlin.de/962/2/mason201009.pdf. 2010 (cited on page 90).

[265] Huang, W. et al. “ART: a next-generation sequencing read simulator”. *Bioinformatics* 28(4) (2012), pp. 593–4 (cited on page 90).

[266] Lysholm, F., Andersson, B., and Persson, B. “An efficient simulator of 454 data using configurable statistical models”. *BMC Research Notes* 4(1) (2011), p. 449 (cited on page 90).

[267] Li, M. et al. “Detecting heteroplasmy from high-throughput sequencing of complete human mitochondrial DNA genomes”. *American Journal of Human Genetics* 87(2) (2010), pp. 237–49 (cited on page 94).

[268] Gilbert, J. A. et al. “The taxonomic and functional diversity of microbes at a temperate coastal site: a ’multi-omic’ study of seasonal and diel temporal variation”. *PLoS One* 5(11) (2010), e15545 (cited on page 95).

[269] Bakker, M. G. et al. “Implications of pyrosequencing error correction for biological data interpretation”. *PLoS One* 7(8) (2012), e44357 (cited on pages 98, 99).

Part II

Contributions

I

BIOINFORMATICS

Vol. 26 ECCB 2010, pages i420–i425
doi:10.1093/bioinformatics/btq365

Characteristics of 454 pyrosequencing data—enabling realistic simulation with flowsim

Susanne Balzer[1,2,*], Ketil Malde[1,*], Anders Lanzén[3,4], Animesh Sharma[2] and Inge Jonassen[2,3]

[1]Institute of Marine Research, PO Box 1870, N-5817, [2]Department of Informatics, University of Bergen, PO Box 7803, N-5020, [3]Computational Biology Unit, Bergen Center for Computational Science, Thormøhlensgate 55, N-5008 and [4]Department of Biology, University of Bergen, PO Box 7803, N-5020, Bergen

ABSTRACT

Motivation: The commercial launch of 454 pyrosequencing in 2005 was a milestone in genome sequencing in terms of performance and cost. Throughout the three available releases, average read lengths have increased to ~500 base pairs and are thus approaching read lengths obtained from traditional Sanger sequencing. Study design of sequencing projects would benefit from being able to simulate experiments.

Results: We explore 454 raw data to investigate its characteristics and derive empirical distributions for the flow values generated by pyrosequencing. Based on our findings, we implement Flowsim, a simulator that generates realistic pyrosequencing data files of arbitrary size from a given set of input DNA sequences. We finally use our simulator to examine the impact of sequence lengths on the results of concrete whole-genome assemblies, and we suggest its use in planning of sequencing projects, benchmarking of assembly methods and other fields.

Availability: Flowsim is freely available under the General Public License from http://blog.malde.org/index.php/flowsim/

Contact: susanne.balzer@imr.no; ketil.malde@imr.no

1 INTRODUCTION

During the last few years novel sequencing technologies have been introduced. The platforms that are currently commercially available are marketed by Roche (454), Illumina (Solexa/Genome Analyzer), and Applied Biosystems (SOLiD), and they give new challenges for bioinformatics due to data volumes, short read lengths, and difference in errors and quality compared to traditional Sanger sequencing. So far, most bioinformatics methods available have been developed for Sanger sequencing data.

In this article, we characterize the data produced by the 454 system and in particular by its latest version named GS FLX Titanium (referred to as Titanium in the rest of the article). We analyze Titanium data sets from genomes for which the sequence has been determined. Specifically, we map each Titanium read to the reference and derive empirical distributions for the flowgram data obtained (see below; Table 1). This provides an improved basis for analysis and algorithm design, e.g. for base calling and alignment. In this article, we present a simulator that generates realistic flowgram data for any chosen DNA sequence.

The article is structured as follows: in the rest of Section 1, we briefly summarize pyrosequencing, specialized methods for analyzing pyrosequencing data (operating in 'flowspace', see Section 1.2), and simulations. Section 2 follows the results obtained from characterizing pyrosequencing data at the flow level, and in Section 3, we present the Flowsim simulator and some results obtained from comparing simulated and real data sets. Finally, in Section 4 a discussion is given.

1.1 Pyrosequencing

The 454 pyrosequencing technology is based on sequencing-by-synthesis and consists in the cyclic flowing of nucleotide reagents (repeatedly flowing T, A, C, G) over a PicoTiterPlate™. The plate consists of approximately one million wells, and each well contains at most one bead carrying a copy of a unique single-stranded DNA fragment to be sequenced. When the flowed nucleotide is complementary to the template strand in a well, the existing DNA strand in this well is extended with additional nucleotide(s) by a polymerase. This hybridization results in a reaction that generates an observable light signal which is recorded by a camera. The light intensity is converted into a 'flow value', a two-decimal non-negative number that is proportional to the length of a homopolymer run, i.e. it designates the number of nucleotides included in the flow, estimated by simply rounding the number to the closest integer (Margulies *et al*., 2005).

The term 'noise flow values' (in literature sometimes referred to as 'negative flow values', in practical terms being between 0 and 0.49) means that the light signal—although existing—is weak and judged not to result from a chemical reaction. A 'positive flow value' thus indicates incorporation of at least one base, and the number of bases (the homopolymer length) is determined from the flow value. Flow values for one bead (one read) can be used to plot a flowgram (Fig. 1a) from which the associated sequence can be determined.

The cyclically flowed nucleotides and the corresponding flow values build the basis for not only base calling, but also per-base quality score calculation (integrated in Titanium output). Obviously, the key to a correct base calling lies in the accuracy of the light signals. The 454 methodology differs from traditional Sanger sequencing in that substitution errors are a lot less frequent than insertions or deletions. Data properties have slightly changed over the three 454 generations (Roche Applied Science, 2008). We focus on the Titanium technology for all further calculations.

1.2 Use of flow values in data analysis

Although 454 sequences can be analyzed as Fasta files with standard bioinformatics tools, the flow values contain information that is not available in the pure nucleotide sequence. Consequently, several

*To whom correspondence should be addressed.

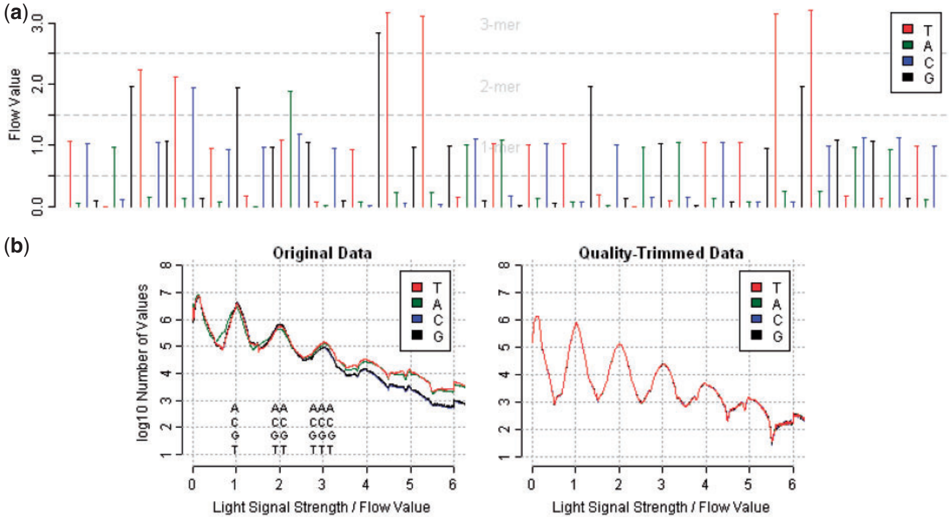

Fig. 1. (**a**) A 454 flowgram: cyclic flowing during one read. The light signal strengths (flow values) are directly translated into homopolymer runs. (**b**) Absolute frequencies of flow values (*E.coli*). Left: original data, no quality-trimming; right: quality-trimmed. The trimming algorithm enhances the separation of the homopolymer length distributions and levels out discrepancies between the nucleotides such that the curves for the four nucleotides are nearly identical.

groups have proposed algorithms to utilize flow values directly. This approach is referred to as operating in 'flowspace' as opposed to 'nucleotide space' and inhibits information loss. For example, the PyroNoise method (Quince *et al*., 2009) uses a maximum likelihood approach to decide whether a set of flowgrams is likely to result from one or several distinct underlying biological sequences. In an analogous manner, using Bayesian statistics, the PyroBayes method (Quinlan *et al*., 2008) determines the length of each homopolymer run as the most likely number of bases given the observed flow value. If the probability for an extra base exceeds a certain threshold, the extra base is added to the homopolymer run. This increases the number of insertion errors, but decreases the number of deletions and substitutions since it is intrinsic to 454 pyrosequencing that substitution errors can only arise from coherent over- and undercalls. This tendency to call more bases in homopolymer runs thus enables a higher SNP identification rate.

For small RNA discovery, direct mapping of flowgrams against a target genome ('FLAT', flowgram alignment tool) has been proved to be an efficient method (Vacic *et al*., 2008). It is also possible to achieve higher per-base accuracy rates in sequence assembly by building consensus sequences in flowspace from highly oversampled data (Huse *et al*., 2007; Margulies *et al*., 2005). Metagenomics is another field where the quality of 454-pyrosequenced data has received much attention (Gomez-Alvarez *et al*., 2009; Huson *et al*., 2007; Quince *et al*., 2009).

Studies have shown that there are several artifacts that heavily influence the processing of data for different purposes (Gomez-Alvarez *et al*., 2009; Huse *et al*., 2007), and especially methods that do not directly use flow values are sensitive to the characteristics of pyrosequencing data. For example, when matching 454 sequences with an indexing approach one can collapse all homopolymer subsequences to length one since pyrosequencing is likely to introduce errors in homopolymer lengths (Miller *et al*., 2008).

Especially for long homopolymers, many errors are caused by broad and overlapping signal distributions leading to ambiguous base calls, although there has also been work on improving 454 sequencing from the chemical aspect (Margulies *et al*., 2005). In addition to the correct determination of homopolymer lengths, the under- or over-calling of bases is especially critical for weak light signals (i.e. noise flow values). A flow value of 0.49 is treated as noise by the 454 base caller although it is almost as likely to originate from a single base call.

1.3 Simulating shotgun data

With Genfrag (Engle and Burks, 1994) and celsim (Myers, 1999), there have been earlier attempts to simulate shotgun read data, but, to the best of our knowledge, MetaSIM (Richter *et al*., 2008) is the only simulator that allows for generating 454 pyrosequencing data. MetaSIM targets Metagenomics. Internally, it uses parametric models for simulating flow values, but its output is Fasta files, and thus it is of limited use for applications that operate in flowspace.

2 FLOW VALUE DISTRIBUTIONS

One of the main challenges in 454 pyrosequencing is the correct determination of homopolymer lengths from flow values. The latter originate from a mixture of overlapping distributions. This is illustrated in Figures 1b and 3, where each distribution is assigned to one homopolymer length and one distribution to noise values. Incorrect homopolymer lengths lead to insertions and deletions during base calling (relative to the underlying biological sequence),

Table 1. Data basis for building the empirical distributions

SFF files	*Escherichia coli*	*Dicentrarchus labrax*	Total
Number of reads[a]	1 176 344	1 270 325	2 446 669
Average read length[a]	534.1	532.8	533.4
Number of bases[a]	92 924 311	85 822 587	178 746 898
Number of flow values[a]	142 361 278	130 621 280	272 982 558
Reference Genome	*Escherichia coli*	*Dicentrarchus labrax*	Total
Number of bases[b]	4 639 675	13 213 695	–
Empirical distributions	*Escherichia coli*	*Dicentrarchus labrax*	Total
Number of flow values in noise distributions	280 763 949	285 227 582	565 991 531
Number of flow values in homopolymer distributions[c]	314 495 947	278 127 101	592 623 048

[a]After 454 quality-trimming; [b]without *N*'s; [c]homopolymer lengths 1–5, equals to number of homopolymer runs in BLAST results.

and, when an over-call follows an under-call or vice versa, to a perceived substitution error. Therefore, if the distributions did not overlap, this would mean an error-free sequencing. An improved understanding of these distributions also improves the basis for designing algorithms that target the analysis of 454 pyrosequencing data.

2.1 Parametric versus empirical approaches

In earlier studies one has approximated flow values by normal, log-normal (Margulies *et al.*, 2005) or non-central student's *t* distributions (Quinlan *et al.*, 2008). However, for our data the fit of these distributions is not satisfying (Fig. 3). An alternative is to use non-parametric empirical distributions estimated from real Titanium data for which reference sequences are available. By mapping 454 data to the originating genome, we characterize the distributions of flow values coming from each homopolymer length.

2.2 Sequence comparisons

After having compared Titanium raw data from two different species, *Escherichia coli* and seabass (*Dicentrarchus labrax*, referred to as *E.coli* and *D.labrax,* respectively in the rest of the article), we decided to combine them—equally weighted—into one empirical distribution per homopolymer length. However, we also decided to include the four different nucleotide types in the same distributions since they appear to give rise to very similar distributions. In order to find the distribution of flow values that arises from one particular homopolymer length, we mapped Titanium flowgrams to a reference genome for the same organism, based on one Titanium plate each for an *E.coli* K-12 strain (Blattner *et al.*, 1997) and *D.labrax* (Kuhl *et al.*, 2010). We used BLAST (Altschul *et al.*, 1990) to identify the location of reads that could be aligned unambiguously to one location on the genome, with default BLAST parameters, except for gap open and extend penalties, which were set to 1.

Table 2. Parameters of the empirical distributions

Homopolymer length	Mean	Standard deviation
0	0.1230	0.0737
1	1.0193	0.1227
2	2.0006	0.1585
3	2.9934	0.2188
4	3.9962	0.3168
5	4.9550	0.3863
Linear regression for $n \geq 6$	n[a]	$0.03494 + n \cdot 0.06856$[a]

[a]Normal distribution. Mean and standard deviation of normal distribution around homopolymer lengths of 6, 7 etc.

To distinguish sequencing errors from true biological variation, we used a bit score threshold of 200 and only the best match for each sequence. Furthermore, we discarded all those matches that had a corresponding second best match with a bit score $<5\%$ worse than the best match, i.e. two matches with bit scores that were approximately equally high.

For *E.coli*, there were uncertainties in terms of which reference genome to choose, as none of the available reference genomes gave us $>97\%$ identity with the pyrosequencing data, but the match filtering mentioned above should account for these problems.

2.3 Calculation of empirical distributions

We aligned the flowgrams to the matching genomic region, assigning each flow value to the corresponding true homopolymer length as known from the reference genome. Thus, we collected the flow values assigned to each homopolymer length distribution from 0 to 5, as shown in Figure 3.

For homopolymer lengths greater than 5, our data is sparse, and it is therefore better to approximate the real distributions by extrapolating parametric distributions from the shorter homopolymer lengths. Table 2 shows the observed mean and standard deviation of the empirical distributions for homopolymer lengths 0 to 5, and the linear regression for these parameters based on normal distributions fitted to homopolymer lengths 1 to 5.

2.4 Degradation and Noise

We find our resulting empirical distributions to be almost symmetrical around the corresponding integers, with relatively low standard deviation for short homopolymer runs. However, when analyzing data from the three 454 generations, we also found that the degree of symmetry varies between them. Quinlan *et al.* (2008) report a significantly higher insertion than deletion rate, which is consistent with an asymmetry in the tails of the distributions, but we found the asymmetry to decrease towards newer generation data.

Nevertheless, we can clearly observe two kinds of degradation: since standard deviation increases for increasing homopolymer lengths, these belong to broader distributions with overlapping tails, where the latter generally means a higher risk of over- and under-calls.

Second, analysis of the flow values associated with sequence parts that have been trimmed off (during standard 454 quality-trimming) indicates that 454 quality-filtering and -trimming calibrates discrepancies between the four nucleotides and increases the separations of the distributions, involving deeper valleys

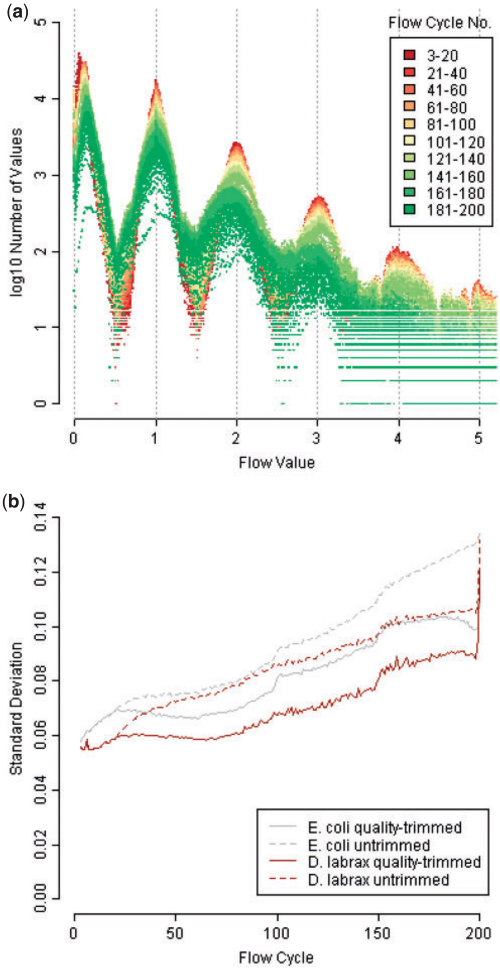

Fig. 2. (**a**) Absolute frequencies of flow values by flow cycle. A total of 200 flow cycles of a Titanium run correspond to 200 × 4 = 800 flows. The first two flow cycles contain the TCAG tag and are omitted here. Towards the end of a run, flow values tend to lie further away from their ideal values (integers), but are obviously less in number because many values from later flow cycles have been trimmed away. (**b**) Standard deviation of flow values (difference in relation to their closest integer), by flow cycle. Standard deviation increases almost linearly. Only flow values <5.5 were included.

between them (Figs 1b and 2a). We therefore use only the subsequences retained after quality-trimming to estimate the empirical distributions, thus being able to treat the nucleotides equally. Also for quality-trimmed raw data, we can see that both read and flow position of a base have a remarkable influence on the accuracy of flow values. We have observed a clear degradation in accuracy over the length of a run, i.e. when comparing earlier to later flow cycles, by measuring for each flow cycle how much the difference between a flow value and its ideal counterpart (i.e. the closest integer) varies (Fig. 2b).

2.5 Read lengths

The length of un-trimmed reads in 454 pyrosequencing is limited by either the number of flows (168 in GS20, 400 in GS FLX and 800 in GS FLX Titanium) or the length of the clones. The longest reads are thus obtained when the clone length exceeds the number of flows, such that the DNA strands in the well are extended until the very last flow cycle.

As quality decreases towards the end of a read, several filters are applied on the reads, which again gives a different read length distribution. We can thus distinguish between the distribution of clone lengths, the distribution of read lengths before filtering and quality-trimming and that after application of those filters. A detailed description of the filtering algorithms is given in the 454 manual (Roche Applied Science, 2008). As visible in Figure 1b, they eliminate (some of the) artifacts in the distributions by trimming low-quality flow values from the end of each read.

3 FLOWSIM—A SIMULATOR FOR 454 DATA

To take advantage of the empirical distributions, we implemented Flowsim, a simulator for pyrosequencing data.

3.1 Implementation of Flowsim

Given an input sequence in Fasta format, Flowsim selects substrings of this sequence with random position and strand, and generates a flowgram by converting the nucleotide sequences to sequences of homopolymer lengths. Each homopolymer length is then altered according to its flow distribution, where the latter is allowed to vary (degrade) with the flow position in the simulated read. To emulate degradation, we derived 20 different sets of empirical distributions from our mapping results (Fig. 3), where each of them represents 10 consecutive Titanium flow cycles, which sums up to 800 flow values.

The simulated flowgram is then analyzed to call nucleotide sequence and quality scores. Finally, all generated information is stored in an SFF file, similar to the ones produced by the 454 software.

One can further specify the number of desired output reads and also incorporate user-defined empirical distributions, either position-specific (degrading) or not.

3.2 Quality scores

It is crucial to assign a quality score to each called base, since sequenced bases are not filtered individually during quality-filtering and -trimming, but rather in the context of their reads. Quality scores are e.g. useful for assembly projects, although some assemblers do not use them. If they do, however, they might rely on them for incorporating Sanger reads since 454 quality scores are expressed as a phred equivalent (Margulies *et al.*, 2005; Roche Applied Science, 2008). On the other hand, scores can also be used by assemblers built for Sanger sequences when assembling 454 sequences.

Although the method for determining quality has been described both for GS20 (Margulies *et al.*, 2005) and Titanium (Brockman *et al.*, 2008), the exact parameters are not known. Instead, Flowsim

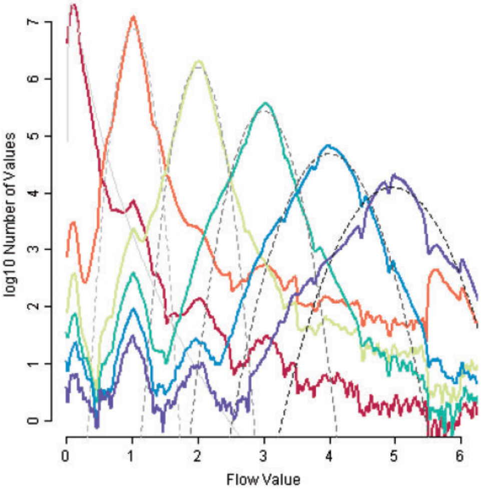

Fig. 3. Empirical distributions (smoothed average of *E.coli* and *D.labrax*) on logarithmic scale. In gray: fitted (log-) normal distributions.

calculates the error probability ('the base in question is an over-call'), using Bayes' Theorem, and transfers it into a phred equivalent. Thus, the quality score corresponds to the true quality of the simulated base call, rather than to the quality the 454 software would produce for the same flowgram.

Flowsim currently supports two quality calling methods based on Bayesian statistics. One produces decreasing quality scores for the bases in a homopolymer, similar to GS20. The second produces a series of identical values for each base in a homopolymer, as in Titanium, but otherwise builds on the same Bayesian approach as the GS20 algorithm. Compared to the quality scores assigned to Titanium by the Roche analysis pipeline, our quality scores are lower. As GS20 appears to use a fixed table mapping each flow value to a set of qualities, there is also a third option of assigning qualities from a table derived from GS20 data.

Bayes' theorem requires both the prior probability for each homopolymer length and the conditional probability for a flow value given a certain homopolymer length. In contrast to Margulies *et al.*, we use both empirical priors (from the input Fasta file) and empirical conditional probabilities (from our empirical distributions). This allows us to assess the quality of our simulated data as accurately as possible. When position-specific empirical distributions are used in Flowsim, we also use these for quality score calculation.

3.3 Simulating data sets

We used Flowsim to generate synthetic data sets, using our empirical distributions as the flow model. Each of the 20 distributions was used for 10 flow cycles (40 flows), giving a realistic degradation of quality along the sequence. We also simulated data sets using 400 flow cycles, simulating a hypothetical 454 generation with twice the read length of the current Titanium generation. The *E.coli* genome (K-12 strain, GenBank ID: 49175990) was used as the input genome.

Table 3. *De novo*-based and reference-based N50 for *E. coli*

Coverage	Real	200 cycles (simulated)	400 cycles (simulated)
De novo-based N50 for *E.coli*			
1	649	651	995
5	2406	7045	7623
10	23 613	132 913	104 012
15	67 231	173 592	178 129
20	86 902	172 127	203 060
25	95 348	176 747	207 011
30	97 821	171 819	207 011
Reference-based N50 for *E. coli*			
1	895	1093	1681
5	8305	31 730	40 321
10	76 687	207 827	2 343 849
15	110 013	207 856	2 496 857
20	118 387	207 740	2 497 013
25	161 266	207 899	2 497 058
30	177 489	207 845	2 724 990

3.4 Simulation results

We have performed both de-novo and reference-based assembly using Newbler assembler version 2.3 (Roche), approximating various coverage (1×, 5×, 10×, 15×, 20×, 25× and 30×). A simulation with 200 flow cycles shows ∼1% inferred error, while 400 flow cycles result in an error rate of ∼0.8%, which is the same as for the real data (Titanium, i.e. 200 flow cycles).

Our results indicate that Flowsim can be useful to estimate the quality of an assembly that can be expected from using Titanium to shotgun sequence a genome. However, the assemblies resulting from our simulations were consistently better in terms of contig sizes (through the N50 summarizing statistic, see Table 3) for the simulated data sets than for the real ones. This may partly be due to all simulated reads coming from the reference genome and thus avoiding strain-specific discrepancies, which leads to the fact that 100% of the reads for 200 and 400 flow cycle simulations can be mapped back to genome, while real data reach only ∼98.7% for all studied coverage values. There may also be other factors such as possible biases in terms of genome coverage in the experimental protocols used to generate the shotgun libraries for Titanium sequencing. Further work will include exploring such biases and other sources of variability as well as characterizing their influence on the simulation accuracy of Flowsim. Also Flowsim will be extended to include simulation of paired-reads, which will be of high value for simulation and planning of projects for de-novo whole-genome sequencing.

4 DISCUSSION

This study aims to sketch the opportunities that arise from analyzing pyrosequencing raw data, culminating in the use of empirical distributions. The empirical distributions give us a very realistic picture of the underlying characteristics of the light signal values that are later translated into DNA sequences. In contrast, earlier approaches to modeling flow data have built on parametric

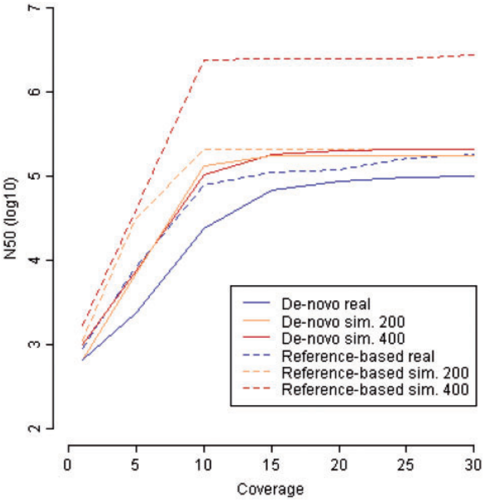

Fig. 4. *De novo* and reference-based N50 for *E.coli*. Both real and simulated 454 data were assembled using Newbler v2.3.

distributions, and the same distributions were used for whole reads, without respect to flow or read positions.

Our findings and the empirical distributions are based on large amounts of data from three different species (*E.coli*, *D.labrax*, *Gadus morhua*), four sequencing labs, both shotgun and paired-end reads with different gap sizes. The empirical flow value distributions are very similar, and we have not observed any factors which influence the shape of the distributions apart from the 454 generation. Thus, we have a good reason to believe that the distributions used in Flowsim are representative.

The flow values that result from 454 sequencing exhibit many interesting characteristics and artifacts, and we do not address them all here. Some of these are generation-specific, some of them have remained stable over the years, and some of them only appear on one certain plate, for one certain species or in one lab. One known artifact, exact or almost-exact duplicates, has been not only described for metagenomics in the literature (Gomez-Alvarez *et al*., 2009), but we also observed them in shotgun sequences for *E.coli* and *D.labrax*.

We do emulate the degradation in empirical flow distributions, and we also calculate the corresponding quality scores. In contrast, we neglect some of the artifacts that we have observed in the empirical distributions, but are not able to interpret properly yet, such as for example: shifts in peaks that lead to systematic over- or under-calls, jumps, neighboring peaks, i.e. subpeaks around the next or preceding integer. These are particularly strong for the noise distribution (with a neighboring peak around 1) and the 1-distribution (with neighboring peaks around 0.1 and 2), but the values causing these peaks are not many in number. Analyzing the corresponding data including the related alignments we found that the subpeaks are likely to be caused by real biological differences. This will be explored further in a separate study. In this context, we also performed a weak smoothing process that helped to reduce subpeaks and jumps.

Furthermore, the 454 image analysis software implements a set of quality filters that sets trimming coordinates to identify the high-quality part of each read. In addition, some reads are eliminated entirely based on quality metrics. Although these filters are documented (Roche Applied Science, 2008), the documentation is not sufficient to re-implement them, and the current version of Flowsim does not attempt to simulate them. We hope to address this in a future release (Fig. 4).

In conclusion, our simulator produces sufficiently realistic 454 files as we model all important phenomena that we have observed. Furthermore, Flowsim allows the user to specify many of its parameters, making it adaptable to new real or hypothetical 454 generations.

ACKNOWLEDGEMENTS

The authors wish to thank Dr Alexander Sczyrba, DOE Joint Genome Institute, Dr Richard Reinhardt, Max Planck Institute for Molecular Genetics, Berlin, for kindly providing us with Titanium raw data and Dr Christopher Quince, University of Glasgow, for the fruitful discussions. Notur is acknowledged for access to the Titan cluster in Oslo.

Funding: The National Program for Research in Functional Genomics in Norway (FUGE) in the Research Council of Norway.

Conflict of Interest: none declared.

REFERENCES

Altschul,S.F. *et al*. (1990) Basic local alignment search tool. *J. Mol. Biol*., **215**, 403–410.

Blattner,F.R. *et al*. (1997) The complete genome sequence of Escherichia coli K-12. *Science*, **277**, 1453–1462.

Brockman,W. *et al*. (2008) Quality scores and SNP detection in sequencing-by-synthesis systems. *Genome Res*., **18**, 763–770.

Engle,M.L. and Burks,C. (1994) GenFrag 2.1: new features for more robust fragment assembly benchmarks. *Comput. Appl. Biosci*., **10**, 567–568.

Gomez-Alvarez,V. *et al*. (2009) Systematic artifacts in metagenomes from complex microbial communities. *ISME J*., **3**, 1314–1317.

Huse,S.M. *et al*. (2007) Accuracy and quality of massively parallel DNA pyrosequencing. *Genome Biol*., **8**, R143.

Huson,D.H. *et al*. (2007) MEGAN analysis of metagenomic data. *Genome Res*., **17**, 377–386.

Kuhl,H. *et al*. (2010) The European sea bass Dicentrarchus labrax genome puzzle: comparative BAC-mapping and low coverage shotgun sequencing. *BMC Genomics*, **11**, 68.

Margulies,M. *et al*. (2005) Genome sequencing in microfabricated high-density picolitre reactors. *Nature*, **437**, 376–380.

Miller,J.R. *et al*. (2008) Aggressive assembly of pyrosequencing reads with mates. *Bioinformatics*, **24**, 2818–2824.

Myers,G. (1999) A dataset generator for whole genome shotgun sequencing. In *Proceedings of International Conference on Intelligent Systems of Molecular Biology*, pp. 202–210.

Quince,C. *et al*. (2009) Accurate determination of microbial diversity from 454 pyrosequencing data. *Nat. Methods*, **6**, 639–641.

Quinlan,A.R. *et al*. (2008) Pyrobayes: an improved base caller for SNP discovery in pyrosequences. *Nat. Methods*, **5**, 179–181.

Richter,D.C. *et al*. (2008) MetaSim: a sequencing simulator for genomics and metagenomics. *PLoS One*, **3**, e3373.

Roche Applied Science. (2008) *Genome Sequencer Data Analysis Software Manual*, Software Version 2.0.00, Roche Diagnostics GmbH.

Vacic,V. *et al*. (2008) A probabilistic method for small RNA flowgram matching. In *Pacific Symposium on Biocomputing*, pp. 75–86.

Characteristics of 454 Pyrosequencing Data – Enabling Realistic Simulation with Flowsim

Susanne Balzer, Ketil Malde, Anders Lanzén, Animesh Sharma and Inge Jonassen

The authors would like to apologize for an error in the calculation of the number of bases, number of flow values and average read length. Our reads turned out to be a lot shorter than previously reported. None of these errors has implications on the method or the results. The corrected table is shown below.

Table 1. Data basis for building the empirical distributions

SFF files	*E. coli*	*D. labrax*	Total
Number of reads*	1,176,344	1,270,325	2,446,669
Average read length*	393.7	424.0	409.4
Number of bases*	463,133,786	538,607,063	1,001,740,849
Number of flow values*	710,777,022	819,636,576	1,530,413,598
Reference Genome	*E. coli*	*D. labrax*	Total
Number of bases**	4,639,675	13,213,695	-
Empirical Distributions	*E. coli*	*D. labrax*	Total
Number of flow values in noise distributions	280,763,949	285,227,582	565,991,531
Number of flow values in homopolymer distributions***	314,495,947	278,127,101	592,623,048

*after 454 quality-trimming **without N's ***homopolymer lengths 1-5, equals to number of homopolymer runs in BLAST results

The error also affects figure 1b, where the left part of the plot is to be compared with the right set of curves. The corrected figure is shown below.

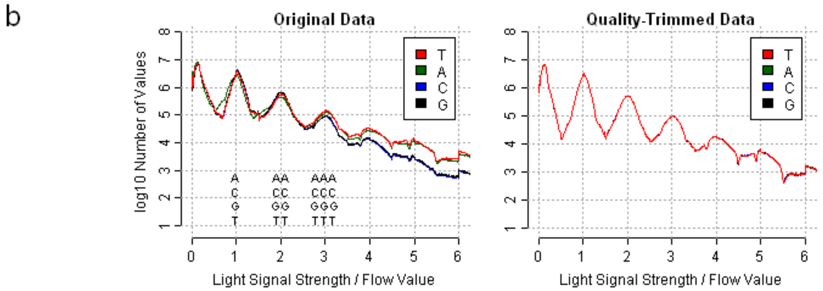

Fig. 1 (b) Absolute frequencies of flow values (*E. coli*). Left: Original data, no quality-trimming; right: quality-trimmed. The trimming algorithm enhances the separation of the homopolymer length distributions and levels out discrepancies between the nucleotides such that the curves for the four nucleotides are nearly identical.

BIOINFORMATICS

Vol. 27 ISMB 2011, pages i304–i309
doi:10.1093/bioinformatics/btr251

Systematic exploration of error sources in pyrosequencing flowgram data

Susanne Balzer[1,2,*], Ketil Malde[1] and Inge Jonassen[2,3]

[1]Institute of Marine Research, P.O. Box 1870, N-5817 Bergen, [2]Department of Informatics, University of Bergen, P.O. Box 7803, N-5020 Bergen and [3]Computational Biology Unit, Uni Computing, Thormøhlensgate 55, N-5008 Bergen, Norway

ABSTRACT

Motivation: 454 pyrosequencing, by Roche Diagnostics, has emerged as an alternative to Sanger sequencing when it comes to read lengths, performance and cost, but shows higher per-base error rates. Although there are several tools available for noise removal, targeting different application fields, data interpretation would benefit from a better understanding of the different error types.

Results: By exploring 454 raw data, we quantify to what extent different factors account for sequencing errors. In addition to the well-known homopolymer length inaccuracies, we have identified errors likely to originate from other stages of the sequencing process. We use our findings to extend the flowsim pipeline with functionalities to simulate these errors, and thus enable a more realistic simulation of 454 pyrosequencing data with flowsim.

Availability: The flowsim pipeline is freely available under the General Public License from http://biohaskell.org/Applications/FlowSim.

Contact: susanne.balzer@imr.no

1 INTRODUCTION

Second-generation sequencing techniques have revolutionized DNA sequencing. In comparison with Illumina (Solexa/Genome Analyzer) and Applied Biosystems (SOLiD), 454 pyrosequencing stands out with its longer reads (up to ~500 bp). However, higher sequencing error rates compared with traditional Sanger sequencing and the lack of a detailed understanding of error characteristics still hamper the effective utilization of pyrosequencing.

In *de novo* whole-genome sequencing, high coverage may compensate for erroneous sequences. However, erroneous reads are problematic for SNP detection (Quinlan *et al.*, 2008) and especially for metagenomics, as they can lead to a considerable overestimation of diversity in a sample (Quince *et al.*, 2009). Hence, there has been a strong focus on examining the quality of 454 pyrosequencing data and noise removal. Also artificial duplicates are an important issue, because they may lead to incorrect conclusions about the abundance of species and genes (Gomez-Alvarez *et al.*, 2009).

1.1 The 454 pyrosequencing technology

The 454 pyrosequencing technology is based on sequencing-by-synthesis which is performed in parallel on around one million beads deposited in wells on a plate. Each bead carries around 10 million molecules resulting from emulsion PCR (emPCR) starting from one single DNA fragment. The sequencing is performed by cyclic flowing (T, A, C, G) of nucleotide reagents over the plate, every bead giving rise to at most one DNA sequence ('read'). Each flow produces a light signal in each of the beads, either a very weak signal ('negative flow value', in practice being between 0 and 0.5, indicating that no base was incorporated) or a stronger signal ('positive flow value'), proportional to the length of a homopolymer run (Margulies *et al.*, 2005).

This chemical process implicates two characteristics that are intrinsic to 454 pyrosequencing data: when the light signal is too strong or too weak, this leads to an over- or under-call for the corresponding nucleotide type. For example, a flow value of 2.48 for nucleotide C gives a homopolymer length of two, while a flow value of 2.52 will give three nucleotides. Apparent substitution errors can occur when an over-call follows an under-call or vice versa. Compared with the called DNA sequence, the underlying flow values thus contain additional information relevant for base calling accuracy and for comparison of reads, which is why analyses often are carried out in 'flowspace' as opposed to 'nucleotide space'.

The latest 454 pyrosequencing version, GS FLX Titanium (referred to as Titanium in the rest of the paper), uses 200 flow cycles, which corresponds to 800 flows. The results of one sequencing run include the light signal intensity data ('flow values') for each well and the base called DNA sequence together with quality information. This is stored in a binary SFF (standard flowgram format) file.

1.2 Duplicate reads

Earlier studies have revealed that between 4–44% (Niu *et al.*, 2010) and 11–35% (Gomez-Alvarez *et al.*, 2009) of sequences in a typical metagenomic dataset are exact or almost-exact duplicates. Both tools 454 Replicate Filter (Gomez-Alvarez *et al.*, 2009) and cd-hit-454 (Niu *et al.*, 2010) are based on the CD-HIT clustering algorithm (Li and Godzik, 2006) and provide a fast way of removing duplicates from pyrosequencing data. While this is a crucial step for the success of metagenomic studies based on 454 pyrosequencing data, we have not observed a comparably high percentage of exact or almost-exact duplicates in shotgun data generated in the context of projects we are involved in.

1.3 Erroneous reads

There are several factors that account for erroneous base calls or reads, especially inaccuracies in the sequencing chemistry, leading to slightly too high or low flow values, and carry-forward and incomplete extension errors (Margulies *et al.*, 2005), accumulating over the read, which reflects the stochastic nature of the base incorporation chemistry. Furthermore, it has been shown that

[*]To whom correspondence should be addressed.

a low percentage of reads accounts for a high percentage of errors (Huse *et al.*, 2007) and that sequencing quality decreases toward the end of a read (Balzer *et al.*, 2010; Hoff, 2009). We have earlier described the characteristics of these inaccuracies, calculated the empirical distributions of flow values and included the results in our simulation tool flowsim (Balzer *et al.*, 2010). However, these models do not adequately explain all the sequencing errors that we have observed, which is reflected in the fact that, when applied to whole-genome shotgun sequencing, our simulator produces data giving better assemblies than does real data (Section 3). Here, we report on a more careful examination of other error sources and suggest a new pipeline for a more realistic simulation of 454 pyrosequencing reads. We are not able to establish the exact source of these errors, but hypothesize that a portion of the errors are introduced during PCR library preparation.

1.4 Filtering and trimming

Some of these error patterns, but not all of them, are addressed by the 454 quality-trimming and read-filtering algorithms. A detailed description is given in the 454 manual (Roche Applied Science, 2008). However, in some applications, improved results are obtained when applying a stricter quality-filtering and -trimming (compared with 454 default settings) or using additional algorithms and tools. Several research groups have suggested methods for noise removal and quality-trimming, the requirements on data quality obviously varying with respect to applications. Whole-read filtering strategies include the complete removal of: chimeric reads, reads with undetermined bases (i.e. *N*'s) or reads showing a certain percentage of flow values in the interval [0.5, 0.7] (termed 'dubious flow values') before reaching a certain flow cycle (Huse *et al.*, 2007; Kunin *et al.*, 2009; Quince *et al.*, 2011). Trimming approaches focus on: a stricter read-trimming based on quality scores, adaptor removal [e.g. with LUCY (Chou and Holmes, 2001)], but also more sophisticated approaches such as multiple assembly strategies with reads obtained by applying several trimming settings (http://www.genome.ou.edu/informatics.html).

2 FACTORS FOR SEQUENCE QUALITY

In this study, we characterize error patterns derived from Titanium 454 pyrosequencing data and estimate to what extent different error types account for sequencing errors.

2.1 Adaptors

Sequences are limited in length by the number of flow cycles. Ideally, clones should be sufficiently long so that the end of the clone is not reached during sequencing, which means that also the adaptor is not reached. If the clone is shorter, the adaptor sequence will be included at the end of the read. This part of the sequence should be masked by the Roche analysis pipeline. However, the trimming procedure sometimes fails if only part of the adaptor is contained in the read or if there are sequencing errors in the adaptor sequence. We have observed both cases in shotgun data from different genomes.

In genome assembly, residual adaptors can block contig extension at the end of reads, especially in lower coverage regions and when working with assemblers that do not use a broad overlap window.

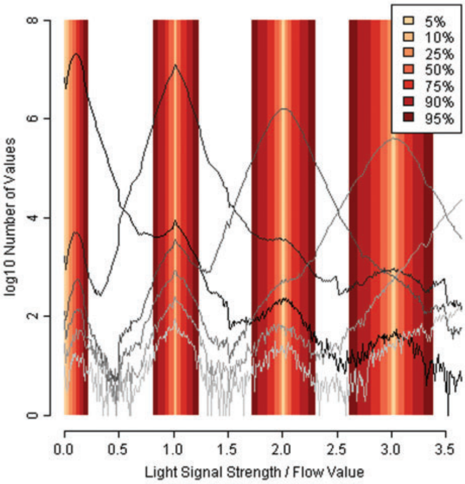

Fig. 1. Empirical flow values distributions (*D.labrax*) and derived intervals.

2.2 Pyrosequencing errors

The light signal strength from the chemical reaction in the sequencing process is the basis for correct determination of homopolymer lengths and hence responsible for data accuracy. Slightly too high or too low signal strengths can lead to over- or under-calls.

Carry-forward errors occur when the flushing between the flows is not sufficient and leftover nucleotides are present in a well. Also the incomplete extension of a template due to insufficient nucleotides within a flow can cause a read to get out-of-sync. These errors are collectively referred to as CAFIE. The Roche software adjusts the flow values in an attempt to correct for these errors, and both the flow values and the DNA data in the SFF file correspond to the corrected data (Roger Winer, Roche Diagnostics, personal communication).

2.3 Putative PCR errors

In a previous work, we derived empirical distributions from *Dicentrarchus labrax* (sea bass) Titanium data: by mapping 454 data to the originating reference genome (Kuhl *et al.*, 2010), we characterized the distributions of flow values belonging to each homopolymer length (Balzer *et al.*, 2010). These flow value distributions, one distribution per homopolymer length, overlap, causing over- and under-calls (Fig. 1). By examining them in detail, an interesting and hitherto unexplained pattern emerges: the flow value distributions often contain one major peak around the integral value representing the correct homopolymer length, but then also smaller peaks around the neighboring integral values (Figs 1 and 3). Although these neighboring peaks have been observed previously, we have not seen any convincing explanation for them. Hypothesizing that they are caused by errors in the emulsion PCR performed prior to sequencing, we make an attempt to estimate to what extent PCR errors contribute to the overall error rate.

Table 1. Flow value intervals from empirical distributions (*D.labrax*)

Size (%)	0-distribution	1-distribution	2-distribution	3-distribution
5	[0.00, 0.02]	[1.01, 1.02]	[2.00, 2.02]	[3.01, 3.03]
10	[0.00, 0.04]	[1.01, 1.03]	[2.00, 2.03]	[3.00, 3.04]
25	[0.00, 0.07]	[1.00, 1.04]	[1.97, 2.05]	[2.97, 3.07]
50	[0.00, 0.11]	[0.96, 1.07]	[1.93, 2.09]	[2.90, 3.12]
75	[0.00, 0.14]	[0.92, 1.12]	[1.86, 2.16]	[2.81, 3.20]
90	[0.00, 0.18]	[0.86, 1.18]	[1.78, 2.24]	[2.69, 3.30]
95	[0.00, 0.22]	[0.81, 1.23]	[1.72, 2.31]	[2.61, 3.39]

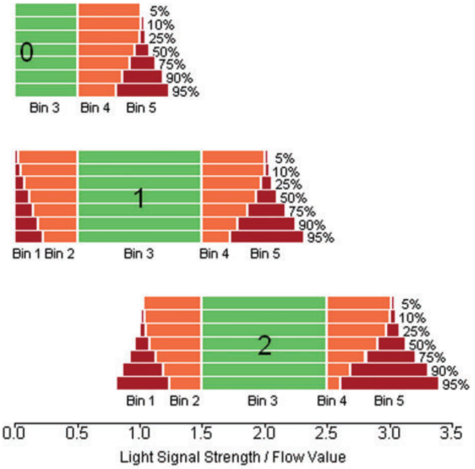

Fig. 2. Bins for homopolymer lengths 0, 1 and 2, based on different flow value interval sizes from Table 1.

Table 2. Estimated fraction of error types in percentage of overall errors

Size (%)	Pyrosequencing errors (%)	Putative PCR errors (%)	Extreme errors (%)
5	80.18	3.97	15.85
10	79.28	5.78	14.94
25	75.69	11.17	13.14
50	67.15	24.65	8.20
75	59.18	36.89	3.93
90	51.62	47.02	1.36
95	46.63	52.77	0.60

In order to quantify and compare the number of errors caused by overlapping distributions with the errors in neighboring peaks, we classified flow values according to narrow intervals around the integral values. Based on the empirical unsmoothed flow value distributions from *D.labrax* (Balzer *et al.*, 2010), the intervals were constructed so that they would contain a certain percentage (the middle part) of flow values for each homopolymer length. The intervals are slightly asymmetric (Table 1), which corresponds to earlier observations that insertion errors are more common than deletions (Huse *et al.*, 2007; Quinlan *et al.*, 2008). For flow values of the 0-distribution (assumed not to correspond to incorporation of a nucleotide, i.e. negative flow values), the interval extends to one side only.

We constructed several series of intervals, containing from 5% (conservative) to 95% (liberal) of the flow values (Table 1 and Fig. 1). In order to decompose the distribution of flow values observed for homopolymers of length n, we assigned each associated flow value to one of several bins. First, flow values that would give a correct homopolymer length call (values between $n-0.5$ and $n+0.49$) were assigned into bin 3. Then, values that were likely to be associated with a neighboring peak at $n-1$ or $n+1$ (subpeaks in Figs 1 and 3) were assigned to bins 1 and 5, respectively (using the values from Table 1 as threshold values). Intermediate values were assigned into bins 2 and 4, while values outside the ranges of bins 1 and 5 were discarded (extreme under- or over-calls).

As an example, when considering a homopolymer of length 2, we would define our bins as follows (using the rather conservative 25% intervals, see Table 1 and Fig. 2): bin 3 contains correct base calls and is thus predefined as [1.5, 2.49]. All flow values that do not fall into this bin are counted as erroneous. Of all flow values in the range [0.5, 1.49], 25% are in [1.0, 1.04]. This interval thus defines bin 1 for homopolymer length 2. Flow values in this bin are assumed to originate from the 1-distribution and are thus—by our hypothesis—likely to be caused by PCR errors. Bin 5 is accordingly defined as [2.97, 3.07] and corresponds to PCR errors giving a triple homopolymer.

Furthermore, flow values that lie beyond bin 1 or 5 are counted as extreme miscalls of unknown origin (‘extreme errors’, see Table 2).

For each flow value together with the correct homopolymer length, we can now determine into which bin it falls. From the absolute counts, we can then for any sequence or set of sequences calculate the fraction of ‘putative PCR errors’ (Table 2), which is the sum of errors falling into bins 1 and 5 divided by the total number of erroneous base calls.

We used BLASTN (Altschul *et al.*, 1990) to map 21 mate-pair runs from *Gadus morhua* (Atlantic cod) against the known mate-pair linker sequence (TCGTATAACTTCGTATAATGTATGCTATAC GAAGTTATTACG) and its reverse complement, assigning each flow value to the corresponding true homopolymer length as known from the linker sequence. This gave us a total of 17 834 274 reads, where 16 836 422 matched the linker sequence or its reverse complement (47% each) when a bit score cutoff of 67 was used. The 997 833 (6%) reads did not or not uniquely match either the linker or its reverse complement.

Further, we discarded 17% of the remaining reads because they had lost synchronism (Section 2.2) or were implausible, or did not match the linker over the whole length of 42 bp, which left us with a total of 14 050 646 complete matches.

From those reads, we examined the flow values for each of the 60 flows (15 flow cycles; 18 positions with negative flow values not leading to a base call; flows 1 and 60 were not counted in error calculations since they could be part of longer homopolymers) that were needed to sequence the 42 bp of the linker (Fig. 3). We assigned each flow value to one of the bins described above. From the total number of errors in each bin, we could calculate the percentage in relation to all observed errors (Table 2). In total,

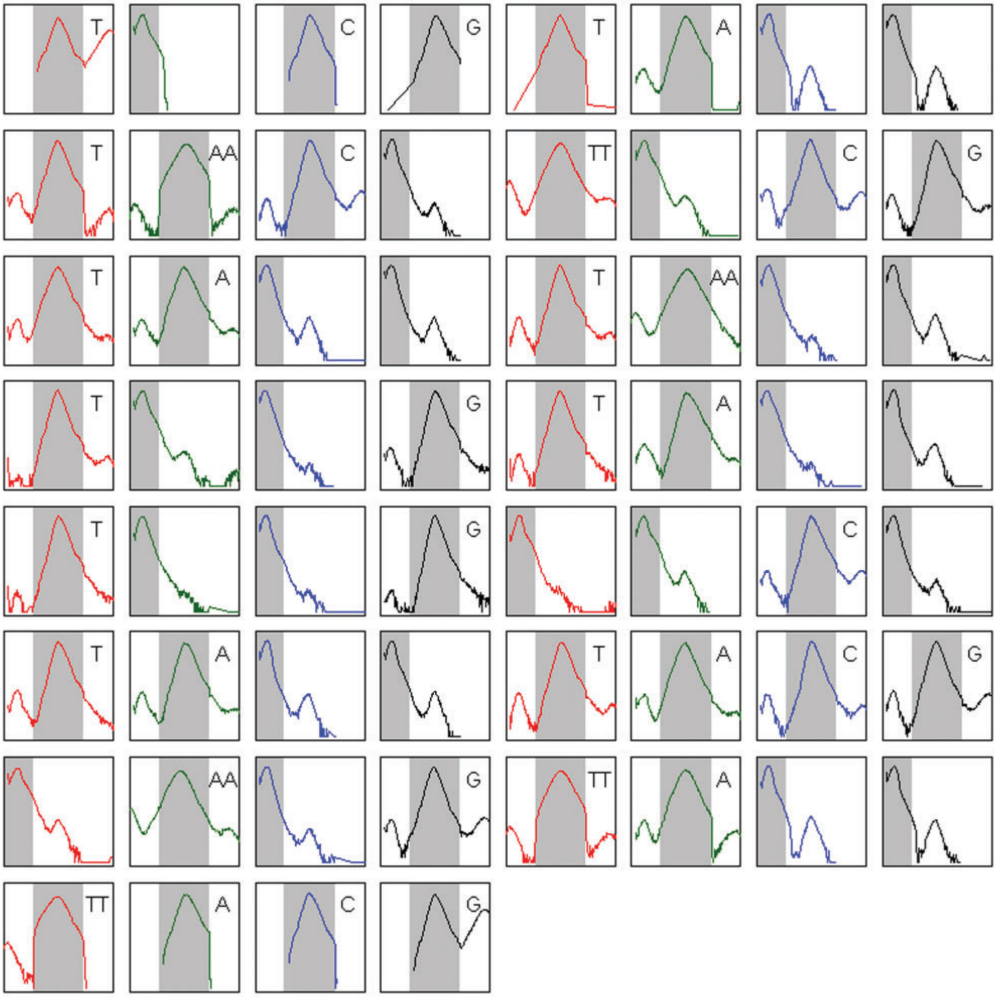

Fig. 3. Flow value histograms for *G.morhua* mate-pair reads (forward matches, $N = 7016764$). The y-axis is on a log10 scale. The 15 flow cycles correspond to the 42 positions of the linker sequence. The gray areas contain correct base calls. Subpeaks point toward putative PCR errors.

we observed a per-flow error rate of 0.153% (including negative flows), which is believed to underestimate the true error rate, first because we have filtered out bad alignments prior to our analysis, and also because the linker sequence only contains 1- and 2mers, and longer homopolymer runs are more likely to contain errors than shorter ones.

Even when using the conservative estimates, we get a fraction of 4–25% putative PCR errors in relation to all errors (Table 2).

This corroborates our theory that PCR errors might be an important error source in pyrosequencing. Notably, the fraction of PCR errors decreases with respect to the corresponding flow cycle in a read (Fig. 4).

3 SIMULATING PYROSEQUENCING DATA

We have in our previous work (Balzer *et al.*, 2010) presented flowsim, a simulation tool for 454 pyrosequencing data that uses empirical distributions of flow values to accurately model the pyrosequencing results and that provides the simulated data as SFF files.

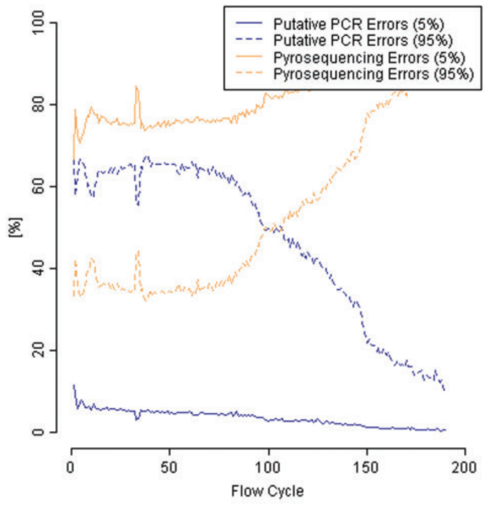

Fig. 4. Putative PCR and pyrosequencing error rates with respect to flow cycles (for underlying flow value intervals of size 5 and 95%).

3.1 The flowsim simulation pipeline

In order to extend flowsim and to take into account the various error types described above, the software is now split into several independent tools, each tool modeling a separate stage in the sequencing process.

The flowsim pipeline currently comprises the following utilities:

- clonesim, which simulates shearing of an input genome according to a user-specified distribution of clone lengths.
- gelfilter, which selects a subset of input clones according to a minimum and a maximum clone size.
- duplicator, which introduces artificial duplicates of clones.
- kitsim, which attaches the end of the A-adaptor (which consists of the four letter 'key' at the beginning of reads, typically TCAG), and the B-adaptor.
- mutator, which mutates the input sequences with random insertions, deletions and substitutions at user-specified rates.
- flowsim, which simulates pyrosequencing of a set of input sequences, calculates quality scores, filters and quality-trims the reads, and outputs the resulting SFF file.

With the exception of flowsim which outputs an SFF file, all utilities work with Fasta sequences as input and output, and by default read from standard input and write to standard output. Thus, a simple command for creating 100 000 reads from an input genome, using default parameters, would be:

'clonesim -c 100000 input.fasta | kitsim | flowsim -o out.sff'.

The separation into multiple programs provides more flexibility, and it is easy for users to implement and apply additional tools. For instance, a user could simulate amplicon sequencing by replacing clonesim with a program that simulates amplicons, and use the remaining flowsim pipeline to simulate the 454 sequencing process. Similarly, mate-pair libraries can be simulated by interposing a program that simulates circularization and fragmentation.

3.2 Simulation results

For simulation, we used a 764 Mb genomic scaffold from sea bass (*D.labrax*) generated from Sanger sequencing (Kuhl *et al.*, 2010), where we also had available approximately 30× coverage 454 shotgun reads for comparison.

We used flowsim to simulate a high number of reads corresponding to 10× coverage, providing sufficient clone lengths for 800 flows (Titanium), using empirical distributions as flow model and quality degradation along the sequence, but only taking into account homopolymer length errors arising from the flow value distributions (i.e. we did not make use of kitsim or mutator). We assembled our simulated reads using Newbler beta version 2.5 (provided by Roche Diagnostics) and compared the assembly results, namely contig sizes, with the assembly of randomly chosen real *D.labrax* Titanium reads corresponding to equal coverage. Our assemblies of simulated reads were substantially better than those of real data in terms of contig sizes.

When carrying out earlier simulations from *Escherichia coli* (Balzer *et al.*, 2010), we assumed strain-specific differences to be responsible for discrepancies between the assembly of real shotgun data and that of simulated data. Since we are now comparing reads that we simulated from the *D.labrax* reference scaffold with shotgun reads from the same individual, we can exclude this factor. Examining the simulation accuracy of flowsim, we identified the following factors to be potentially relevant for our assemblies having better statistics than the assemblies of real reads: coverage (average overall coverage, coverage distribution, zero-coverage regions), adaptors, putative PCR errors, pyrosequencing errors. Other errors, such as multiple DNA fragments associated with one bead, are likely to have been eliminated by the Roche quality-filtering.

In Section 2, we have examined each of these sources of variability and can make use of the updated flowsim pipeline described above for further simulations.

After having added errors to the same simulated clones that we used in earlier assemblies, i.e. first attaching adaptor sequences and subsequently introducing PCR noise at rates comparable with those found in real shotgun data, we ran flowsim and performed a new assembly of our simulated reads. It still outperforms an assembly of real reads, but assembly statistics like contig sizes and the percentage of aligned reads and bases are closer to the assembly of real reads when simulating additional error sources. We will also more closely examine to what extent the real pyrosequencing *D.labrax* data contain heterozygosity (coming from a diploid fish) and how a similar effect can be introduced into the simulated reads.

While the current version of our simulator uses a uniform coverage distribution over the input genome, we assume that this approach is not sufficiently realistic. Typically, there is greater than a 100-fold variation in coverage (Harismendy *et al.*, 2009). This is in agreement with our data, finding per-base coverage up to 760 in *D.labrax* (average 33) and 1152 in *E.coli* (average 110).

Using cd-hit-454 (Niu *et al.*, 2010), we observed duplicate read rates between 2.73 and 19.13% for *D.labrax* and between 0.19 and 10.71% for *E.coli*, with 98–100% sequence identity, while—as expected—our simulated reads (*D.labrax*, $10-30\times$ coverage) only contained very few (0.01%) duplicates or almost-duplicates.

4 DISCUSSION AND CONCLUSIONS

In this study, we have explored different error sources of 454 pyrosequencing. Previously, light signal distributions from the pyrosequencing chemistry and carry-forward/incomplete extension have been seen as the major sources of noise. Neighboring peaks in flow value distributions, observed in earlier analyses when aligning reads to a reference, were believed to arise from biological differences between reads and reference, but by matching reads against a known mate-pair linker sequence and only using these short alignments for our analyses, we eliminate this source of error. We speculate that, beside pyrosequencing errors due to inaccuracies in the sequencing process, also errors from the PCR library preparation step could account for a high percentage of observed errors. Hence, we present an empirical approach to support our assumptions, based on the presence of strong neighboring peaks in the distributions of flow values that correspond to the linker sequence. We see a clear decrease in the proportion of errors assigned to neighboring peaks as we move towards the end of the read, which is most likely due to the increase in pyrosequencing errors caused by widening flow value distributions. This implies that neighboring peak errors occur at an approximately constant rate along the read.

Furthermore, it is difficult to see how the neighboring peaks could arise from known error sources. Random noise in flow values should result in distributions similar to Gaussian, and we see no reason for CAFIE errors to concentrate around integral values. Thus, we believe that the neighboring peaks are caused by real differences in the library clones, but we cannot currently suggest an explanation on how these arise.

Finally, our new additions to the simulation pipeline enable us to simulate many of the identified errors, and we see that the resulting assemblies are approaching those obtained from real data. Nevertheless, we are examining further factors that we believe to be relevant in read simulation and quality assessment.

ACKNOWLEDGEMENTS

The authors wish to thank Dr Richard Reinhardt, Max Planck Institute for Molecular Genetics, Berlin, and Dr Kjetill Jakobsen, University of Oslo, and the Cod Genome Project team for kindly providing us with Titanium raw data. We also would like to acknowledge Anders Lanzén, University of Bergen, Dr Niina Haiminen, IBM T.J. Watson Research Center, Dr Christopher Quince, University of Glasgow and Markus Grohme, TH Wildau, for the fruitful discussions and assistance in analyses.

Funding: National Program for Research in Functional Genomics in Norway (FUGE) in the Research Council of Norway.

Conflict of Interest: none declared.

REFERENCES

Altschul,S.F. *et al*. (1990) Basic local alignment search tool. *J. Mol. Biol.*, **215**, 403–410.

Balzer,S. *et al*. (2010) Characteristics of 454 pyrosequencing data–enabling realistic simulation with flowsim. *Bioinformatics*, **26**, i420–i425.

Chou,H.H. and Holmes,M.H. (2001) DNA sequence quality trimming and vector removal. *Bioinformatics*, **17**, 1093–1104.

Gomez-Alvarez,V. *et al*. (2009) Systematic artifacts in metagenomes from complex microbial communities. *ISME J.*, **3**, 1314–1317.

Harismendy,O. *et al*. (2009) Evaluation of next generation sequencing platforms for population targeted sequencing studies. *Genome Biol.*, **10**, R32.

Hoff,K.J. (2009) The effect of sequencing errors on metagenomic gene prediction. *BMC Genomics*, **10**, 520.

Huse,S.M. *et al*. (2007) Accuracy and quality of massively parallel DNA pyrosequencing. *Genome Biol.*, **8**, R143.

Kuhl,H. *et al*. (2010) The European sea bass Dicentrarchus labrax genome puzzle: comparative BAC-mapping and low coverage shotgun sequencing. *BMC Genomics*, **11**, 68.

Kunin,V. *et al*. (2009) Wrinkles in the rare biosphere: pyrosequencing errors can lead to artificial inflation of diversity estimates. *Environ. Microbiol.*, **12**, 118–123.

Li,W. and Godzik,A. (2006) Cd-hit: a fast program for clustering and comparing large sets of protein or nucleotide sequences. *Bioinformatics*, **22**, 1658–1659.

Margulies,M. *et al*. (2005) Genome sequencing in microfabricated high-density picolitre reactors. *Nature*, **437**, 376–380.

Niu,B. *et al*. (2010) Artificial and natural duplicates in pyrosequencing reads of metagenomic data. *BMC Bioinformatics*, **11**, 187.

Quince,C. *et al*. (2009) Accurate determination of microbial diversity from 454 pyrosequencing data. *Nat. Methods*, **6**, 639–641.

Quince,C. *et al*. (2011) Removing noise from pyrosequenced amplicons. *BMC Bioinformatics*, **12**, 38.

Quinlan,A.R. *et al*. (2008) Pyrobayes: an improved base caller for SNP discovery in pyrosequences. *Nat. Methods*, **5**, 179–181.

Roche Applied Science (2008) Genome Sequencer Data Analysis Software Manual, Software Version 2.0.00, Roche Diagnostics GmbH.

III

BIOINFORMATICS ORIGINAL PAPER

Vol. 29 no. 7 2013, pages 830–836
doi:10.1093/bioinformatics/btt047

Sequence analysis

Advance Access publication February 1, 2013

Filtering duplicate reads from 454 pyrosequencing data

Susanne Balzer[1,2], Ketil Malde[1,*], Markus A. Grohme[3] and Inge Jonassen[2,4]

[1]Norwegian Marine Data Centre, Institute of Marine Research, P.O. Box 1870, N-5817 Bergen, Norway, [2]Department of Informatics, University of Bergen, P.O. Box 7803, N-5020 Bergen, Norway, [3]Department of Molecular Biotechnology and Functional Genomics, University of Applied Sciences Wildau, Bahnhofstraße 1, D-15745 Wildau, Germany and [4]Computational Biology Unit, Uni Computing, Thormøhlensgate 55, N-5008 Bergen, Norway

Associate Editor: Martin Bishop

ABSTRACT

Motivation: Throughout the recent years, 454 pyrosequencing has emerged as an efficient alternative to traditional Sanger sequencing and is widely used in both *de novo* whole-genome sequencing and metagenomics. Especially the latter application is extremely sensitive to sequencing errors and artificially duplicated reads. Both are common in 454 pyrosequencing and can create a strong bias in the estimation of diversity and composition of a sample. To date, there are several tools that aim to remove both sequencing noise and duplicates. Nevertheless, duplicate removal is often based on nucleotide sequences rather than on the underlying flow values, which contain additional information.

Results: With the novel tool JATAC, we present an approach towards a more accurate duplicate removal by analysing flow values directly. Making use of previous findings on 454 flow data characteristics, we combine read clustering with Bayesian distance measures. Finally, we provide a benchmark with an existing algorithm.

Availability: JATAC is freely available under the General Public License from http://malde.org/ketil/jatac/.

Contact: Ketil.Malde@imr.no

Supplementary information: Supplementary data are available at *Bioinformatics* online

Received on June 6, 2012; revised on January 24, 2013; accepted on January 25, 2013

1 INTRODUCTION

When 454 Life Sciences (now Roche Diagnostics) released the GS20 sequencing platform in 2005 (Margulies *et al.*, 2005), it was the start of a revolution in sequencing technology. It has since been followed by other platforms, both subsequent generations from 454 and competing technologies like Illumina/Solexa and ABI/SOLiD. The increased throughput and decreasing per base cost of these second-generation sequencing technologies have made high-throughput sequencing an affordable tool for many new organisms and applications. The traditional Sanger sequencing is now 30 years old (Sanger *et al.*, 1977), and the error characteristics and artifacts intrinsic to the method are well characterized. Consequently, there are established methods for describing sequence quality (Ewing *et al.*, 1998; Ewing and Green, 1998). Standard methods and tools for detecting and dealing with common contamination like vector sequences or genomic contamination exist, some of them applicable to one or several second-generation sequencing technologies (Chou and Holmes, 2001; Falgueras *et al.*, 2010; Kong, 2011; White *et al.*, 2008). Experienced researchers will also be aware of the risk of artifacts like chimeric sequences arising through different mechanisms (Houseley and Tollervey, 2010; Kanagawa, 2003).

There are numerous approaches to the removal or correction of erroneous sequences or parts of sequences for different applications. These are especially tailored to metagenomics, but also to SNP detection, small RNA discovery and so forth, some of them using 454 pyrosequencing flow data instead of nucleotide sequences, with good results (Huse *et al.*, 2007; Kunin *et al.*, 2009; Quince *et al.*, 2009; Quince *et al.*, 2011; Quinlan *et al.*, 2008; Sogin *et al.*, 2006; Vacic *et al.*, 2008).

1.1 Background

Apart from sequencing errors, a second issue accounts for incorrect conclusions in metagenomic studies. Gomez-Alvarez *et al.* (2009) discovered that 454 sequence data contain an overabundance of reads that are exact or almost-exact duplicates of each other. This comprises both identical reads and reads that start at the same position in the genome but have different lengths or vary slightly, putatively owing to pyrosequencing errors. Although erroneous reads lead to an overestimation of the number of operational taxonomic units in a sample, duplicates artificially inflate the number of reads per operational taxonomic unit, used as an abundance measure. Gomez-Alvarez *et al.* (2009) report between 11% and 35% sequences in metagenomic datasets being artificial duplicates. With the 454 Replicate Filter (Gomez-Alvarez *et al.*, 2009; Teal and Schmidt, 2010), they provide a web-based solution for removing these artifacts, making use of the CD-HIT suite (Li and Godzik, 2006), a fast clustering program for sequences. However, CD-HIT was not specifically designed for 454 pyrosequencing data and operates on fasta input, i.e. on nucleotide sequences rather than on flow data, which is accompanied by information loss (see Section 1.2). With cd-hit-454, Niu *et al.* (2010) provide both a web and a stand-alone tool for the removal of artificial duplicates in metagenomic pyrosequencing data. Also, PyroCleaner (Mariette *et al.*, 2011) has been specifically designed for 454 data, but all these tools work on nucleotide sequences. Our main motivation for developing JATAC was to aid metagenomic projects in the tradition of 454 Replicate Filter and cd-hit-454, but leveraging additional information present in flow data. JATAC targets both the assembly of (meta)genomes and the accurate estimation of

*To whom correspondence should be addressed.

community compositions. Gomez-Alvarez *et al.* have shown that failure to remove duplicates resulted in misleading conclusions on the gene space in soil metagenomes (Gomez-Alvarez *et al.*, 2009). Furthermore, methods using sequence coverage to identify repeats (e.g. Malde *et al.*, 2006; Phillippy *et al.*, 2008) should not be applied to pyrosequencing data without first filtering duplicates.

1.2 Nucleotide space versus flow space

In 454 pyrosequencing, around one million DNA molecules are sequenced in parallel (~100 000 in the benchtop solution GS Junior), generating a series of so-called flow values for each molecule. One flow value corresponds to the number of identical bases incorporated in a single flow. The cycling order of the nucleotides is maintained throughout the sequencing process (T, A, C, G representing one flow cycle). The underlying sequence is inferred from the respective flow values of each nucleotide.

Flow values refer to the signal strength of the sequencing reaction (for details on the sequencing chemistry, see Margulies *et al.*, 2005). With increasing homopolymer length, the signal differences and thereby the discriminatory power of the base calling decrease, resulting in a well-known uncertainty about exact homopolymer lengths, especially for long homopolymers (Gilles *et al.*, 2011; Huse *et al.*, 2007; Margulies *et al.*, 2005). As nucleotide homopolymer length can only be expressed in integers, it is indispensable to carry out analyses based on flow data (expressed as double decimal values) instead of nucleotide sequences, i.e. in ‘flow space’ instead of ‘nucleotide space’.

The native output format of 454 pyrosequencing is the binary standard flowgram format (*.sff). It contains the flowgram for each read, whereby each flowgram consists of a sequence of flow values representing base incorporations. One flowgram corresponds to 800 flows (200 flow cycles) in the GS FLX/Junior Titanium chemistry, i.e. one flow value per position 1-800. The GS FLX+ chemistry uses 1600 flows (400 flow cycles).

In the following, we present a reference-free method and algorithm named JATAC that identifies duplicate reads based on the flowgram. Methods operating in flow space have been shown to be superior to methods working in nucleotide space, e.g. for noise removal in metagenomics amplicon data (see earlier in the text). Our results indicate that this is also the case for duplicate removal.

2 DUPLICATE FILTERING

2.1 Natural versus artificial duplicates

Library generation for 454 pyrosequencing involves an emulsion polymerase chain reaction (PCR) step where water-oil droplets are formed (Tawfik and Griffiths, 1998; Williams *et al.*, 2006). This segregates the complex reaction mixture into miniaturized compartments and allows for highly multiplexed DNA amplification reactions. In these so-called micro-reactors, single DNA molecules are clonally amplified onto beads and are then deposited on a PicoTiterPlateTM (PTP) for sequencing (Leamon *et al.*, 2003; Margulies *et al.*, 2005). An inherent artifact of 454 library preparation and sequencing is the generation of artificial duplicate sequences as a result of the emulsion PCR step.

There are three suspected sources for artificial duplicates: Emulsion PCR, background amplicon contamination and signal cross-talk on the PTP sequencing device.

Usually, the low DNA-to-bead ratio minimizes the possibility of loading a single bead with two distinct DNA molecules, thereby generating mostly single-copy beads for sequencing (Zheng *et al.*, 2010). Conversely, many beads will remain empty, and droplets containing several beads and a single DNA molecule will therefore result in loading these beads with identical copies of the original DNA molecule. The strongest manifestation of overloading empty beads with identical molecules can be observed during unwanted emulsion breakage, when the emulsions become chemically unstable during thermal cycling and the micro-reactors fuse into larger droplets.

An amplicon contamination of amplified library DNA molecules from a previous sequencing run can also lead to duplicate reads in following runs, but these types of duplicate errors can normally be avoided by preventing cross-contamination of sequencing library samples.

Signal duplicates are an effect of well-to-well cross-talk, where strong signals ‘bleed’ into neighbouring empty wells (Briggs *et al.*, 2007). With the launch of the 454 Titanium chemistry, well cross-talk has been minimized by metal coating of the PTP well surface (Roche Applied Science, 2008).

Most likely, the main source of duplicates can be attributed to the emulsion PCR step. As the beads are randomly distributed on the plate, and the DNA on each bead is amplified and sequenced independently, the final length and error content of the sequence read can differ, but in all cases, the starting position of the read will be identical for all duplicates.

In contrast to artificial duplicates, duplicates can also arise ‘naturally’, i.e. by chance through sampling DNA molecules that start at identical positions or in repetitive regions of a genome. For genomic shotgun sequencing projects, there is a correlation between genome coverage and the percentage of natural duplicates. With increasing read density, the amount of natural duplicates will also increase. In metagenomic datasets of high complexity, i.e. in the absence of dominant species, the percentage of natural duplicates should be very low. For metatranscriptomic samples, the discrimination of natural and artificial duplicates is much more difficult, as some highly expressed RNAs will be sequenced much more often. For such datasets, it is challenging to distinguish between artificial and natural duplicates (Niu *et al.*, 2010).

2.2 Benchmark dataset construction

To compare the performance of JATAC and cd-hit-454, we generated three benchmark datasets, each consisting of a dataset of (real) reads and information about duplicates within each set of reads. We chose sequence datasets where a reference was available to accurately assess duplicate removal. Benchmarking on reference-free metagenome datasets would have resulted in a set of duplicate clusters and an expected duplication rate but would give no indication of the accuracy of each method for duplicate detection.

We used the GS Reference Mapper v. 2.6 (Roche Applied Science, 2008) with default settings and processed the results from the benchmark datasets in the following way: to precisely get the correct alignment for the beginning of each read, we independently mapped our data to the original and reverse complement genome. The BAM file generated by the mapper was converted into SAM format using samtools (Li *et al.*, 2009) and split into matches to the forward and reverse strands of the genome, retaining only forward matches relative to the respective reference (original/reverse complement). A subset of alignments was identified by extracting only unique alignment start positions and 16-nucleotide sequence prefixes, discarding alignments where the initial part of the read was masked (i.e. having 'H' as the first element of the field). Clusters of duplicate alignments were then extracted by grouping all reads with the same prefix and aligned position. This procedure is for reference dataset generation only and not to be confused with the JATAC algorithm (see Section 2.3).

For the first benchmark dataset, we mapped 1 270 325 *Dicentrarchus labrax* (sea bass) 454 GS FLX Titanium reads to the corresponding (Sanger-sequenced) reference scaffold (Kuhl *et al.*, 2010). As a result, 35.80% of the 1 270 325 reads are part of a cluster of at least two flowgrams that map to the same position in the reference genome. By subtracting one representative per duplicate cluster, we estimated the overall duplicate rate for *D.labrax* to be 20.18%. Of all duplicate clusters, 75% contain two, another 18% contain three and 5% contain four flowgrams. The biggest cluster contains 159 flowgrams (see Figs 1 and 2). The genomic reference used for sea bass is incomplete leading to a possible over-estimation of artificial duplicates. However, this does not introduce any bias in favour of any of the clustering algorithms. In other respects, this dataset is ideal as a benchmark, as the 454 sequences stem from the same individual on which the reference is based while the reference was constructed using a separate sequence set.

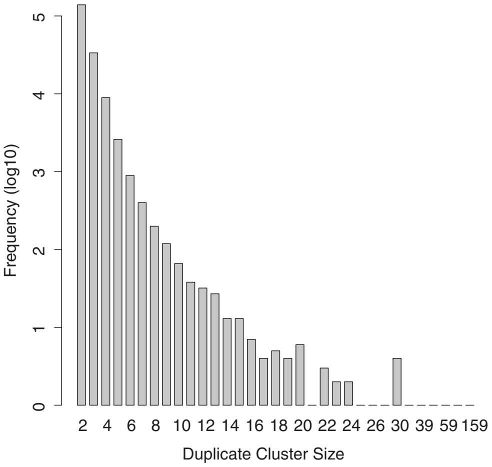

Fig. 1. True duplicate cluster sizes from *D.labrax* benchmark dataset. The biggest cluster contains 159 reads (see Fig. 2)

The second and third benchmark dataset consisted of two 454 GS Junior Titanium runs of an isolate of *Escherichia coli* O104:H4, containing 137 528 and 135 992 reads, respectively. This Shiga toxin producing strain was responsible for an outbreak of food poisoning in Germany in 2011 (Loman *et al.*, 2012).

2.3 Removal of duplicates with JATAC

We cluster flowgrams rather than reads and operate solely in flow space (see Section 1.2). We take into account the 454 key and quality trimming information included in the flow data files, which means that only informative flow values are used in the duplicate removal algorithm [see Equation (3)].

2.3.1 Preclustering Our clustering algorithm involves calculating the pairwise distances of all flowgrams. As this is computationally expensive on a dataset with more than a million flowgrams (typical 454 FLX Titanium run), we perform a preclustering step that creates subsets of flowgrams. Subsequent clustering is only performed on these subsets, which means that flowgrams from different subsets cannot be identified as duplicates of each other.

For preclustering, we use a varying seed of at least eight flows, starting with the first flow. For each of these flows, we only take into account if the flow value was 'negative' (i.e. <0.5) or 'positive' (i.e. ≥ 0.5, leading to at least one called base).

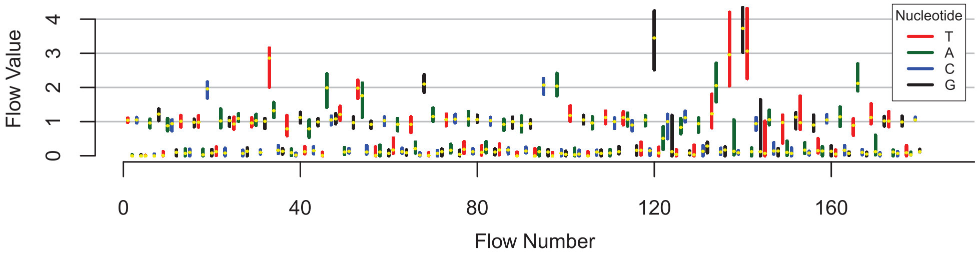

Fig. 2. Biggest flowgram cluster from *D.labrax* reference dataset (159 reads). Each vertical bar represents the range of flow values in this flow. The median flow value is plotted in yellow. The wide range of flow values in longer homopolymers, as well as the broad distributions of flow values at flow 122-124 and 144-145 represent under- and overcalls leading to indels and substitutions in the resulting nucleotide sequences. The longest flowgram was trimmed after flow no. 180 by the 454 software. The reads in the cluster have an average length of 88 bp in nucleotide space (+/− 14 bp, maximum 102 bp)

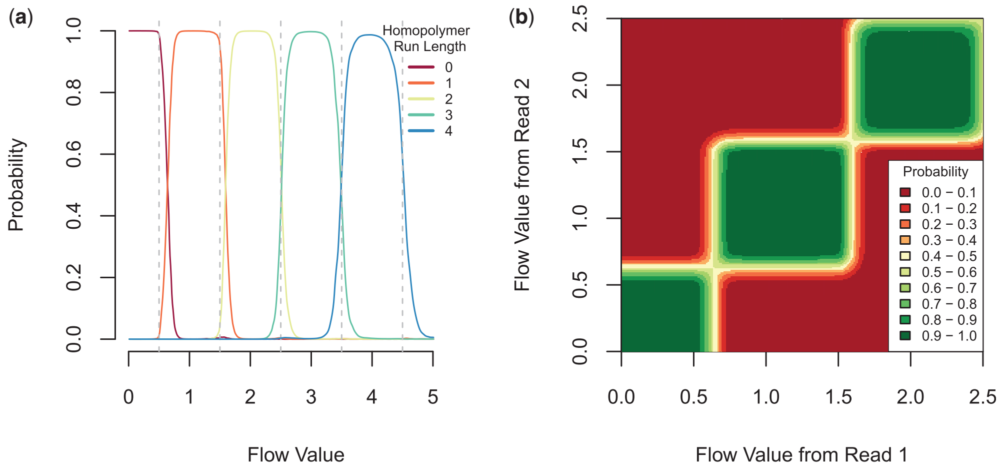

Fig. 3. (**a**) Probability for homopolymer lengths given a flow value [see Equation (1)]. (**b**) Probability for two homopolymer lengths being equal, given two flow values [see Equation (2)]. Both figures show the probabilities related to the first 10 flow cycles; for details, see Balzer *et al.* (2010)

For preclusters containing >2000 flowgrams, we gradually increase this seed to further split them up. In addition, we require flowgrams within one precluster to start with the same homopolymer length.

2.3.2 Distance measures To assess how similar two flowgrams are, we define a distance measure. This is similar to the distance definition by Quince *et al.* (2011) but directly compares two flowgrams rather than one flowgram with a perfect flowgram consisting of integers. We begin by applying Bayes' Theorem to calculate the probability for a homopolymer length being equal to h when observing a flow value f (see Fig. 3a):

$$P(h|f) = \frac{P(f|h) \cdot P(h)}{P(f)}. \qquad (1)$$

The prior—the homopolymer length distribution $P(h)$, the flow value distribution $P(f)$ and the likelihood distribution $P(f|h)$ are taken from earlier analyses and consist of an average smoothed distribution of *D.labrax* and *E.coli* flowgrams, mapped to their respective reference genomes and taking into account quality degradation towards later flow cycles. Determination of these distributions has been described in detail in Balzer *et al.* (2010). We argued earlier that the distributions are representative for other species for homopolymer lengths up to 5, and they can be downloaded from the flower website (http://biohaskell.org/Applications/Flower). Furthermore, we excluded any overfitting issues by demonstrating that the probability lookup tables are more or less interchangeable without impacting the outcome too much: when clustering *D.labrax* data with the use of a lookup table created from *E.coli* flow value distributions, our results were equally good as when using the smoothed average distribution from *D.labrax* and *E.coli* (see Section 2.3.2).

If we assume that two flowgrams, fg_a and fg_b, are independent from each other, then we can further calculate the probability that the homopolymer lengths, h_{ai} and h_{bi}, are equal, given two flow values, f_{ai} and f_{bi} (see Fig. 3b), the latter being flow values from fg_a and fg_b in the same flow (i.e. position) *i*.

$$P(h_{ai} = h_{bi}|f_{ai}, f_{bi}) := \begin{cases} 1 & \text{if } f_{ai} \text{ or } f_{bi} > 5.5 \\ 1 & \text{if } f_{ai} \text{ and } f_{bi} > 2.5 \\ \sum_{k=0}^{5} P(h_{ai} = k|f_{ai}) \cdot P(h_{bi} = k|f_{bi}) & \text{else.} \end{cases} \qquad (2)$$

For reasons of algorithm robustness, we assign a fixed probability score of 1 if at least one flow value is >5.5 or if both flow values are >2.5, thereby giving lower and better resolved flow values more weight in similarity calculations [see Equation (3)]. The latter corresponds to the observation that the most common sequencing error in 454 pyrosequencing is due to incorrectly determined homopolymer stretches (see Section 1.2).

In all other cases, we sum up the probabilities for the two flow values leading to the same homopolymer length $0, \ldots, 5$ to obtain a realistic estimate for the two values resulting in homopolymers of equal length. The flow-position-wise calculation of probabilities ensures that the two flow values in question always relate to the same nucleotide (see Fig. 2).

It is assumed that the flow values of one flowgram are not correlated. The assumption is strictly speaking invalid owing to the occurrence of carry forward and incomplete extension, phenomena that the 454 software partly corrects for. Under this assumption, we can define the distance $d(fg_a, fg_b)$ between two flowgrams as follows:

$$\begin{aligned} d(fg_a, fg_b) : &= -log(\prod_{i=l}^{m} P(h_{ai} = h_{bi}|f_{ai}, f_{bi}))/(m-(l-1)) \\ &= \sum_{i=l}^{m} -log(P(h_{ai} = h_{bi}|f_{ai}, f_{bi}))/(m-(l-1)) \end{aligned} \qquad (3)$$

with

$$l = \max\{\text{left trimpoint}(fg_a), \text{left trimpoint}(fg_b)\},$$
$$m = \min\{400, \text{right trimpoint}(fg_a), \text{right trimpoint}(fg_b)\},$$

the trimpoints being defined by the 454 software.

2.3.3 Hierarchical flowgram clustering Once we have defined our distance measure, we iterate through the files that contain the preclustered flowgrams (see Section 2.3.1) and perform agglomerative clustering on one file at a time.

We now start with one flowgram per cluster (i.e. each cluster being a singleton) and calculate all pairwise distances between flowgrams. In each clustering step, the two clusters, which have the smallest distance from each other, are combined into a new cluster. Two updates are then performed: First, a consensus flowgram is determined for the new cluster by calculating the per-flow median of flow values from all flowgrams in this cluster (quality-trimmed regions only). Second, the distances between the new cluster and all other clusters are updated. We continue clustering until all pairwise distances between clusters exceed a given stringency threshold.

We experimented with different threshold settings for the distance measure. Also, we only use the first 400 flow values of a flowgram [or all flow values up to the lowest trimpoint, see Equation (3)].

Our method of calculating a consensus flowgram is based on our observation that flow values in true duplicate clusters tend to stretch out to one side of the integer for each flow position (see Fig. 2). Correspondingly, we calculate the median flow value per flow.

2.3.4 Output We have implemented three modes for determining a representative of a flowgram cluster: 'longest', 'best' or 'consensus'. Also, we provide both fasta and sff output to meet the needs of a broad range of users. Choosing the longest read from a cluster is straightforward; choosing the best read involves calculating the squared sum of the flow values' distance to the corresponding integers, normalized by flowgram length. Obviously, flow values that lie close to integers have a high accuracy. The consensus flowgram is the median flowgram that previously has been used to (re-)calculate the distances between clusters in the clustering algorithm. When using the consensus option, the output of a cluster is therefore an artificial consensus flowgram of all flowgrams in the cluster (at least if a cluster contains more than one read).

2.4 Benchmark of methods

In general, when calculating the duplicate rate for a dataset without comparing with a reference, the result strongly depends on the stringency at which reads are regarded as being 'similar enough'. We ran JATAC on all *D.labrax* FLX Titanium and *E.coli* Junior Titanium reads (see Section 2.2) and clustered them at different stringency thresholds, the threshold being the maximum allowed distance when combining two clusters [see Equation (3)]. Also, we used the command line version of cd-hit-454 (v. 4.6, Li and Godzik, 2006; Niu *et al.*, 2010) to cluster our shotgun data at different stringency settings (between 91% and 100%), where 98% is the default stringency in cd-hit-454. Results are given in Table 1.

Table 1. Duplicate clustering results for cd-hit-454 and JATAC

Stringency[a]	Estimated duplicate rate/Jaccard index		
	E.coli (Run 1)	*E.coli* (Run 2)	*D.labrax*
cd-hit-454			
100%	3.24%/0.30	6.56%/0.29	2.73%/0.09
99%	8.20%/0.75	15.64%/0.73	13.21%/0.45
98%	9.29%/0.82	17.59%/0.81	19.13%/0.64
97%	9.57%/0.83	18.04%/0.82	20.82%/0.66
96%	9.67%/0.83	18.18%/0.82	21.35%/0.65
95%	9.72%/0.83	18.25%/0.83	21.58%/0.63
94%	9.74%/0.83	18.29%/0.83	21.72%/0.61
93%	9.76%/0.83	18.30%/0.83	21.81%/0.59
92%	9.77%/0.83	18.31%/0.82	21.88%/0.59
91%	9.77%/0.83	18.32%/0.82	21.88%/0.59
JATAC			
0.00	0.00%/0.00	0.00%/0.00	0.00%/0.00
0.01	7.66%/0.71	15.10%/0.72	18.28%/0.65
0.02	8.60%/0.78	16.67%/0.79	20.40%/0.72
0.03	9.11%/0.82	17.54%/0.83	21.36%/0.74
0.04	9.41%/0.84	18.05%/0.85	21.89%/0.75
0.05	9.63%/0.85	18.41%/0.86	22.22%/0.76
0.06	9.77%/0.86	18.65%/0.86	22.45%/0.77
0.07	9.89%/0.86	18.82%/0.87	22.61%/0.77
0.08	9.97%/0.86	18.96%/0.87	22.75%/0.77
0.09	10.03%/0.87	19.08%/0.88	22.85%/0.77
0.1	10.08%/0.87	19.16%/0.88	22.93%/0.77
True duplicate rate	9.65%	18.61%	20.18%

[a]The clustering stringency corresponds to a sequence identity threshold for cd-hit-454 and to a distance threshold for JATAC. For the latter, a higher distance corresponds to lower identity.

To evaluate to what extent our JATAC algorithm allows for a more effective removal of artificial duplicates compared with the nucleotide sequence-based cd-hit-454, we need a measure that compares two sets of clusters. The Jaccard index

$$\text{Jaccard} := a/(a + b + c) \tag{4}$$

can be used to compute the degree of similarity between the real set of true duplicate clusters (from our reference, see Section 2.2) and the set of duplicate clusters identified by the respective clustering algorithm. Those flowgram pairs that are correctly identified as duplicates of each other are counted as *a*; those that are not identified as duplicates, although they map to the same position in the reference genome, are counted as *b*; and those that are incorrectly identified as duplicates are counted as *c* (see Fig. 4). The flowgram pairs *b* and *c* can vaguely be understood as false positives and false negatives from a classification problem. However, the calculation of common classification indicators such as sensitivity and specificity would be misleading here, as it is not sufficient to identify a flowgram as an artificial duplicate of *some* other flowgram, but it is relevant *which* flowgrams are clustered together.

JATAC outperformed cd-hit-454 on all three datasets, regardless of sequencing platform (GS FLX/Junior Titanium), actual duplication rate or complexity (see Table 1 and Fig. 4) at similar

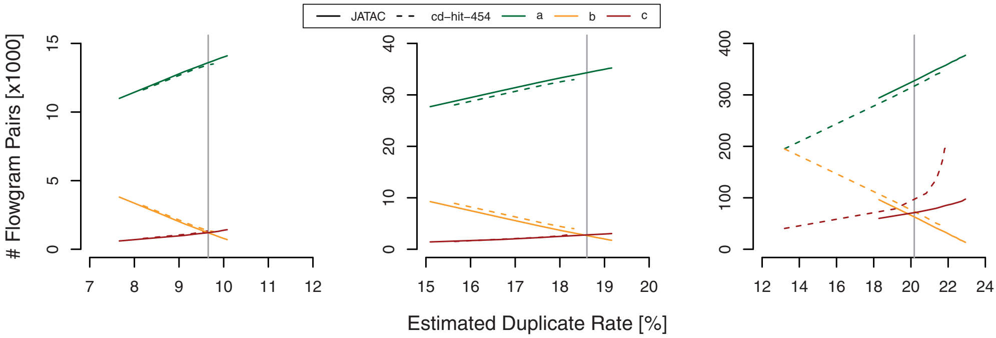

Fig. 4. Comparison of JATAC and cd-hit-454 duplicate clustering at different stringency settings and estimated duplicate rates surrounding the true duplicate rate (vertical grey line). The range of parametrization lies between 0.02 and 0.10 (distance threshold) for JATAC and between 99% and 92% for cd-hit-454 (identity threshold). Left: *E.coli* (Run 1). Centre: *E.coli* (Run 2). Right: *D.labrax*. For explanation of a, b and c pairs, see the text

estimated duplicate rates. We have experienced that a slight overestimation of the true duplicate rate gives the best results in terms of Jaccard index. This is true for both JATAC and cd-hit-454.

For the second *E.coli* dataset, cd-hit-454 underestimated the true duplicate rate even at a similarity threshold of 90% (data not shown). This illustrates one caveat when using duplicate removal tools such as JATAC or cd-hit-454, namely to determine at which stringency the reads should be filtered. However, the cd-hit-454 identity threshold and the JATAC distance threshold are not directly comparable. A JATAC distance of 0 does not exactly correspond to a cd-hit-454 stringency of 100%, as it is a lot more probable that two artificial duplicates share the same nucleotide sequence than that they share the exactly identical flowgram to the second decimal place. We have found that a distance measure of 0.05 is a good starting point for duplicate analyses resulting in a reasonable Jaccard index.

Additionally, we tested the effect of duplicate removal on assembly performance of the *E.coli* genome. Therefore, the two datasets were independently filtered for duplicates (keeping the longest read per cluster) and assembled together using Newbler. The rationale behind this was to reduce assembly artifacts from low coverage. In addition, owing to the separate duplicate filtering, we only removed a minimal amount of natural duplicates. We scored the resulting assemblies for a limited parameter set using Mauve assembly metrics (Darling *et al.*, 2011) and found no striking differences between JATAC and cd-hit-454 filtered assemblies. For both tools, the N50 increased to 126 844 bp in comparison with the unfiltered assembly with an N50 of 106 414 bp (see Supplementary Material). We conclude that the high and identical N50 value obtained using both approaches is likely to represent the highest possible assembly continuity for the given dataset and read length (Cahill *et al.*, 2010).

3 DISCUSSION

In this article, we have quantified the room for improvements when filtering 454 pyrosequencing shotgun data for artificial duplicates. We have successfully shown that, by the use of 454 flow data, a higher rate of artificial duplicates can be identified than by using sequence data only. Artificially duplicated reads can—apart from a generally higher processing and memory requirement—lead for example to incorrect conclusions about metagenomic dataset composition (Gomez-Alvarez *et al.*, 2009) or to biased quantification in digital karyotyping experiments (Dong *et al.*, 2011). Another likely problem could be false positive single nucleotide polymorphism calls in the presence of duplicated erroneous sequences. However, too stringent filtering might lead to an underestimation of abundance (Niu *et al.*, 2010).

Both JATAC and cd-hit-454 cannot distinguish natural from artificial duplicates, but the percentage of natural duplicates can be estimated from sequencing coverage by calculating the probability of multiple reads randomly starting at the same position (Niu *et al.*, 2010).

Although cd-hit-454's estimated duplicate rates were comparable with JATAC's estimations, the calculated cluster composition at similar duplication rates was of lower quality, manifested in a lower Jaccard index. This is likely the result of JATAC being better at handling homopolymer discrepancies and taking flow order into account, whereas cd-hit-454 is operating mostly on global similarity scores. The distance calculation in JATAC is a more robust way of finding duplicates, as it first identifies read pairs with different homopolymer lengths at low distances. Only with higher distance thresholds, reads with substitutions are taken into account. This behaviour closely models the 454 sequencing chemistry where substitution errors are less common than indels. Interestingly, the Jaccard index calculated from running cd-hit-454 on the *D.labrax* dataset degraded much faster around the true duplicate rate when compared with JATAC. This degradation could not be observed in the bacterial datasets and is likely due to a higher probability of matching unrelated sequences from a complex background. This phenomenon could also be relevant to metagenomic experiments of highly diverse communities, where tools such as cd-hit-454 and JATAC are most useful. A comprehensive overview of applications and effects of duplicate filtering, e.g. on genome assembly, can be found in Li *et al.* (2012).

JATAC's improved duplicate identification comes at a computational price, and its speed depends on the number of reads and the degree of duplication. JATAC takes up to several hours to filter an sff file for duplicates, ~1.5 h for a typical GS Junior run.

We have also evaluated JATAC on IonTorrent flow data, as both platforms share the same data format (sff). Although it is in principle possible to analyse ionograms using JATAC, the underlying flow data model has been optimized for pyrosequencing data, which is why we do not recommend JATAC for IonTorrent data in its present version.

ACKNOWLEDGEMENT

The authors wish to thank Dr Richard Reinhardt, Max Planck Institute for Molecular Genetics, Berlin, for kindly providing us with Titanium flow data.

Funding: National Program for Research in Functional Genomics in Norway (FUGE) in the Research Council of Norway (RCN grant number 183640/S10).

Conflict of Interest: none declared.

REFERENCES

Balzer,S. *et al.* (2010) Characteristics of 454 pyrosequencing data–enabling realistic simulation with flowsim. *Bioinformatics*, **26**, i420–i425.

Briggs,A.W. *et al.* (2007) Patterns of damage in genomic DNA sequences from a Neandertal. *Proc. Natl Acad. Sci. USA*, **104**, 14616–14621.

Cahill,M.J. *et al.* (2010) Read length and repeat resolution: exploring prokaryote genomes using next-generation sequencing technologies. *PLoS One*, **5**, e11518.

Chou,H.H. and Holmes,M.H. (2001) DNA sequence quality trimming and vector removal. *Bioinformatics*, **17**, 1093–1104.

Darling,A.E. *et al.* (2011) Mauve assembly metrics. *Bioinformatics*, **27**, 2756–2757.

Dong,H. *et al.* (2011) Artificial duplicate reads in sequencing data of 454 Genome Sequencer FLX System. *Acta Biochim. Biophys. Sin. (Shanghai)*, **43**, 496–500.

Ewing,B. *et al.* (1998) Base-calling of automated sequencer traces using phred. I. Accuracy assessment. *Genome Res*, **8**, 175–185.

Ewing,B. and Green,P. (1998) Base-calling of automated sequencer traces using phred. II. Error probabilities. *Genome Res*, **8**, 186–194.

Falgueras,J. *et al.* (2010) SeqTrim: a high-throughput pipeline for pre-processing any type of sequence read. *BMC Bioinformatics*, **11**, 38.

Gilles,A. *et al.* (2011) Accuracy and quality assessment of 454 GS-FLX Titanium pyrosequencing. *BMC Genomics*, **12**, 245.

Gomez-Alvarez,V. *et al.* (2009) Systematic artifacts in metagenomes from complex microbial communities. *ISME J.*, **3**, 1314–1317.

Houseley,J. and Tollervey,D. (2010) Apparent non-canonical trans-splicing is generated by reverse transcriptase *in vitro*. *PLoS One*, **5**, e12271.

Huse,S.M. *et al.* (2007) Accuracy and quality of massively parallel DNA pyrosequencing. *Genome Biol.*, **8**, R143.

Kanagawa,T. (2003) Bias and artifacts in multitemplate polymerase chain reactions (PCR). *J. Biosci. Bioeng.*, **96**, 317–323.

Kong,Y. (2011) Btrim: a fast, lightweight adapter and quality trimming program for next-generation sequencing technologies. *Genomics*, **98**, 152–153.

Kuhl,H. *et al.* (2010) The European sea bass Dicentrarchus labrax genome puzzle: comparative BAC-mapping and low coverage shotgun sequencing. *BMC Genomics*, **11**, 68.

Kunin,V. *et al.* (2009) Wrinkles in the rare biosphere: pyrosequencing errors can lead to artificial inflation of diversity estimates. *Environ. Microbiol.*, **12**, 118–123.

Leamon,J.H. *et al.* (2003) A massively parallel PicoTiterPlate based platform for discrete picoliter-scale polymerase chain reactions. *Electrophoresis*, **24**, 3769–3777.

Li,H. *et al.* (2009) The sequence alignment/map format and SAMtools. *Bioinformatics*, **25**, 2078–2079.

Li,W. *et al.* (2012) Ultrafast clustering algorithms for metagenomic sequence analysis. *Brief. Bioinform.*, **13**, 656–668.

Li,W. and Godzik,A. (2006) Cd-hit: a fast program for clustering and comparing large sets of protein or nucleotide sequences. *Bioinformatics*, **22**, 1658–1659.

Loman,N.J. *et al.* (2012) Performance comparison of benchtop high-throughput sequencing platforms. *Nat. Biotechnol.*, **30**, 434–439.

Malde,K. *et al.* (2006) RBR: library-less repeat detection for ESTs. *Bioinformatics*, **22**, 2232–2236.

Margulies,M. *et al.* (2005) Genome sequencing in microfabricated high-density picolitre reactors. *Nature*, **437**, 376–380.

Mariette,J. *et al.* (2011) Assessment of replicate bias in 454 pyrosequencing and a multi-purpose read-filtering tool. *BMC Res. Notes*, **4**, 149.

Niu,B. *et al.* (2010) Artificial and natural duplicates in pyrosequencing reads of metagenomic data. *BMC Bioinformatics*, **11**, 187.

Phillippy,A.M. *et al.* (2008) Genome assembly forensics: finding the elusive mis-assembly. *Genome Biol.*, **9**, R55.

Quince,C. *et al.* (2009) Accurate determination of microbial diversity from 454 pyrosequencing data. *Nat. Methods*, **6**, 639–641.

Quince,C. *et al.* (2011) Removing noise from pyrosequenced amplicons. *BMC Bioinformatics*, **12**, 38.

Quinlan,A.R. *et al.* (2008) Pyrobayes: an improved base caller for SNP discovery in pyrosequences. *Nat. Methods*, **5**, 179–181.

Roche Applied Science. (2008) Genome Sequencer Data Analysis Software Manual, Software Version 2.0.00. *Roche Diagnostics GmbH*.

Sanger,F. *et al.* (1977) DNA sequencing with chain-terminating inhibitors. *Proc. Natl Acad. Sci. USA*, **74**, 5463–5467.

Sogin,M.L. *et al.* (2006) Microbial diversity in the deep sea and the underexplored 'rare biosphere'. *Proc. Natl Acad. Sci. USA*, **103**, 12115–12120.

Tawfik,D.S. and Griffiths,A.D. (1998) Man-made cell-like compartments for molecular evolution. *Nat. Biotechnol.*, **16**, 652–656.

Teal,T.K. and Schmidt,T.M. (2010) Identifying and removing artificial replicates from 454 pyrosequencing data. *Cold Spring Harb. Protoc.*, **2010,** pdb.prot5409.

Vacic,V. *et al.* (2008) A probabilistic method for small RNA flowgram matching. *Pac. Symp. Biocomput.*, **2008**, 75–86.

White,J.R. *et al.* (2008) Figaro: a novel statistical method for vector sequence removal. *Bioinformatics*, **24**, 462–467.

Williams,R. *et al.* (2006) Amplification of complex gene libraries by emulsion PCR. *Nat. Methods*, **3**, 545–550.

Zheng,Z. *et al.* (2010) Titration-free massively parallel pyrosequencing using trace amounts of starting material. *Nucleic Acids Res.*, **38**, e137.