FROM SEQUENCING READS TO MICROBIAL DIVERSITY:

BIOINFORMATIC ALGORITHMS FOR PROCESSING AMPLICON SEQUENCING DATA

Mohamed Mysara Ahmed

Supervisors:
Prof. Dr. D. Charlier
Prof. Dr. J. Raes
Dr. P. Monsieurs

Dissertation submitted in fulfilment of the requirements for the degree of Doctor (Ph.D.) in Bio-Engineering Sciences from department of Bio-Engineering Sciences, Faculty of Sciences and Bio-Engineering Sciences, Vrije Universiteit Brussel and Doctor (Ph.D.) of Biomedical Sciences from department of Biomedical Sciences, Faculty of Medicine, Katholieke Universiteit Leuven.

September 2016

Mohamed Mysara Ahmed: From Sequencing Reads To Microbial Diversity: Bioinformatic Algorithms For Processing Amplicon Sequencing Data

© September 2016, Vrije Universiteit Brussel

Uitgegeven in eigen beheer, Mohamed Mysara Ahmed, Mol Belgium

Alle rechten voorbehouden. Niets uit deze uitgave mag worden vermenigvuldigd en/of openbaar gemaakt worden door middel van druk, fotokopie, microfilm, elektronisch of op welke andere wijze ook zonder voorafgaandelijke schriftelijke toestemming van de uitgever.

All rights reserved. No part of the publication may be reproduced in any form by print, photoprint, microfilm, electronic or any other means without written permission from the publisher.

ISBN 978 90 769 7123 0

ACKNOWLEDGEMENT

First and foremost, I thank God for giving me power, strength and patience to finalize my PhD dissertation. My utter gratitude goes to my PhD mentor Pieter Monsieurs for giving me this opportunity to do a PhD at SCK•CEN microbiology lab. I cannot thank Pieter enough for his guidance, patience and support throughout this thesis work and of course for all the flexibility he gave me and the trust he had in me. His efforts exceeded the work limits, helping me in all aspects of my life. I am glad to have him as my mentor and my friend.

I would also like to express my sincere gratitude to my promotors Jeroen Raes and Daniel Charlier for their support and numerous efforts during my PhD. It was a privilege to have them both as my promotors.

Besides my promotors, I would like to thank the rest of my jury members for dedicating their time and efforts to review this work. I am greatly thankful for their insightful comments and hard questions which encouraged me to widen my research from various perspectives.

My gratitude goes to all my colleagues at SCK•CEN, particularly Natalie Leys and the rest of the microbiology and radiobiology unit. Special thanks goes to Hugo Moors, I will always cherish his stimulating conversations, and Sarah Baatout for her support.

I am greatly thankful to all the researchers I have collaborated with. This gave me an amazing opportunity to interact with such dynamic society gaining a lot of experience.

Thanks to my friends in Belgium, Ahmed Nagy, Mercy Njima, Salem Ben Hamouda, Hanène Badri, Hanane Derradji for the wonderful memories and amazing time. Special thanks goes to those in my home country particularly Amr Ali, Tarek Abouelwafa, Khalid Aboud, I am grateful to them for being part of my life.

I am utterly grateful for my Mum and Dad for their infinite love and support. Also I am thankful for my Brothers, Mohab and Haitham, and to my Sister, Basma.

My endless thanks to my wife, Raghda Ramadan, and my daughter Lilly for being in my life. You are the reason of my strength, I cannot have done it without you. This thesis is dedicated to you and to everyone who tries to make a difference.

ABSTRACT (English version)

The development of high-throughput sequencing technologies has revolutionized the field of microbial ecology by offering a cost-efficient method to assess microbial diversity at an unseen depth using 16S rRNA amplicon sequencing approaches. Different preprocessing algorithms need to be performed to obtain a collection of highly reliable sequencing reads, ending with a clustering step to group them into Operational Taxonomic Units (OTUs) However, this approach is posing various challenges: the removal of PCR artefacts (called chimera), correction of sequencing errors resulting from the sequencing technologies and clustering those sequences into OTUs. In this work various bioinformatics tools were developed to tackle those challenges. First, an ensemble classifier for chimera detection was developed named CATCh, which obtained a higher performance on different types of sequencing data compared to existing tools. Secondly, two artificial intelligence-based algorithms, NoDe and IPED, able to treat sequencing errors within 454 Pyrosequencing and Illumina MiSeq data respectively, were introduced. A benchmarking study was performed comparing NoDe and IPED, showing a more pronounced decrease of the error rate compared to other state-of-the art tools. Thirdly, a new method was developed introducing an adaptive cut-off score in the OTU clustering step, as such making the results of the OTU clustering less sensitive to variations in evolutionary rates between taxonomic lineages and to the region of the 16S rRNA gene targeted for amplification. Implementing such a dynamic cut-off value resulted in closer correspondence between the number of OTUs and the actual diversity of the samples. Finally, a benchmark analysis comparing existing pipelines for 16S rRNA metagenomics data processing was performed, showing that an integration of our in-house developed algorithms achieved the highest accuracy. Conclusively, the newly developed pipeline within this PhD translates amplicon sequencing data into high-quality OTUs tendering robust diversity estimates.

ABSTRACT (Dutch version)

Ontwikkelingen op het vlak van hoge-doorvoer sequeneringstechnologieën hebben geleid tot een ware revolutie binnen microbiële ecologie. Deze nieuwe technologieën bieden een goedkope en efficiënte manier om de microbiële diversiteit te bepalen op basis van 16S rRNA amplicon sequeneringsmethoden. Verschillende dataverwerkingsstappen moeten echter uitgevoerd worden om te komen tot betrouwbare en hoogkwalitatieve sequentieproducten, met als laatste stap een clustering van deze sequentiefragmenten in operationele taxonomische eenheden (OTUs). De verwerking van deze data vormt echter een belangrijke uitdaging: de detectie van artefacten te wijten aan de PCR reactie om amplicons aan te maken (chimera), de correctie van sequeneringsfouten veroorzaakt door de sequeneringstechnologie, en het groeperen van deze sequentiefragmenten in OTUs. In dit doctoraat hebben we verschillende bio-informatica-methoden ontwikkeld die bijgedragen hebben tot het oplossen van deze computationele uitdagingen. In een eerste fase ontwikkelden we een efficiënte methode voor het detecteren van chimere sequenties (CATCh), een methode die beter presteerde dan vergelijkbare algoritmen, en dit op een uitgebreide collectie van verschillende types sequeneringsdata. In een tweede fase werden twee artificiële intelligentie algoritmen ontwikkeld – NoDe en IPED – voor het corrigeren van sequeneringsfouten in respectievelijk 454 pyrosequencing en Illumina MiSeq data. Op basis van een vergelijkende studie bleek dat onze methodes leidden tot een sterkere daling van het foutenpercentage vergeleken met andere state-of-the-art algoritmen. In een derde stap werd een geoptimaliseerde methode ontwikkeld voor het bepalen van het minimale percentage sequentiesimilariteit dat vereist is om twee sequeneringsfragmenten te groeperen in dezelfde OTU. Door dit percentage afhankelijk te maken van de variaties in evolutionaire snelheid tussen verschillende taxonomische klassen, en afhankelijk van de hypervariable regio's in het 16S rRNA gen die onderworpen worden aan sequenering, werden meer accurate resultaten bekomen met een nauwkeurigere overeenkomst tussen het aantal OTUs en de werkelijke diversiteit in de biologische stalen. In een laatste fase van het onderzoek deden we een uitgebreide vergelijkende studie tussen verschillende gekende methodologieën, waarbij we konden aantonen dat een methodologie die onze algoritmen bevatten, tot de hoogste nauwkeurigheid leiden. Samengevat hebben de nieuwe algoritmen zoals ontwikkeld binnen dit doctoraat geleid tot een meer nauwkeurige methodologie voor het vertalen van 16S rRNA sequeneringsdata naar hoogkwalitatieve OTUs, daarbij resulterend in meer robuuste schattingen van de werkelijke microbiële diversiteit.

TABLE OF CONTENTS

ACKNOWLEDGEMENT III

ABSTRACT (English version) IV

ABSTRACT (Dutch version) V

TABLE OF CONTENTS VI

LIST OF FIGURES IX

LIST OF TABLES X

LIST OF ABBREVIATIONS XI

LIST OF PUBLICATIONS XII

Chapter 1 Introduction 1

1.1. Background of the study 2

1.1.1. ***Prokaryotes*** 2

1.1.2. Bacterial species 4

1.1.3. Bacterial species and 16S rRNA gene 6

1.1.4. 16S rRNA gene databases 10

1.2. 16S rRNA gene and the advancement in sequencing technologies 13

1.2.1. Roche (454 Pyrosequencing) 14

1.2.2. Illumina (MiSeq) 15

1.2.3. Ion Torrent (PGM) 16

1.2.4. PacBio (SMRT) 17

1.2.5. NanoPore (MinION) 18

1.3. 16S rRNA amplicon sequencing 20

1.3.1. Hight-throughput sequencing of 16S rRNA gene 20

1.3.2. Problems with data analysis 21

1.4. Bioinformatics intervention 24

1.4.1. Pre-processing 24

1.4.2. Processing 28

1.4.3. Post-processing 29

1.5. Thesis outline 31

Chapter 2 NoDe: a Fast Error-Correction Algorithm for Pyrosequencing Amplicon Reads 35

2.1. Abstract 36

2.2. Introduction 37

2.3. Materials and methods 40

2.3.1. Mock communities 40

2.3.2. Error calculation and chimera detection 42
2.3.3. Denoising algorithms 43
2.3.4. Preprocessing sequencing data 44
2.4. Results *46*
2.4.1. NoDe (Noise Detector) algorithm 46
2.4.2. Benchmarking of NoDe 49
2.4.3. Impact of error correction methods on OTU clustering 53
2.5. Discussion *57*
2.6. Conclusion *59*
Chapter 3 IPED: a Highly Efficient Denoising Tool for Illumina MiSeq Paired-end 16S rRNA Amplicon Sequencing Data **61**
3.1. Abstract *62*
3.2. Introduction *63*
3.3. Materials and methods *65*
3.3.1. Mock Communities 65
3.3.2. Pre-Processing Steps 67
3.3.3. Training Data 69
3.3.4. Evaluation Parameters 70
3.4. Results *70*
3.4.1. IPED Development 71
3.4.2. Benchmarking of IPED 75
3.5. Discussion *83*
3.6. Conclusion *85*
Chapter 4 CATCh: an Ensemble Classifier for Chimera Detection in 16S rRNA Sequencing Studies **87**
4.1. Abstract *88*
4.2. Introduction *89*
4.3. Materials and methods *93*
4.3.1. Included chimera detection algorithms 93
4.3.2. Building CATCh 94
4.3.3. Training and testing datasets 96
4.3.4. Validation datasets 96
4.3.5. Evaluation parameters 99
4.4. Results *100*
4.4.1. Feature selection 100
4.4.2. Training and testing CATCh 101
4.4.3. Performance of reference based algorithms 103
4.4.4. Performance of de novo based algorithms 105
4.4.5. Benchmarking on 454 Pyrosequencing data 107
4.4.6. Benchmarking on Illumina MiSeq data 111
4.4.7. Benchmarking on 18S rRNA amplicon sequencing data 111

4.4.8. Impact of CATCh on OTU clustering ... 111
4.4.9. Computational resources ... 112
4.5. Discussion ... *115*
4.6. Conclusion ... *119*
Chapter 5 Reconciliation Between Operational Taxonomic Units and Species Boundaries ... **121**
5.1. Abstract ... *122*
5.2. Introduction ... *123*
5.3. Materials and methods ... *126*
5.4. Results ... *128*
5.4.1. Divergent evolution of 16S rRNA variable regions ... 128
5.4.2. Robustness of OTU clustering when combining different 16S rRNA hypervariable and conserved regions ... 130
5.4.3. Robustness of OTU clustering using short read lengths ... 132
5.4.4. Dynamic family-dependent cut-off for OTU clustering ... 133
5.4.5. Comparative analysis using mock data ... 136
5.4.6. Application on real-life samples ... 137
5.5. Discussion ... *138*
5.6. Conclusion ... *143*
Chapter 6 From Reads to Operational Taxonomic Units: an Ensemble Processing Pipeline for MiSeq Amplicon Sequencing Data ... **145**
6.1. Abstract ... *146*
6.2. Introduction ... *147*
6.3. Materials and methods ... *151*
6.3.1. Data Description ... 151
6.3.2. Standardization of the Pipelines ... 152
6.3.3. Evaluation Criteria ... 155
6.4. Results and discussion ... *157*
6.5. Conclusion ... *163*
Chapter 7 General Discussion and Perspectives ... **165**
7.1. General discussion ... *166*
7.2. Perspectives ... *177*
BIBLIOGRAPHY ... **183**
APPENDIX A: ADDITIONAL INFORMATION ... **197**
APPENDIX B CONTRIBUTIONS ... **219**
APPENDIX C CURRICULUM VITAE ... **229**

LIST OF FIGURES

Figure 1:1. Illustration of the secondary structure of the 16S rRNA gene 9

Figure 1:2. Illustration of the growth in the 16S rRNA gene sequences over the years. 11

Figure 1:3. Venn diagram showing the number of shared taxa at genus level between SILVA, RDP-II and Greengenes. 12

Figure 1:4. Illustration of the various steps performed for Illumina, Roche, Ion torrent, PacBio and NanoPore sequencing platforms. 19

Figure 1:5. Overview of the various steps involved in amplicon sequencing data analysis, and three phases included: Pre-processing, Processing and Post-processing analysis. 26

Figure 1:6. Demonstration of the various alpha and beta diversity analysis of four samples. ... 30

Figure 1:7. Schematic overview of the thesis structure. 33

Figure 2:1. Effect of denoising algorithms with respect to position in read 51

Figure 3:1. Schematic overview showing the different steps of the IPED algorithm. 75

Figure 3:2. Plot showing the error rate versus the position in the read after being treated with Pre-cluster, UNOISE, and IPED. 78

Figure 4:1. Illustration of different steps and tools included in CATCh reference and *de novo*. 95

Figure 4:2. Illustration of the effect of chimeric sequences divergence on the sensitivity of various reference-based chimera detection tools. 104

Figure 4:3. Plots illustrating the robustness of each tool (reference and *de novo*) to changing their (scaled) cut-off parameters 109

Figure 5:1. Different levels of conservation within the 16S rRNA gene compared between all bacterial and archaeal families. 129

Figure 5:2. Heatmap illustrating the percentage of retrieved OTUs upon applying the various cut-offs, thereby using various combinations of variable regions 131

Figure 5:3. Illustration of the proposed lookup table approach for defining a dynamic threshold for OTU clustering of 16S rRNA gene sequencing reads 134

Figure 5:4. Plot illustrating the correspondence between the number of OTUs derived based on different amplicons (V1-V3, V3-V5, and V6-V9) as calculated for the HMP samples. 138

Figure 6:1. Overview of the different steps within each pipeline. 153

Figure 6:2. Average number of reads removed within the various pipelines, due to improper assembly, quality filtering or chimera removal. 158

Figure 6:3. Rarefaction curves of the different samples 161

Figure 6:4. Composition of the OTUs produced via the various approaches 162

LIST OF TABLES

Table 1:1. Comparison between high throughput sequencing technologies 20

Table 2:1. Benchmarking of different denoising algorithms using the MOCK1 dataset. 52

Table 2:2. Tabular overview of the computational cost of the different denoising algorithms. . 53

Table 2:3. OTUs produced after treating the data with different noise removal approaches. 56

Table 3:1. Overview of the mock sequencing data discussed in this work. 67

Table 3:2. Overview table comparing error rates of the samples various preprocessing 76

Table 4:1. Detailed description of each of the datasets used in this work either for training or testing 100

Table 4:2. Illustration of the performance difference between CATCh (default score) and each individual tool after applying a cut-off which results in the same specificity 106

Table 4:3. Sensitivity and specificity values of all tested chimera prediction tools. 110

Table 4:4. Execution times of different reference based and *de novo* chimera detection tools . 113

Table 5:1. Illustration the number of Bacteria and Archaea within the LTP database 126

Table 5:2. Illustration of the percentage of over-merged OTUs upon applying both the 97% default cut-off and our proposed lookup table, ... 135

Table 6:1. Overview of the algorithms available for different steps within amplicon sequencing data analysis. 150

Table 6:2. The error rates for the different samples after applying various pipelines 159

LIST OF ABBREVIATIONS

Abbreviation	Meaning
16S	16S ribosomal RNA, a component of prokaryotic ribosomes
MBs	megabases
LTP	Living Tree project
nt	Nucleotide
bp	Base pairs
NIH	National Institutes of Health
NGS	Next Generation Sequencing
PCR	Polymerase Chain Reaction
RDP	Read Distance Profile
RefSeq	Reference Sequence database of the NCBI
SSU	ribosomal small subunit
SRA	Sequence Read Archive of the NCBI.
SMRT	Single Molecule Real Time
ZMWs	Zero-mode wave-guides
CATCh	Combining Algorithms to Track Chimeras
NoDe	Noise Detector
IPED	Illumina MiSeq Paired-End Denoiser
DynamiC	Dynamic Cut-off
OCToPUS	Optimized CATCh, mothur, IPED, UPARSE and SPAdes

LIST OF PUBLICATIONS

Grunert O, Robles A.A, Aguilar R, Reheul D, **Mysara M** et al (2016) *Plants rather than fertilizer drive rhizosphere bacterial community in organic growing medium blended with recovered nutrients* (in preparation).

De Ryck T, Kerckhof F.M, **Mysara M**, Monsieurs P, Duprez F, Vanhoecke B, Van de Wiele T. (2016) *Microbial Biomarkers For Radiation-induced Oral Mucositis In Head And Neck Cancer Patients, Oral Oncology* (submitted).

Mysara M, Njima M, Leys N, Raes J, Monsieurs P. (2016). From Reads to Operational Taxonomic Units: an ensemble processing pipeline for MiSeq amplicon sequencing data, GigaScience (submitted).

Mysara M, Vandamme P, Leys N, Raes J, Monsieurs P. (2016). *Reconciliation between Operational Taxonomic Units and Species Boundaries.* FEMS Microbiology Ecology (submitted).

Props R, Monsieurs P, **Mysara M**, Clement L, Boon N. (2016*). Measuring the Biodiversity of Microbial Communities by Flow Cytometry*. Methods in Ecology and Evolution.

Mysara M, Leys N, Raes J, Monsieurs P. (2016). *IPED: a highly efficient denoising tool for Illumina MiSeq Paired-end 16S rRNA gene amplicon sequencing data.* BMC Bioinformatics 17:192.

Mysara M, Leys N, Raes J, Monsieurs P. (2015). *NoDe: a fast error-correction algorithm for pyrosequencing amplicon reads.* BMC Bioinformatics 16:88.

Mysara M, Saeys Y, Leys N, Raes J, Monsieurs P. (2015). *CATCh, an ensemble classifier for chimera detection in 16S rRNA sequencing studies.* Appl. Environ. Microbiol. 81:1573–84.

Michaud L, Giudice A, **Mysara M**, Monsieurs P, Raffa C, Leys N, et al- *Snow Surface Microbiome on the High Antarctic Plateau (DOME C)*.-PLOS one, 9:8(2014).- ISSN 1932-6203.

Moors H, Cherkouk A., **Mysara M**, Bleyen N., Boven P, Selenska-Pobell S, et al (2013).*BN Experiment: Intermediate results of the microbiological analyses*, Belgium: Mont Terri consortium. 68 p (Technical Note).

Chapter 1

Introduction

1.1. Background of the study

1.1.1. *Prokaryotes*

Prokaryotes represent the most diverse domain in the tree of life (Pace 1997), with estimated population of 5×10^{30}. They can be divided into two domains, Archaea and Bacteria, together occupying a wide range of common environments such as soil, air and seas (Fierer et al. 2007; Schloss & Handelsman 2006; Roesch et al. 2007; Brodie et al. 2007; Alonso-Sáez & Gasol 2007; Frias-Lopez et al. 2008; Sogin et al. 2006; Stevens & Ulloa 2008) and including extreme environments –referred to as extremophiles- as in Arctic, deserts, uranium-contaminated soil, and even the deepest spot on the Earth (Bottos et al. 2008; Fierer & Jackson 2006; Johnson & Cameron 1973; Barns et al. 2007; Nunoura et al. 2015). Altogether they form a significant part of the world's biomass (Whitman et al. 1998). With various symbiotic associations, prokaryotes thrive in complex communities, making the identification of their composition and understanding their diversity (the number of species and their relative abundances) one of the most fundamental questions (Ward 2002).

Bacteria form the largest domain within Prokaryotes, involved in nutritional distribution and degradation of various compounds within the ecosystem. Due to their diverse metabolic capabilities, they can utilize a wide range of energy sources, as such contributing to bioavailability of these sources for other organisms, to the extent that in their absence all life on earth would perish (Newman & Banfield 2002; Whiteman 2008). Within the human body, bacteria are present in the same order of magnitude as the number of human cells, reaching up to 3.9×10^{13} bacterial cells in or on the human body (Sender et al. 2016). They exhibit various functionalities affecting human health, as they are involved in e.g. immune responses, food digestion and production of vitamins (Zhang et al. 2015; Leblanc et al. 2011). They also contribute to the

pathogenicity of various diseases, where – as an example – strong associations have been established between changes in gut microbiota and *Clostridum difficile* infections, inflammatory bowel diseases, type II diabetes, obesity, cardiovascular diseases, and even autoimmune diseases (e.g. rheumatoid arthritis) and mental diseases (e.g. autism), as reviewed in Feng (2015). Apart from that, humans have been exploiting the use of bacteria in various applications, for instance in the preservation, fermentation and flavouring of food (Campbell-Platt 1994; González-Aguilar et al. 2009), the production of chemicals within the industrial and pharmaceutical applications (Demain 2000), the recovery of metal and the production of biofuels (Gerlach 2012).

Several important aspects of Bacteria colonizing various environments are poorly understood, making them the largest unexplored reservoir of biodiversity on Earth. As a result various initiatives have recently been launched to characterize the composition and study the diversity of Bacteria among other microorganisms in wide range of environments, such as: the Human Microbiome Project sponsored by the NIH (HMP, http://nihroadmap.nih.gov/hmp/) and Metagenomics of Human Intestinal Tract sponsored by the EU (MetaHIT, http://www.metahit.eu), both devoted to human-associated microbes studies, the Seagrass Microbiome Project (https://seagrassmicrobiome.org/) for seagrass microbiome studies, the International Census of Marine Microbes (ICoMM, http://icomm.mbl.edu) dedicated to marine microbes, the Tara oceans project (http://www.embl.de/tara-oceans/start/news/index.php) studying organisms within oceans including Bacteria, and Earth Microbiome Project (EMP, http://www.earthmicrobiome.org/) for studying the microbial communities across the globe.

If species designations are not well founded, or phenotype not well circumscribed by the 'species', serious confusion and damage may occur (Konstantinidis et al., 2006).

Researchers are not only concerned with understanding the microbial shifts but also identifying and naming the individual species composing the various communities. Inability to properly characterize those species, could highly impact various aspects such as diagnosis of infections were the same genus (or even species) could include both pathogenic and non-pathogenic strains.

1.1.2. Bacterial species

The species could be described as 'a monophyletic and genomically coherent cluster of individual organisms that show a high degree of overall similarity in many independent characteristics, and is diagnosable by a discriminative phenotypic property' (Rosselló-Mora & Amann 2001).

Species is the unit for characterizing a discrete form of life organized based on the shared characteristics among same members distinguishable from other species. Due to the large species diversity, an urge to name and classify our surroundings has likely started long time ago, which was crystalized into what is named the taxonomy science, concerned with the characterization, classification and nomenclature of those biological entities (Yarza et al. 2014). In the field of microbiology the boundary that distinguishes one species from another is still a matter of debate, to the extent that even the concept of bacterial species is questionable (Hutchinson 1968; Mayr 1982; Brenner et al. 2001; Staley 2009; Connor et al. 2010; Cohan & Perry 2007). Perhaps the crux of this problem, according to Doolittle & Zhaxybayeva (2009), is coupling the theories justifying the formation of a discrete clusters of similar individuals (i.e. the concept of prokaryotic species) with criteria for recognizing such clusters (i.e. prokaryotic species definitions). Characterization of microorganisms is essentially needed to grasp our understanding on how they evolve and diversify over time. That

concern is where multiple disciplines intersect, including microbial systematics, ecology, population genetics, evolution and genomics (Achtman & Wagner 2008).

There is a general agreement within the field of microbiology that all taxonomic information about a bacterium is incorporated within its genomic sequence (Wayne 1988; Stackebrandt et al. 2002). In bacterial systematics, a consensus definition has emerged in 1980s to discriminate bacterial species based on the standardized DNA-DNA hybridization (DDH) experiment. DDH starts with shearing the genomic DNA of both the assayed organism and the reference organism(s), denaturation of the DNA double-strand by increasing the temperature, then allowing the temperature to decrease until annealing of sheared DNA fragments(Auch et al. 2010). A value below 70% in DDH experiment, is used as a gold standard criterion for allocating the tested organism to a different species than the type strain(s) used as reference(s) (Gevers et al. 2005). Other measures have been commonly used to demarcate species such as average nucleotide identities (ANI) of shared genes and the genome-to-genome distance (GGD) (Stackebrandt & Goebel 1994; Vandamme et al. 1996; Richter & Rosselló-Móra 2009; Meier-Kolthoff et al. 2013). An ANI value of 95% corresponds roughly to the traditional definition of a bacterial species (or the 70% DNA-DNA hybridization value). While for GGD, a variety of similarity search methods (such as BLAST) have been established for the analysis of high-scoring segment pairs followed by calculations of genome-to-genome distances mimicking DDH *in silico* (Auch et al. 2010). Species delineated this way comprise phenotypically coherent strains that share a varying percentage of core genes (up to 70%) and that have a unique set of accessory genes that commonly represents up to 30% of their gene content (Lapierre & Gogarten 2009).

One of the major constraints of the gold standard DDH experiment is its dependency on the time-consuming and labour-intensive culturing step. It has been shown that only a very small fraction (< 1%) of bacteria in an environment can be grown under lab conditions, due to complex metabolic requirements, rendering the vast majority of ecological niches inaccessible by these type of approaches (Curtis et al. 2002; Venter et al. 2004; Giovannoni & Stingl 2005; Staley 2006; Koeppel & Wu 2013; Hugenholtz 2002; Torsvik & Øvreås 2002; Amann et al. 1995).

1.1.3. Bacterial species and 16S rRNA gene

Over the past few decades our perspective on microbial diversity has improved vastly using amplicon sequencing approaches, involving a PCR amplification step followed by DNA sequencing. Polymerase chain reactions (PCR, developed in 1983) allow amplifying a specific region of the DNA into thousands to millions of copies utilizing synthetic oligonucleotides named primers (Bartlett & Stirling 2003). Those primers are designed to complementarily attach to a start and end of a region within the DNA – commonly referred to as amplicon- that would be later amplified using the DNA polymerase enzyme.

With the application of PCR amplification and sequencing techniques, it is possible to infer the phylogenetic information via comparing the primary structures of macromolecules as hypothesized by Zuckerkandl & Pauling (1965). Although other macromolecules such as cytochromes and ferredoxins (Fitch & Margoliash 1967) were initially used, the bacterial 16S rRNA gene has been widely used to approximate bacterial species (Fox et al. 1992).

Being involved in the translation of genes into proteins inside cells, the ribosomal small subunits are essential for all living cells granting them functional constancy. The

gene coding for the bacterial 16S rRNA gene exhibits both evolutionarily conserved regions and highly variable structural elements. Basically it is comprised of nine variable regions, that evolved at different rates, and preceded by regions of high conservation, where up to 67% of its bases are paired internally into a total of ~50 helices (Boone et al. 2001), as illustrated in **FIGURE 1:1**.

As early as 1994, an agreement was reached within the field that if two strains share 16S rRNA gene sequence similarity below 97% they are considered as two distinct species (Stackebrandt & Goebel 1994). However, if they exhibit a similarity exceeding this 97% threshold, it is still not guaranteed that they belong to the same species. Yet, it has already been reported that this cut-off underestimates the total microbial diversity (Koeppel & Wu 2013; A. Murat Eren et al. 2013; Pedrós-Alió 2006; Chen et al. 2013), thus the bacterial taxonomists nowadays use a more stringent cut-off (up to 98.65%) to delineate species (Kim et al. 2014; Janda & Abbott 2007; Stackebrandt & Ebers 2006). It should however be stressed that organisms that share more than 98.65% of their full 16S rRNA gene sequences may or may not represent the same species (Fox et al. 1992), so the potential for underestimating bacterial diversity through 16S rRNA amplicon sequencing is intrinsic to the molecule studied and cannot be avoided.

The discriminative power of the 16S rRNA gene is not only associated to the variable regions but also to the more conserved regions (Vinje et al. 2014). This varying degree of sequence conservation together with its length (~ 1,500 nucleotides) allow distinguishing between different taxa (Gutell et al. 1985; Noller 1984; Patel 2001; Ludwig & Klenk 2001) and reconstructing phylogenies resolving ancient to modern lineages, from the domain to the genus or even species level (Ludwig & Klenk 2001; Ludwig & Schleifer 1994; Van de Peer et al. 1996). Additionally, the high level of conservation within the 16S rRNA gene enables one to develop primers for

amplification the 16S rRNA gene, which was first introduced via Böttger (1989). This discovery allowed for the design of universal primers targeting the 16S rRNA gene from a wide range of bacterial species making it possible to detect unculturable bacteria (Greisen et al. 1994; Rådström et al. 1994; Amann et al. 1995).

However, there are some drawbacks linked to the use of the 16S RNA gene as a marker gene. Being a subject of homogenization through gene conversion, these ribosomal subunits may possess multiple copies (termed paralogs) within the same species (Hashimoto et al. 2003). These paralogs may possess a significant degree of sequence divergence, which might result in an overestimation of the number of species as reported in *Clostridium paradoxum* (Rainey et al. 1996) and *Paenibacillus polymyxa* (Nübel et al. 1996). The average number of 16S rRNA gene copies per genome was found to be 2.2 copies on average (ranging from 1-15 copies) and the amount of 16S rRNA gene variants was estimated to be 2.5-fold greater than the number of bacterial species (Pei et al. 2010; Eichorst et al. 2007).

Another drawback, illustrated by (Stackebrandt & Goebel 1994), is the occasional disagreement between the 16S rRNA gene and the DNA-DNA hybridization (DDH) results, where some species can be separated via the latter technique while having highly similar 16S rRNA genes, sometimes to the extent of possessing identical 16S rRNA gene sequences. This might be a consequence of the functional importance of this gene, hindering its structural elements from changing freely, as argued by Ludwig & Klenk (2001). Nonetheless, the 16S rRNA gene is regarded as one of the most convenient and widely used measures for microbial diversity analysis, (Ludwig & Klenk 2001; Cook et al. 2003; Hodkinson & Grice 2015), outcompeting other phylogenetic markers, such as cytochromes and ferredoxins (Woese 1987; Olsen & Woese 1993; Olsen et al. 1994; Ludwig & Schleifer 1994; Ludwig et al. 1998).

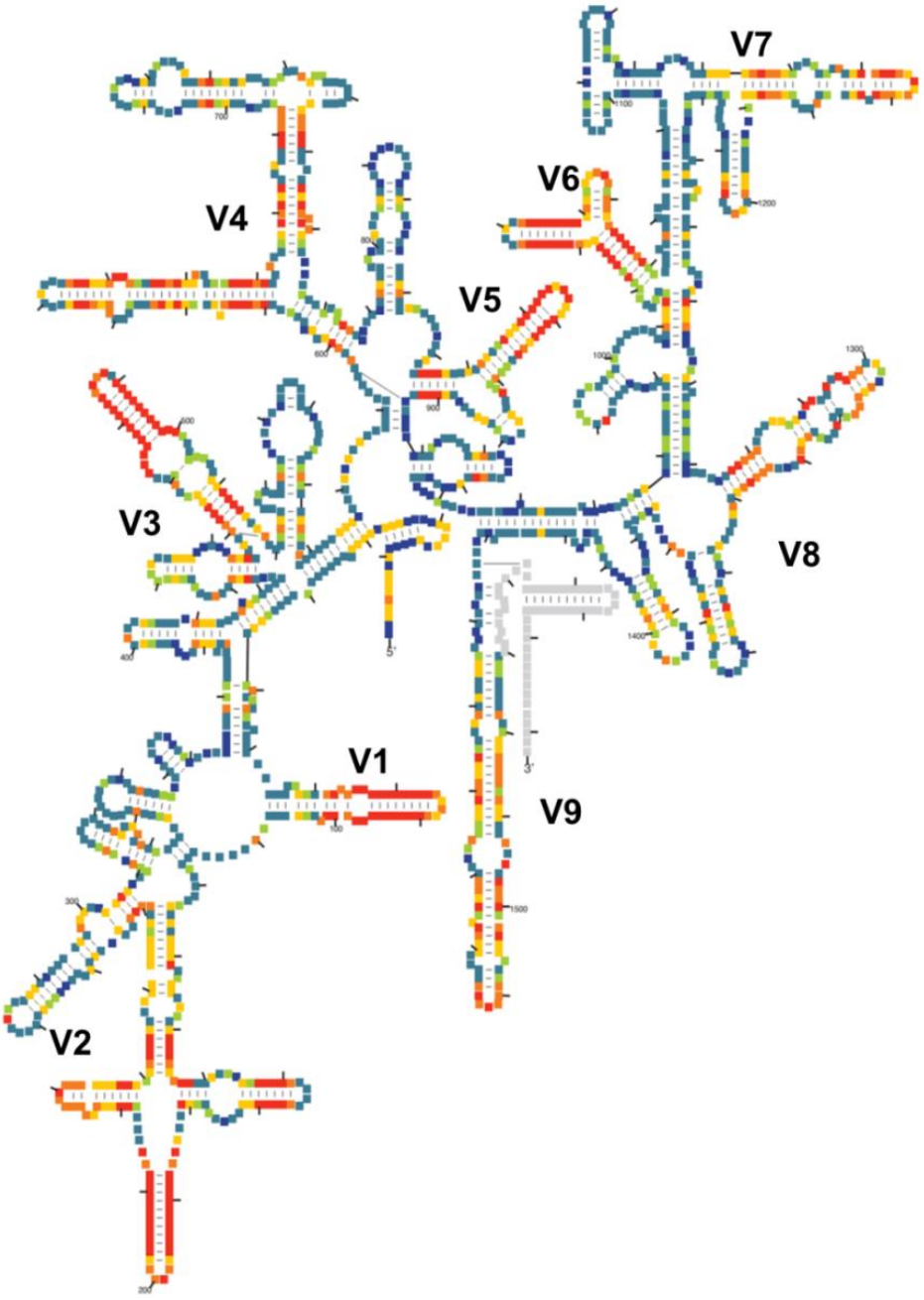


Figure 1:1. Illustration of the secondary structure of the 16S rRNA gene, showing the nine variable regions. First, the consensus sequences were derived for species belonging to each bacterial family within the RDP database (Cole et al. 2014). Next, these consensus sequences were aligned using the SSU-ALIGN algorithm (Nawrocki 2009), and the bit-scores were calculated for each position. Finally, a colour code representation of the bit-score was performed ranging from red to blue, (with red indicating the most variable and blue the most conserved positions).

1.1.4. 16S rRNA gene databases

One of the main advantages of the 16S rRNA gene approach over the DNA-DNA hybridization method, is that gene sequences have been archived in incremental databases, allowing microbiologists to distribute, reanalyse and undertake comparative analysis. As a consequence of relying on the 16S rRNA gene for the taxonomic classification and reconstruction of phylogenetic relationships among prokaryotes, the number of 16S rRNA gene sequences deposited in public databases has increased exponentially during recent years, as shown in **FIGURE 1:2**. 16S rRNA gene sequences are archived and maintained at the National Center for Biotechnology Information (NCBI), accessible via http://www.ncbi.nlm.nih.gov/refseq/. However, due to lack of comprehensive quality validation, other curated databases have emerged such as RDP (Cole et al. 2014), SILVA (Quast et al. 2013), and Greengenes (McDonald et al. 2012).

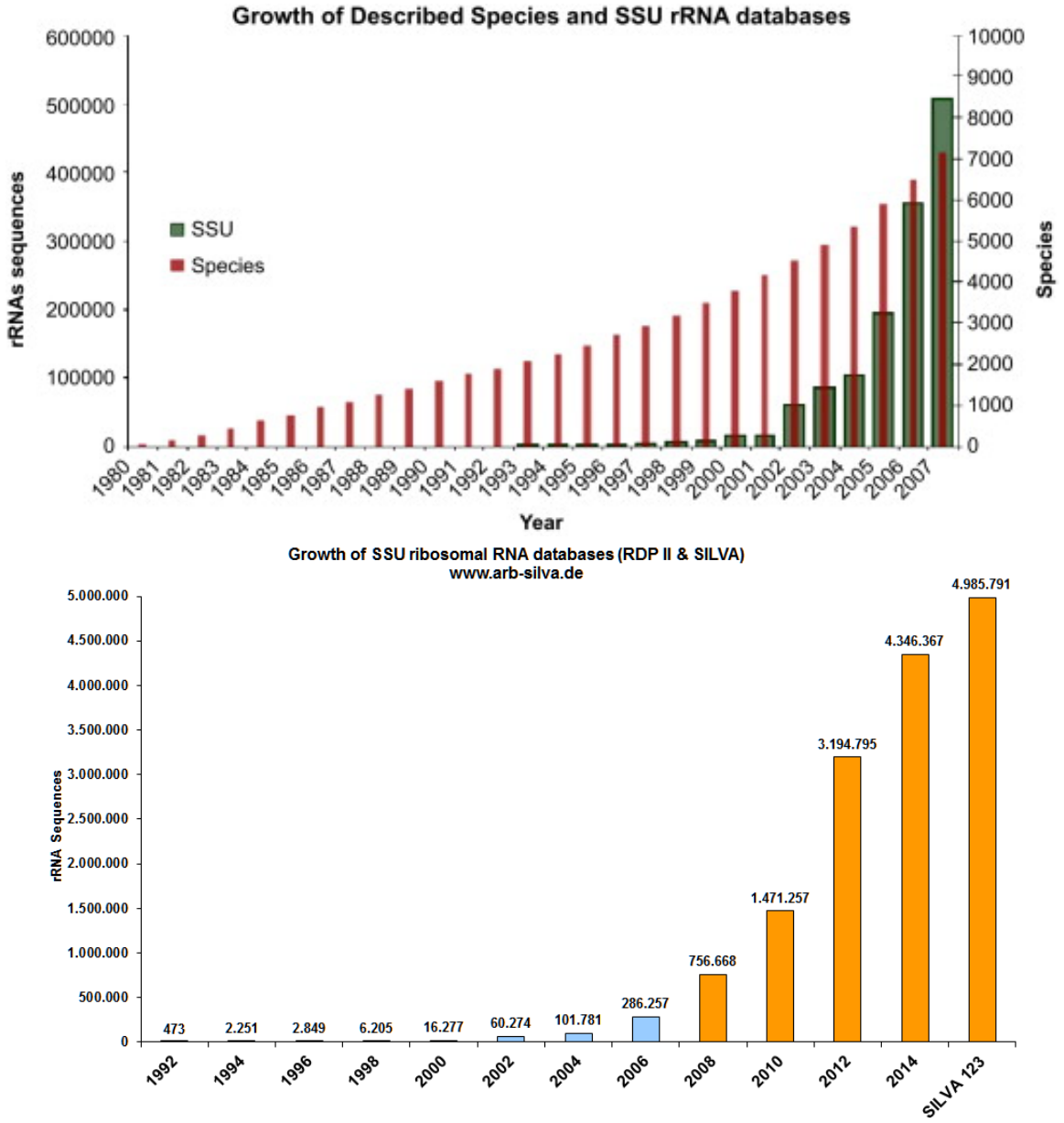


Figure 1:2. Illustration of the growth in the 16S rRNA gene sequences over the years. On top, the number of validated species from 1980 to 2007 in contrast to the SSU sequence submissions to public databases until SILVA release 93 is illustrated (Yarza et al. 2008). At the bottom the number of 16S rRNA gene sequences in the recent SILVA release 123 is shown (source: http://www.arb-silva.de/documentation/release-123/) .

The Ribosomal Database Project (RDP) provides the research community with a database of aligned and annotated small subunits (16S/18S, SSU) and large subunit (28S, LSU) ribosomal RNA sequences along with a related data analysis framework (Cole et al. 2014), available via http://rdp.cme.msu.edu/. SILVA (Quast et al. 2013)

provides comprehensive, quality checked and regularly updated datasets of aligned small (16S/18S, SSU) and large subunit (23S/28S, LSU) ribosomal RNA (rRNA) sequences, available via http://www.arb-silva.de/. Greengenes, (McDonald et al. 2012) is a quality controlled, comprehensive 16S rRNA reference database, available via http://greengenes.secondgenome.com. Several efforts have been made to compare these databases, as shown in **FIGURE 1:3**, and to consolidate between them providing a curated and comprehensive set of (near) complete 16S rRNA gene sequences for phylogenetic and evolutionary analyses, as the 16S ribosomal RNA targeted loci project, hosted by NCBI (BioProject: PRJNA33175).

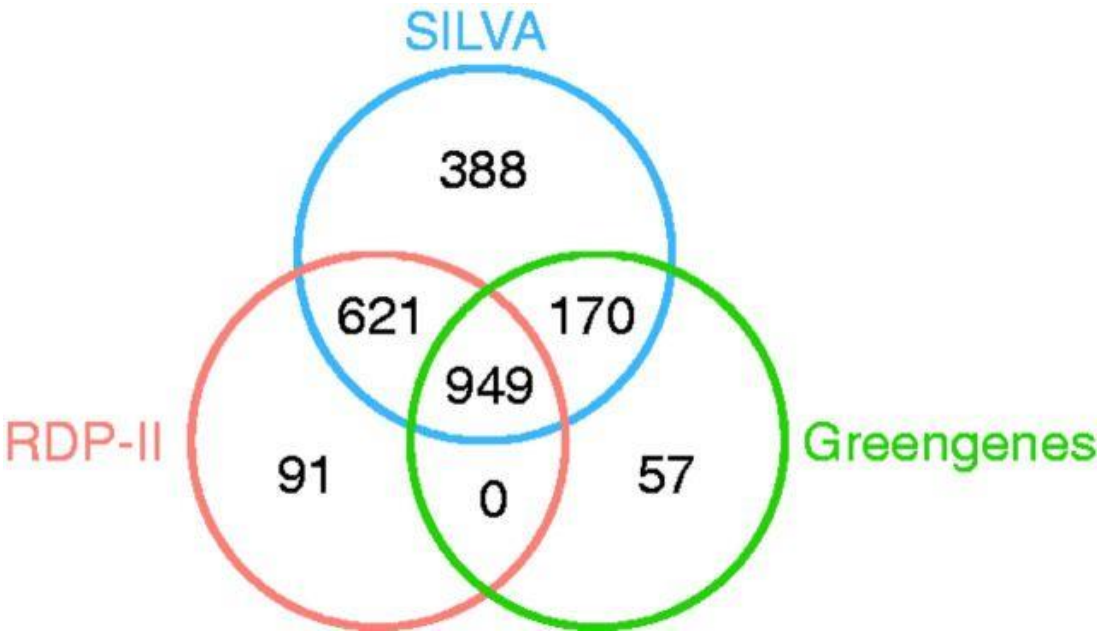


Figure 1:3. Venn diagram showing the number of shared taxa at genus level between SILVA, RDP-II and Greengenes. Only taxa containing sequences from cultivated organisms are included in this comparison. The overlapping part in the middle shows the number of all taxa jointly shared by all three databases; the other overlaps show taxa shared between two databases, but not the third. RDP-II and Greengenes share no other taxa in addition to the 949 shared jointly by all three databases (Yilmaz et al. 2014).

1.2. 16S rRNA gene and the advancement in sequencing technologies

Soon after the identification of the molecular structure of DNA by Watson and Crick (1953), different methods have been developed to determine the precise order of nucleotides within a DNA molecule, referred to as DNA sequencing. The first DNA sequencing technique was introduced by Wu (1970), which was demonstrated to be applicable for any DNA sequence using synthetic location-specific primers. Sanger adapted this approach and proposed the chain-termination approach, commonly referred to as Sanger sequencing (1977), that outperformed the DNA sequencing by chemical degradation as developed by Maxam & Gilbert (1977). From the 1980s until the mid-2000s, Sanger sequencing (read lengths from ~450 to ~900 bp) was the gold standard for DNA sequencing, until the era of high-throughput sequencing (Medini et al. 2008). Nonetheless, over the years advancements in Sanger sequencing are still being pursued to reduce the workload and duration (Chen et al. 2014).

The main limitation of the Sanger sequencing approach for 16S rRNA based metagenomics is the high cost as a result of the time-consuming and labour-intensive cloning step (Medini et al. 2008). Due to the high demand for low-cost DNA sequencing, recent high-throughput sequencing (also named second-generation sequencing or next-generation) technologies have been developed, which are able to produce much more data at a lower cost. An illustration of the various sequencing techniques is shown in **FIGURE 1:4**. An overview of the advantages of these technologies is given in **TABLE 1:1**.

1.2.1. Roche (454 Pyrosequencing)

The first high-throughput sequencing technology was brought on the market in 2005 by Pyrosequencing AB company, that was later licensed to 454 Life Sciences, which was on its turn eventually purchased by Roche in 2007. The initial technique of pyrosequencing was developed by Mostafa Ronaghi and Pal Nyrén (1998), where tiny beads are loaded with DNA fragments, one DNA fragment per bead, which will be later subjected to PCR amplification via emulsion PCR and fixation on a single-bead sized microwells. Each microwell is incubated with the DNA polymerase (DNA synthesis), sulfurylase (conversion of pyrophosphate into ATP), luciferase (generation of light proportional to the amount of ATP) and apyrase (degradation of unincorporated nucleotides and ATP), and with the substrates adenosine 5´ phosphosulfate and luciferin (required for sulfurylase and luciferase activity respectively). Sequencing-by-synthesis starts by the addition of the different nucleotide bases sequentially. Whenever a base is attached to a DNA fragment, it triggers pyrophosphate release and light emission, hence the name pyrosequencing. The light intensities are recorded for each well throughout the addition of the bases allowing the deciphering of the nucleotide order.

The overall sequencing accuracy of 454 pyrosequencing has been improved over the years (Margulies et al. 2005; Droege & Hill 2008), reaching accuracy exceeding 99%. However, this technology always had difficulties identifying the length of homopolymers (a homopolyer is a stretch of the same base e.g., AAA). Additionally, the presence of homopolymers might also cause an insertion or substitution near to it (i.e. carry-forward events). Another shortcoming of the 454 pyrosequencing approach is that the quality tends to drop with the increase of the read length and is occasionally affected by the position of the bead on the plate (edge effect), as reported in Huse et al.

(2007), Gilles et al. (2011), Schloss et al. (2011), Balzer et al. (2011). Noticeable improvement was provided via their most recent GS FLX+ platform, which is capable of producing around 1 million reads per run with high quality reads up to 1,000 bases in length. Due to its long reads and high accuracy, 454 Pyrosequencing has for many years been the preferred approach for microbiological ecologists doing 16S rRNA amplicon sequencing for microbial studies. Although pyrosequencing was the first emerging high-throughput sequencing technology available on the market, 454 Life Sciences will no longer support the platform after mid-2016, thus opening the window for other technologies.

1.2.2. Illumina (MiSeq)

The Solexa platform, developed by Turcatti and colleagues (2008), was bought by Illumina to become one of the most widely used sequencing platforms. The main difference with the 454 pyrosequencing technique is that the former one does not rely on pyrophosphate release, but rather on a reversible dye-termination approach. While both approaches use a sequencing-by-synthesis principle, in Illumina this occurs in a flow cell with a field of oligos attached to it, instead of microwells with beads used by 454 Pyrosequencing. Sequences are extended one nucleotide at a time and imaging is performed at a delayed moment, allowing for very large arrays of DNA colonies to be captured in throughput, which in turn leads to a dramatic reduction of the costs.

The main drawback of the Illumina technology is that it produces rather short reads as such compromising the 16S rRNA analysis discriminatory power. Currently, longer reads can be obtained using the Illumina MiSeq platform (compared to the HiSeq platform), which can produce paired-end reads of 300 bases in length each, and up to 25 million paired reads per run. Similarly to 454 Pyrosequencing, the quality drops with

an increasing length. However, this can partially be overcome via overlapping both the forward and reverse reads, as such making sure that the low quality positions in the DNA-fragment (i.e. the 3' ends) are sequenced twice. These advances regarding throughput and read length, and the announcement of Roche to shut down its 454 pyrosequencing services by the summer of 2016, allowed Illumina becoming the most important sequencing technology for assessing microbial diversity using amplicon sequencing.

Illumina sequencing data do not suffer from indel-errors to the same extent as 454 Pyrosequencing, but rather from nucleotide substitutions (miscalling), mainly originating from two effects: 1) high correlation between A and C as well as G and T intensities due to similar emission spectra of the fluorophores (Kircher et al. 2009; Bentley et al. 2008; Rougemont et al. 2008), and 2) dependency of the signal of each cycle on the signal before and after this cycle, caused by inadequate flushing of fluorophores, incomplete removal of the 3' terminators, or integration of nucleotides without effective 3' terminators (Kircher et al. 2009), known as phasing and pre-phasing. Additionally, it has been shown that such substitutions are often linked to the presence of the GGC motif, or more in general to GC rich motifs, of the amplified region (Allhoff et al. 2013; Tsai et al. 2014; Schirmer et al. 2015).

1.2.3. Ion Torrent (PGM)

The Ion Torrent system is procedurally similar to 454 pyrosequencing with DNA fragments attached to beads via emulsion PCR that are subsequently trapped within a microwell for sequencing. However, opposed to the optical detectors used by 454 pyrosequencing, Illumina and SMRT sequencing platforms, the Ion Torrent system uses a semiconductor based detection system, which detects the release of a hydrogen ion,

hence the name of the technology. The release of this hydrogen ion will result in a change in the pH, indicating that a reaction has occurred and consequently reports the base used for that cycle.

This results in a very efficient cost-effective sequencing technology with read lengths reaching up to 400 bases and a throughput of around 25 million reads (Hodkinson & Grice 2015). However, the quality drops with increasing nucleotide position within the read and the reads still suffer from indel sequencing errors, similarly to 454 pyrosequencing, mainly due to the presence of homopolymers.

1.2.4. PacBio (SMRT)

Unlike previously described sequencing platforms, the Single Molecule Real Time (SMRT) sequencing approach developed by Pacific Bioscience (PacBio) renounces the need for an internal amplification step. SMRT applies a sequencing by synthesis approach in a microwell with the presence of fluorescently labelled nucleotides and their detectors named zero-mode wave-guides (ZMWs) developed by Levene et al. (2003). Whenever a nucleotide is incorporated the detectors recognize the fluorescent signal, and identify the base according to the corresponding dye fluorescence.

Although the read length is remarkably high (exceeding 8,000 bases with the P6-C4 chemistry), the single pass error rate is approximately 15% having both substitutions and indels errors (Schloss et al. 2016). However, the company has incorporated the circular consensus sequencing (CCS) technique, that circularizes the DNA fragments allowing repetitive sequencing of the same fragment (Au et al. 2012), resulting in a reduced error rate that could (repeatedly) cover the full length of the 16S rRNA gene. In contrast to the 454 Pyrosequencing and Illumina MiSeq platforms where the

sequencing quality decays towards the 3’ end of the read, the consensus based approach of PacBio generates a uniform distribution of errors.

1.2.5. NanoPore (MinION)

Another platform that is currently finding its way to the application area of amplicon sequencing is the NanoPore platform, introduced via Oxford NanoPore Technologies. The company offers various platforms such as the MinION, a portable pocket-size device for real-time biological analysis, SmidgION, an even smaller device currently under development to be used with mobile devices, and PromethION, with increased throughput allowing for the analysis of different samples concurrently. The eight nanometer pores are capable of identifying the DNA nucleotides, via detecting changes in the ionic current as they pass. This approach allowed for the generation of thousands of reads that are up to thousands of nucleotides in length (Benítez-Páez et al. 2016). Although predicted to reach an error rate of 0.2% (Rusk 2009), extra improvement of the chemistry is needed to increase the reliability for amplicon sequencing.

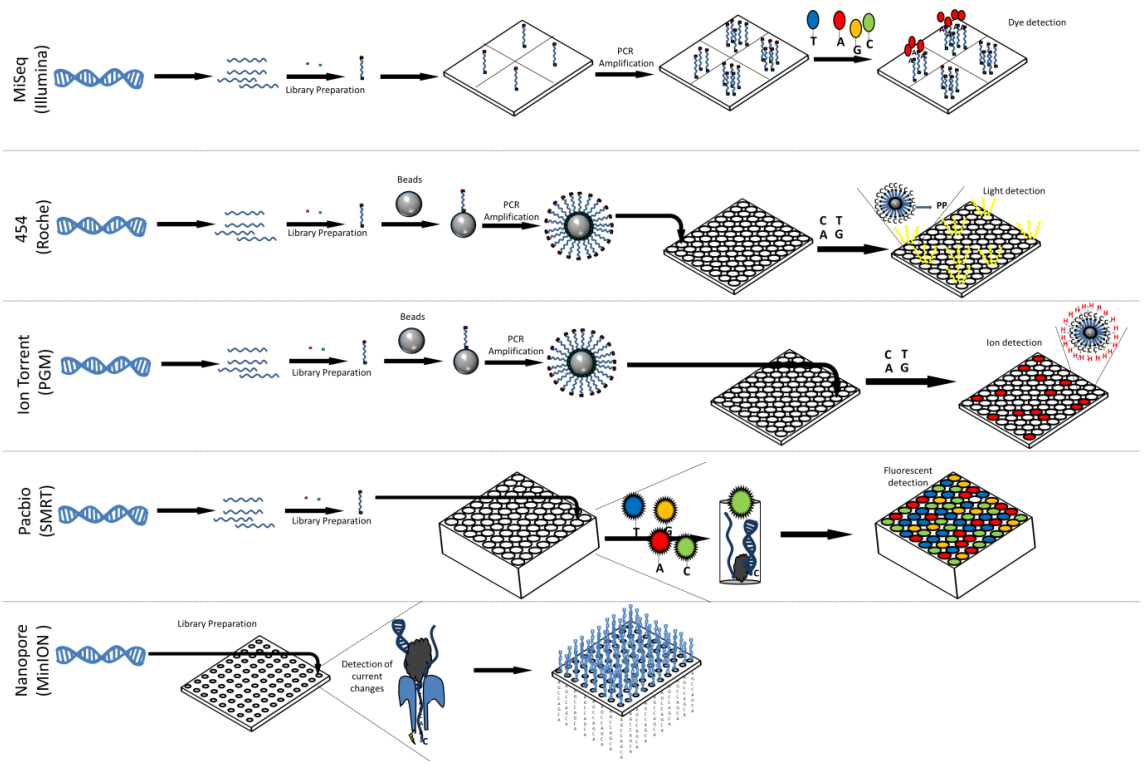


Figure 1:4. Illustration of the various steps performed for Illumina, Roche, Ion torrent, PacBio and NanoPore sequencing platforms.

Other sequencing techniques exist but were not as popular as the ones described above for 16S rRNA amplicon sequencing analysis. Therefore they are not described in this work. It is worth mentioning that all sequencing techniques were intended for genome sequencing, where various reads – that harbour sequencing errors – would be eventually assembled into a high quality consensus. However, as this assembling approach is not used for amplicon sequencing analysis, only those sequencing techniques producing long enough high quality reads can be used.

1.3. 16S rRNA amplicon sequencing

1.3.1. High-throughput sequencing of 16S rRNA gene

16S rRNA amplicon sequencing, or what is referred to as 16S rRNA metagenomics or less commonly as 16S rRNA metagenetics (Delhalle et al. 2016), is a type of amplicon sequencing analysis that involves the amplification of the 16S rRNA genes (or a part of them) of uncultured microorganisms and sequence them via a high-throughput sequencing approach to analyse bacterial communities. In 16S rRNA amplicon sequencing the DNA is extracted directly from heterogeneous microbial communities providing – to a great extent – a full picture on the microbial composition within these communities.

Table 1:1. Comparison between high throughput sequencing technologies

Platform	Sequencing chemistry	Read length (nt)/ version	Raw error rate	Run time	Max. data (per run)
454 Pyrosequencing (Roche)	pyrosequencing	400-600 (FLX)	1.06%	10 hrs	450
		Up to 1000 (FLX+)	NA	23 hrs	700 Mb
MiSeq (Illumina)	Reversible terminator	250 x 2 (V2)	0.92%	4-39 hrs	8.5 Gb
		300 x 2 (V3)	NA	4–55 hrs	15 Gb
PGM (Ion torrent)	Ion semiconductor	200-400 (PGM)	1.47%	4-7 hrs	2 GB
MinION (NanoPore)	real-time	Up to 50 kb (MinION)	NA	NA	NA
SMRT (PacBio)	real-time	~ 1000 (P6-C4) [eight passes]	1.11%	15- 240 mins	5-10 Gb

Information derived from companies websites, (D'Amore et al. 2016) & (Benítez-Páez et al. 2016).

Historically, the most commonly used high-throughput sequencing technology for 16S rRNA amplicon sequencing has been the Roche 454 Pyrosequencing (Salipante et al. 2014; D'Amore et al. 2016). However, with the recent advancements in other high

throughput sequencing – with respect to the length, throughput and cost – allowed for other technologies to be implemented such as the most commonly used Illumina MiSeq (e.g., Gloor et al. 2010; Kozich et al. 2013), and to a lesser extent Ion Torrent (e.g. Jünemann et al. 2012; Milani et al. 2013; Yergeau et al. 2012), PacBio (e.g. Fichot & Norman 2013; Mosher et al. 2013; Mosher et al. 2014; Singer et al. 2016; Schloss et al. 2015; Schloss et al. 2016) and NanoPore (Kilianski et al. 2015; Benítez-Páez et al. 2016).

A few comparative studies between various sequencing approaches where conducted in the field of 16S rRNA amplicon sequencing, (Salipante et al. 2014; Schloss et al. 2016; D'Amore et al. 2016), suggesting the benefits of Illumina MiSeq and 454 Pyrosequencing in respect to the error rate, PacBio and 454 Pyrosequencing in respect to read length, and Ion torrent and Illumina MiSeq in respect to the throughput/cost (NanoPore not included). An overview of the advantages of these technologies is given in **TABLE 1:1**. Despite the fact that current single molecule sequencing technologies like PacBio and NanoPore are not yet producing high-quality data to the same extent as the Illumina MiSeq platform, it is generally believed that those technologies will become inevitable in the field due to the combination of their single-molecule character with the increased read length.

1.3.2. Problems with data analysis

We are drowning in information and starving for knowledge, Rutherford D. Rogers (Branin & Case 1998).

Deciphering the sequencing data into useful knowledge and solid conclusions is of utter importance as it helps understanding the underlying microbial diversity. As illustrated earlier, there are fundamental differences between the various sequencing

approaches, their advantages and disadvantages, which pose bioinformatics challenges to handle these differences and resolve their disadvantages.

Various problems have been known to accompany 16S rRNA gene amplicon sequencing analysis, either related to the use of high-throughput sequencing, PCR amplification or the marker gene selection (i.e. 16S rRNA gene). A first source of error originates from the use of PCR amplification step required to amplify the targeted region within the 16S rRNA gene. Chimera formation within the PCR amplification step is one of the major problems. It originates due to incomplete extension of the DNA fragment which acts as a primer in the next round of the PCR (by binding to the template DNA of a different species). As a result a chimeric sequence is consisting of two or more fragments amplified from the DNA of distinct species (Ashelford et al. 2005; Odelberg et al. 1995; Judo et al. 1998; Smyth et al. 2010). As those chimeras will propagate in the same way as any other DNA sequence, they can take up to 30-40% of all unique sequencing reads (Wang & Wang 1996; Wang & Wang 1997; Porazinska et al. 2012). As such, falling short in the removal of these artificial sequences will have a significant impact on the diversity estimates, since chimeras that go undetected will be interpreted as novel species (Wang & Wang 1996; Wang & Wang 1997).

Another type of error originating from the usage of the PCR polymerase enzymes is their tendency to incorporate incorrect bases. This step has been estimated to have an error rate of 0.0001% to 0.001% (Cline et al. 1996). These mistakes would propagate through the PCR cycles, resulting in nucleotide-specific spikes of the error rate at specific positions as argued by Schirmer et al. (2015). Although the necessity for an internal PCR amplification step is phasing out within the newer sequencing technologies (PacBio and NanoPore), it is still unavoidable to specifically target the desired region within the genome.

A second source of errors originates from DNA sequencing platforms. The high-throughput character of these platforms comes at the cost of a decreased accuracy, as such posing important challenges at the level of data analysis. As a general rule, the sequencing quality drops with the increase of the length with the sequencing by synthesis approaches. However, other factors contribute to the sequencing approaches, as discussed above, increasing the vulnerability of Roche and Ion torrent platforms to indel errors, and Illumina platform to substitutions errors, while both indels and substitutions errors are frequently occurring in PacBio. For Illumina MiSeq an extra approach has been proposed to obtain reads with lower error rate. Via allowing both forward and reverse reads to be partially/completely overlapping, it is possible to confirm the base-calling prediction using both reads generating a higher quality consensus amplicon (Kozich et al. 2013). Yet, this overlapping region spans those parts of the reads with the lowest quality scores. Such practice often fails to resolve conflicts between both reads which contribute greatly to the error rate. For PacBio, the error rate can be dramatically reduced via increasing the coverage – i.e. reading the same region several times – which was found to reduce the error rate by an order of magnitude to approximately 1.5% (Schloss et al. 2016). Nonetheless, as the sequencing technology was not mainly intended for amplicon sequencing, but rather genome sequencing with the possibility to build a consensus sequence with much lower error rate, the level of error rate is considered high, i.e. between 1-2% (Schloss et al. 2016; Salipante et al. 2014; D'Amore et al. 2016). These errors tend to accumulate in specific reads, resulting in an inflation of the OTUs (Schloss et al. 2011; Kozich et al. 2013; Schloss et al. 2016). It was illustrated by D'Amore et al. (2016), that around 4% of the reads of Illumina MiSeq, 454 Pyrosequencing and PacBio suffer from accumulation of the sequencing errors exceeding 5% of the positions (also Ion torrent to a greater extent), thus resulting in the generation of spurious OTUs and an inflation of the number of

OTUs. This will affect the accuracy of the diversity estimates and emphasises the need for bioinformatics interventions.

The ultimate goal for this bioinformatics analysis is clustering the sequencing reads into operational taxonomic units (OTUs), which should resemble the microbial composition. Ideally each OTU should represent a bacterial species within the sample, and the OTU count should give an indication on the relative abundance of this species within the sample. Various analyses have been done with artificial samples, commonly referred to as mock communities, where the exact composition of the community is known (see Additional Information 1:1), in order to evaluate the sequencing approach and library preparation (Schirmer et al. 2015; Kozich et al. 2013; D'Amore et al. 2016; Schloss et al. 2011; Schloss et al. 2016), in addition to comparing the various bioinformatics algorithms. It has been shown that the previously described problems account for a noticeable inflation in the number of OTUs, and jeopardize the accurate assessment of the diversity parameters within a microbial sample.

1.4. Bioinformatics intervention

The analysis from sequencing reads to informative biological data is far from straightforward, requiring the intervention of various bioinformatics approaches. Different steps need to be followed for accurate interpretation of 16S rRNA gene amplicon sequencing data, and they can be classified into three phases, *Pre-processing*, *Processing* and *Post-processing*, see **FIGURE 1:5**.

1.4.1. Pre-processing

In the *Pre-processing* step, the raw data produced from the various sequencing techniques are being processed with the aim of resolving the sequencing errors and chimeras. As one 16S rRNA amplicon sequencing run is normally producing thousands

(or even millions) of reads (e.g. on the Illumina MiSeq platform), such a run is normally split over various biological samples via the use of barcodes or MIDs to acquire reasonable sequencing depth and reduce the cost per sample. Demultiplexing is therefore often the first step applied to the raw sequencing data, splitting those data into various samples based on the barcode/MID sequences assigned to each sample. In case of Illumina MiSeq, both the forward and reverse reads are assembled together into one consensus sequence, using tools as FLASH (Magoč & Salzberg 2011) PANDAseq (Masella et al. 2012), COPE (Liu et al. 2012) and PEAR (Zhang et al. 2014).

Next, the sequences are quality filtered, to identify those reads with poor quality or sequencing anomalies. The Phred score (Ewing et al. 1998) is a commonly accepted measure of base calling quality, which was found very useful in allocating drops in the read quality and consequently truncating the reads. Others have identified various signs of poor quality such as homopolymer stretches (exceeding 8 bases), ambiguous bases (unclear base-calling thus assigned as "N"), or sequences that are abnormally short or long (Schloss et al. 2011; Kozich et al. 2013; Schloss et al. 2016; Salipante et al. 2014).

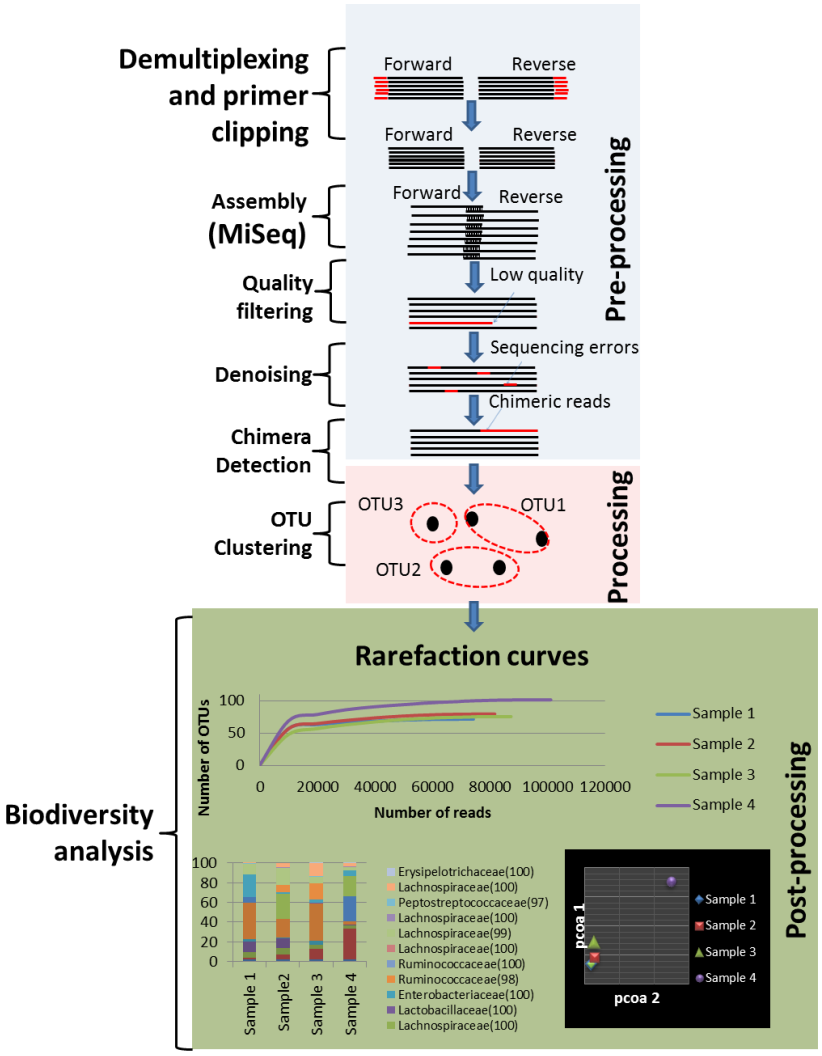


Figure 1:5. Overview of the various steps involved in amplicon sequencing data analysis, and three phases included: Pre-processing, Processing and Post-processing analysis.

Although these measures, among others, where found useful in identifying reads with major sequencing anomalies, they are unable to detect (and in a later stage fix) the remainder of reads harbouring base calling errors. For that various tools have emerged, named denoising tools. As the name suggest, they try to correct the noise (sequencing errors) produced by the sequencing platforms. Some tools have been developed with a general application (for any sequencing platform) while others incorporate platform specific measures (more adapted to the characteristics of each platform e.g. the homopolymer problem for 454 platforms). As a general principle, via clustering the

reads (taking into account their sequencing quality) across each sample, it is possible to calculate the probability of a position to be erroneous.

As for the chimera detection, a plethora of tools has been developed to detect them, either via a reference based approach such as Pintail (Ashelford et al. 2005), Bellerophon (Huber et al. 2004), ChimeraSlayer (Haas et al. 2011), and DECIPHER (Wright et al. 2012), a *de novo* based method such as Perseus (Quince et al. 2011) or a method capable of using both approaches such as UCHIME (Edgar et al. 2011). The general principle of these tools is that each read is compared to a certain dataset to check whether it is similar to a single sequence or two (or more) sequences within that dataset. Two types of approaches can be applied. The first approach uses a reference dataset consisting of presumably chimera free reads, such as the GOLD dataset described in Haas et al. (2011), which is referred to as a reference based approach. The second type of datasets is comprised of reads with abundances higher than the read in question, a methodology referred to as *de novo* approach. It is generally based on the fact that parents of any chimeric sequence have gone through at least one PCR cycle more than chimeric sequences, as such providing them with a higher abundance level..

Depending on the software being used, other interventions can be applied in the *Pre-processing* step, such as the elimination of taxonomic abnormalities (e.g., eukaryotic DNA) and the application of multiple sequence alignment (MSA). Using MSA (Schloss 2009) it is possible to identify anomalies such as PCR primer miss-location and non-16S rRNA gene contaminants within the sequencing reads, as proposed for 454 Pyrosequencing, Illumina MiSeq and PacBio data analysis respectively in (Schloss et al. 2011; Kozich et al. 2013; Schloss et al. 2016). In addition, this allows for fine tuning of the distance calculation step (required for OTU-clustering, see *Processing*

step) via the incorporation of secondary structure information into the alignment and the proceeding distance calculation as argued by Schloss (2013).

1.4.2. Processing

In the most straightforward approach for amplicon sequencing, the reads resulting from the high-throughput sequencing platforms are classified based on their sequence similarity to a reference taxonomic dataset, a methodology most often referred to as phylotyping. However, such a binning procedure will be biased towards the existing taxonomic classification – inheriting its classification errors – which is largely based on cultivable organisms (Amann et al. 1995; Rosselló-Móra 2012; Wang et al. 2007). Therefore, the most widely used bioinformatics pipelines like mothur (Schloss et al. 2009), QIIME (Caporaso et al. 2010) and UPARSE (Edgar 2013) implemented a second approach, where all sequencing reads are grouped together based on their sequence similarity to each other, resulting in clusters of reads used to approximate microbial taxa (Schmidt et al. 2015). Such clusters are often referred to as Operational Taxonomic Units (OTUs).

Several methods have been developed for clustering 16S rRNA amplicon sequencing data, such as *hierarchical clustering* algorithms, e.g., *DOTUR* (Schloss & Handelsman 2005), *mothur* (Schloss et al. 2009), *ESPRIT* (Sun et al. 2009) and *HPC-CLUST* (Matias Rodrigues & von Mering 2014) which compute the divergence between all pairs of reads ending with a distance matrix that is used to generate OTU clusters. Their greedy heuristic approximations such as *UCLUST* (Edgar 2010), *UPARSE* (Edgar 2013) and *CD-HIT* (Fu et al. 2012), require fewer pairwise comparisons thus lesser computational time to estimate the OTU clusters. Various efforts have been made to compare the different clustering algorithms (Schloss & Westcott 2011; Westcott & Schloss 2015;

Edgar 2013; Sun et al. 2012). The final goal of the *Processing* step is to calculate the OTU table, reporting the frequency each OTU for every single sample. This OTU table can later be used to estimate the diversity between different samples to answer the biological question at hand. In addition to the OTU table, the representative sequences of each OTU can be inferred, which can be used to assign their taxonomic classification for example via the usage of the naïve Bayesian classifier (Wang et al. 2007).

1.4.3. Post-processing

Two main biodiversity estimates can be informed from 16S rRNA amplicon sequencing data, alpha and beta diversity, initially introduced by Whittaker (1972). Alpha diversity consists of two components: species richness (how many different species are present) and species evenness (how close are the relative abundances of those species) within a sample. As such, a high diversity within a sample can be obtained by having a high number of species with similar abundances. Various richness and diversity estimators have been applied such as Chao1, inverse Simpson, Shannon and rarefaction curves as discussed in (Magurran 2004; Schloss et al. 2011; Kuczynski et al. 2011). Beta diversity, on the other hand, measures the turnover of species between various samples in terms of gain or loss of species. A wide range of beta diversity indices are available such as Chi-square, bray Curtis, jaccard. Additionally, various visualisation methods have been implemented to compare the microbial composition of the various samples, such as heatmaps (with a visual representation indicating the relative abundance of each OTU) or Venn diagrams (illustrating the overlapping OTUs between different samples versus the unique OTUs for others).

Dendrograms can be applied to describe the similarities of the samples to each other in a tree like diagram, where statistical significance of the clusters within the tree can be assessed for instance using unifrac (Lozupone & Knight 2005). Another popular way of visualizing beta diversity results is through ordination plots such as Principal Coordinates Analysis (PCoA) and Principal Component Analysis (PCA) implementing an eigenvector-based approach to represent multidimensional data into fewer dimensions (Borg & Groenen 2005). Relationships can be established between the variation within the samples on the one side and experimental conditions on the other side, which can later be statistically evaluated via approaches such as AMOVA (Excoffier et al. 1992). An illustration of the various alpha and beta diversity analysis is given in **FIGURE 1:6**.

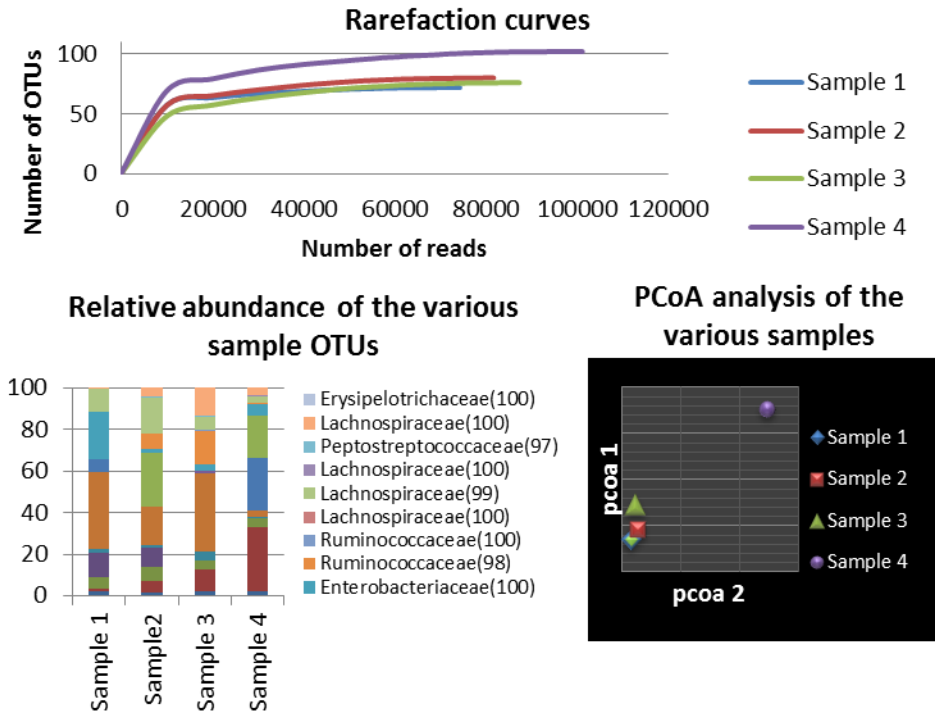


Figure 1:6. Demonstration of the various alpha and beta diversity analysis of four samples. Where sample 4 shows higher richness and diversity (illustrated by rarefaction curves), in addition to being more divergent to the remainder three samples (illustrated by the PCoA, barplots, and heatmap).

1.5. Thesis outline

This PhD work describes the application of 16S rRNA gene amplicon sequencing for the assessment microbial diversity (referred to as 16S metagenomics or 16S metagenetics). Various sources of errors have been accompanying this type of analysis, where obtaining error-free results remains a challenging task. The primary aim of this PhD work is to increase the accuracy of the current practices for 16S rRNA amplicon sequencing data analysis via proposing novel approaches for fine-tuning the *Pre-processing* and *Processing* steps. The main contribution of this work is addressed in chapter 2 until Chapter 6. Chapter 7 provides a general discussion and overall conclusion of the work, as well as perspectives for future work. A schematic illustration of the various chapters is shown in **FIGURE 1:7**.

Chapter 2 describes a denoising algorithm that segregates sequencing errors from actual biological variation. The algorithm presented, named NoDe, is trained to identify 454 Pyrosequencing errors (the most commonly used platform in the last decade), and correct them. It has been compared with existing algorithms in respect to the error rate, throughput, number of OTUs and computational cost. Chapter 2 is based on the following publication:

> Mysara M, Leys N, Raes J, Monsieurs P. (2015). *NoDe: a fast error-correction algorithm for Pyrosequencing amplicon reads.* BMC Bioinformatics 16:88.

Chapter 3 describes a denoising algorithm developed with integral understanding of the fundamental differences of errors associated with Illumina MiSeq platforms (the currently most used sequencing technology). A denoising algorithm named IPED is introduced and subsequently compared with existing algorithms. Chapter 3 is based on the following publication:

> Mysara M, Leys N, Raes J, Monsieurs P. (2016). *IPED: a highly efficient denoising tool for Illumina MiSeq Paired-end 16S rRNA gene amplicon sequencing data.* BMC Bioinformatics 17:192.

Chapter 4 discusses the chimera detection problem. It compares various tools which have been developed to remove chimeric reads, using either a reference-based or a *de novo* approach. It presents a robust methodology able to improve upon the existing practices. In this chapter, a novel tool – named CATCh – is introduced and compared thoroughly with existing algorithms. This chapter is based on the publication:

> Mysara M, Saeys Y, Leys N, Raes J, Monsieurs P. (2015). *CATCh, an ensemble classifier for chimera detection in 16S rRNA sequencing studies.* Appl. Environ. Microbiol. 81:1573–84.

Chapter 5 discusses the limitation of the short lengths of the fragments currently produced by most sequencing technologies. Furthermore, it addresses the bias of applying the 97% cut-off on those short reads and how this influences the interpretation of the OTU clustering results. In this chapter, a novel approach is presented using a dynamic threshold instead of a fixed one, depending on the taxonomic classification and the particular region being amplified within the 16S rRNA gene. This chapter is now under consideration for publication in FEMS Microbiology Ecology:

> Mysara M, Vandamme P, Leys N, Raes J, Monsieurs P. (2016). *Reconciliation between Operational Taxonomic Units and Species Boundaries,* submitted.

Chapter 6 describes a comparative analysis between different comprehensive pipelines incorporating algorithms for paired-end reads assembly, chimera removal, denoising and OTU clustering. In addition a novel pipeline is introduced combining the tools presented within this PhD work together with other tools into a one-stop software, that is compared with the commonly used pipelines. This chapter is currently under consideration for publication in GigaSciences:

> Mysara M, Njima M, Leys N, Raes J, Monsieurs P. (2016). *From Reads to Operational Taxonomic Units: an ensemble processing pipeline for MiSeq amplicon sequencing data,* submitted.

Chapter 7 provides a general discussion of the outcome of previous chapters and concludes their results. It also provides some guidelines for proper data analysis and demonstrates the future perspectives.

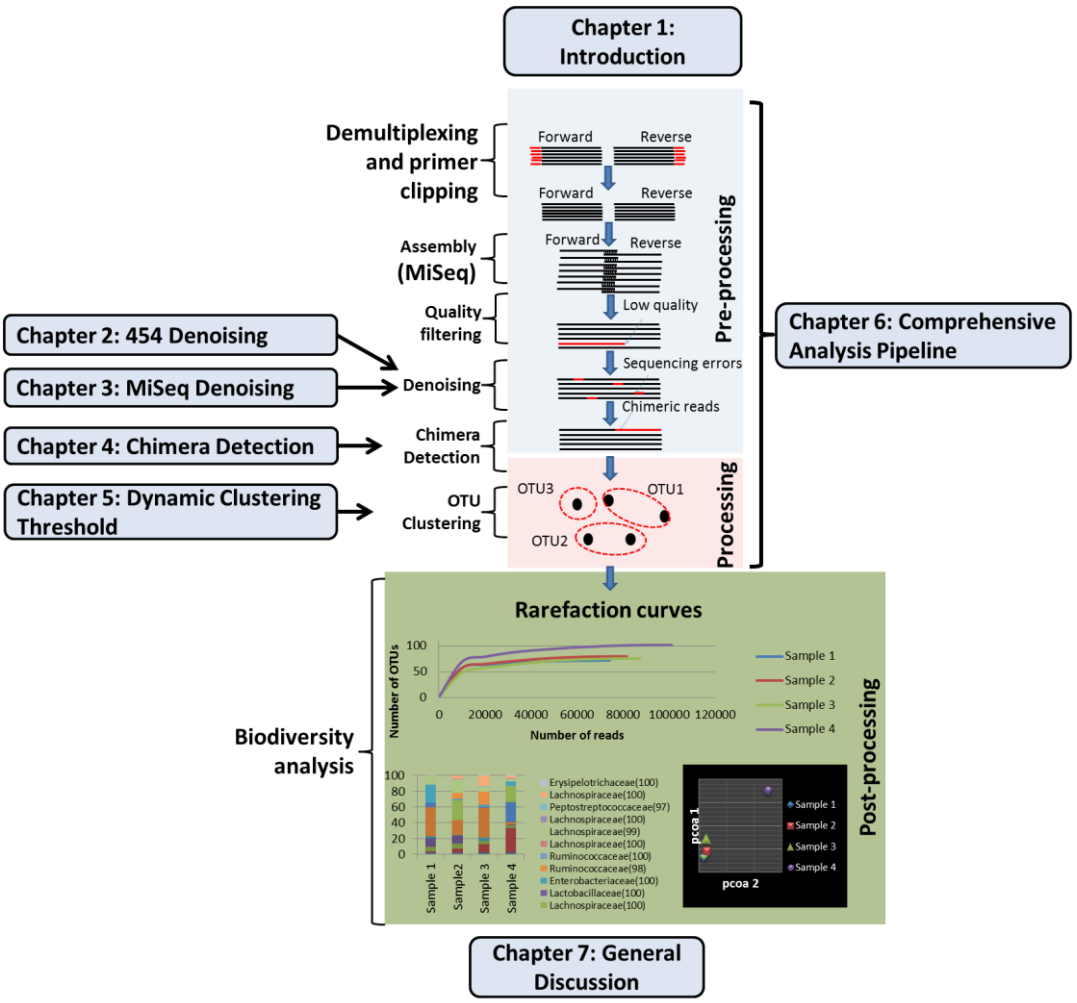


Figure 1:7. Schematic overview of the thesis structure.

Chapter 2

NoDe: a Fast Error-Correction Algorithm for Pyrosequencing Amplicon Reads

Mohamed Mysara[1,2,3], Natalie Leys[1], Jeroen Raes2,[3,4] and Pieter Monsieurs[1]

[1] Unit of Microbiology, Belgian Nuclear Research Centre (SCK-CEN), Mol, Belgium.
[2] Department of Bio-engineering Sciences, Vrije Universiteit Brussel, Brussels, Belgium.
[3] VIB Center for the Biology of Disease, VIB, Leuven, Belgium.
[4] Department of Microbiology and Immunology, REGA institute, KU Leuven, Belgium.

This chapter is based on the following publication:

Mysara M, Leys N, Raes J, Monsieurs P. (2015). NoDe: a fast error-correction algorithm for pyrosequencing amplicon reads. BMC Bioinformatics 16:88.

2.1. Abstract

The popularity of new sequencing technologies has led to an explosion of possible applications, including new approaches in biodiversity studies. However each of these sequencing technologies suffers from sequencing errors originating from different factors. For 16S rRNA amplicon sequencing studies, the 454 Pyrosequencing technology is one of the most frequently used platforms, but sequencing errors still lead to important data analysis issues (e.g. in clustering in taxonomic units and biodiversity estimation). Moreover, retaining a higher portion of the sequencing data by preserving as much of the read length as possible while maintaining the error rate within an acceptable range, will have important consequences at the level of taxonomic precision. The new error correction algorithm proposed in this work - NoDe (Noise Detector) - is trained to identify those positions in 454 sequencing reads that are likely to have an error, and subsequently clusters those error-prone reads with correct reads resulting in error-free representative read. A benchmarking study with other denoising algorithms shows that NoDe can detect up to 75% more errors in a large scale mock community dataset, and this with a low computational cost compared to the second best algorithm considered in this study. The positive effect of NoDe in 16S rRNA gene studies was confirmed by the beneficial effect on the precision of the clustering of 454 Pyrosequencing reads in operational taxonomic units. NoDe was shown to be a computational efficient denoising algorithm for Pyrosequencing reads, producing the lowest error rates in an extensive benchmarking study with other denoising algorithms.

2.2. Introduction

The introduction of next generation sequencing technologies has led to important breakthroughs throughout the life sciences, with applications in *de novo* genome, exome or amplicon sequencing, gene expression analysis, identification of transcription factor binding sites, and so on. Also in clinical and environmental microbial community analysis, 16S rDNA sequencing and metagenomics have been instrumental. For the assessment of microbial community structures based upon 16S rDNA amplicon sequencing, the 454 Pyrosequencing platform has already been in use for many years, mainly due to its longer read length (Huber et al. 2007) nowadays allowing up to 1000 bp reads.

Having access to highly reliable sequencing data is a necessary requirement for biodiversity assessments using 16S rRNA amplicons (Edwards et al. 2006; Sogin et al. 2006) as this approach is highly sensitive to sequencing errors. Indeed, the natural variation in the 16S marker genes for different bacterial species significantly complicates the problem of distinguishing between erroneous sequences on the one side, and sequencing reads representing rare taxa on the other side. This may lead to an overestimation of the number of operational taxonomic units (OTUs) and sample biodiversity as a whole (Huber et al. 2007; Sogin et al. 2006; Kunin et al. 2010). Therefore, error-correction prior to starting biological interpretation of the data is a matter of utmost importance.

Several efforts have been made to study the sources of those errors and how to eliminate or correct them (Schloss et al. 2011; Quince et al. 2009; Schloss et al. 2009; Balzer et al. 2011). There are three major causes of errors in sequencing data at different stages in the sequencing process: i) errors originating from usage of the PCR polymerase enzymes (with error rate of 0.000,001 to 0.000,01 per nucleotide (Cline et al. 1996)), ii) PCR artefacts, known as chimeras (Haas et al. 2011), and iii) errors

originating from the sequencing platform (Schloss et al. 2011). Concerning the latter type of errors, different indicators have already been identified for the GS20 (Huse et al. 2007) and GS-FLX Titanium (Gilles et al. 2011; Schloss et al. 2011; Balzer et al. 2011) platforms: (i) position within the read i.e. the quality of the read is dropping with increasing position in the read, (ii) the presence of homopolymers gives difficulties to identify the correct length of the homopolymer or cause an insertion or substitution near it (i.e. carry-forward events), (iii) abnormality in the read length (i.e. suspiciously short or long) that may be caused by extreme quality filtering, accumulation of errors or stochastic polymerization ending, (iv) position of the bead on the plate (edge effect), (v) distribution of the errors, as errors tend to accumulate in small subset of sequencing reads, meaning that a majority of reads will be error-free or only contain a single position error while a minority of reads is problematic, (vi) nucleotide type, as mismatch transitions are not of equal rates.

Different methods have been developed to enhance the quality of Pyrosequencing reads, starting with the most basic approaches e.g. by removing those reads where no perfect match with the PCR primer could be identified (Dostie et al. 2006). Another approach for correcting sequencing errors was introduced by Huse and co-workers (Huse et al. 2007), including: i) removal of reads with one or more ambiguous bases, ii) reads with a length outside the main distribution or iii) reads containing mismatches in their primer sequence. Additionally, a major improvement was achieved via the use of the read quality scores (i.e. the Phred scores) as proposed in Kunin et al. (2010) by trimming reads from the most upstream position where the quality dropped below the assigned cut-off quality score. Similarly, Schloss et al. (2011) implemented a method where trimming of sequencing reads was based on a drop in the average quality of the read in total or within a specific sliding window.

In addition to methodologies trimming or removing sequencing reads, more sophisticated approaches were made by clustering of the 454 standard flowgrams, to better handle homopolymer related errors, either by applying an expectation maximization (EM) algorithm as in PyroNoise (Quince et al. 2009) or greedy scheme as implemented in Denoiser (Reeder & Knight 2010). Three sequence based clustering algorithm with a much faster computation time were developed quite recently, namely the Single Linkage Pre-clustering method (SLP) (Huse et al. 2010), AmpliconNoise (including an update version of PyroNoise and SeqNoise) (Quince et al. 2011) and Acacia (Bragg et al. 2012). SLP clusters low redundant reads containing errors with the more redundant error-free ones. This is clustering is based on the pairwise distance scores between both reads, thereby tolerating some errors (aiming not to cross the interspecies threshold). AmpliconNoise consists of two steps: i) the PyroNoise algorithm, which applies a clustering approach directly on the flowgrams rather than relying on the quality scores assigned by the 454 Pyrosequencing platform, and ii) SeqNoise, which applies a sequence clustering step.

Acacia (Bragg et al. 2012) is denoising algorithm that aligns each read to dynamically updated cluster consensus sequences, thereby avoiding the time-consuming all-against-all alignments, and is mainly focusing on correcting sequencing errors in homopolymer regions. SLP was re-implemented in mothur (Schloss et al. 2009) as the *pre.cluster* command. In contrast to SLP, *pre.cluster* is applied on the aligned sequences thereby avoiding the computationally intensive all-against-all alignments. Pre-cluster was found to outperform SLP in terms of speed and performance as it avoids overclustering of sequences (i.e. described as the "chaining effect" of SLP (Schloss et al. 2011; Gaspar & Thomas 2013)) since Pre-cluster only groups reads when they are within a maximum distance to the cluster center. These implementations make use of the sequence frequencies to remove errors based on the relative abundance of an error

versus a correct nucleotide. Since SLP and Pre-cluster both work directly on the sequencing data (and not on the flowgrams), those algorithms do not distinguish between PCR point errors and errors produced by the sequencing technology.

When dealing with error-correcting algorithms, researchers need to find a balance between the quantity (in terms of the number of sequences and their average length), and the quality of their Pyrosequencing reads (in terms of the error rates i.e. rates of deletions, insertions or substitutions). Retaining more sequencing data by increasing the average read length, while keeping the error rate within an acceptable range, will have important consequences for taxonomic precision. Another factor that needs to be taken into consideration is the computational cost, which is getting more important with the dramatic increase of the size of sequencing data, resulting in a need for fast and accurate methods to analyse them. Here, we introduce a novel way of artificial intelligence-based prediction of position-dependent erroneous nucleotides in sequencing reads (using a support vector machine trained classifier) and subsequent clustering which will correct error-containing reads in the sequencing data by grouping them with error-free reads (using a modified version of the SLP algorithm), thereby fulfilling the need for quality and speed. This methodology was benchmarked against other state-of-the-art algorithms and multi-step methodologies.

2.3. Materials and methods

2.3.1. Mock communities

In this work, previously published mock datasets were used, as well as a new in-house made mock community. First we used the mock datasets presented in Schloss et al. (2011) as available online (Project "SRP002397" in NCBI Short Read Archive) consisting of 69 samples (21 samples missing in the Short Read Archive), targeting three hypervariable regions in the 16S rRNA gene i.e. V1 to V3, V3 to V5 and V6 to

V9 (total of 888,635 reads. Due to missing data as described above, the sff-files were reconstructed (information provided by P. Schloss, personal communication), and are made available on the NoDe website. One of these samples was randomly selected for training the classifier (accession number F01QS4Z01_rep1_v35, 13,598 reads) and the other 68 samples were used for testing (referred to as TrainingDB and MOCK1 respectively in the remainder of this work). Although the composition of MOCK1 samples was known (20 bacterial species belonging to 18 families), we did not have access to the exact concentrations for each species.

The in-house built mock community (called MOCK2), consists of 17 bacterial species, namely *Acidovorax facilis* (DSM 649), *Pseudomonas xanthomarina* (KMM 1447), *Pseudomonas aeruginosa* (DSM 50071), *Paracoccus denitrificans* (NBRC 102528), *Rhodospirillum rubrum* (ATCC 11170), *Microbacterium phyllosphaerae (P 369/06)*, *Arthrobacter oryzae* (KV-651), *Delftia tsuruhatensis* (NBRC 16741), *Nitrosomonas europaea* (ATCC 25978), *Cupriavidus metallidurans* (CH34), *Clostridium botulinum* (ELTDK 103), *Staphylococcus aureus* (NBRC 100910), *Bacillus cereus (ATCC 14579)*, *Arthrospira platensis* (PCC 7345), *Enterococcus faecium* (NBRC 100486), *Yersinia enterocolitica* (Y11), and *Desulfovibrio oxamicus* (DSM 1925) ordered by their concentrations in descending order.

The DNA was extracted from the individual cultures, mixed, and PCR amplified (30 cycles). Next, the DNA mixture was sequenced in triplicate using the 454 GS-FLX Titanium sequencing platform, covering the region V1-V3 (primer pair AGAGTTTGATCCTGGCTCAG and TTACCGCGGCTGCTGGCAC. Creating an uneven mock community (i.e. all organisms are present in different relative abundances within a theoretical range between 0.5% and 50%) allowed us to test the capability of each tool and pipeline to recover the exact initial microbial composition. To have absolute confidence in the reference genomes used for calculating the error rates, the

exact 16S rRNA gene sequence for the 17 species was obtained using Sanger sequencing. For all paralogous 16S rRNA genes within the mock community, no differences between paralogs could be observed for 13 species. For the 4 species with sequence variations between the paralogs (all in species present in lower than 1% in the uneven mock community), only one difference (for eight paralogs) or three differences (for two paralogs) could be observed. However, this variation could not contribute more than 0.1% to the total error rate, and will certainly not lead to an inflation of OTUs in the downstream analysis as the percentage difference is much lower than the 3% difference cut-off used. As we have to take into account the technical (e.g. pipetting errors) and PCR bias that might result in an aberration from the presumed concentrations, the raw sequencing reads were mapped back using NAST (DeSantis, Hugenholtz, Keller, et al. 2006) to the 17 reference genomes, leading to a compositional range between 0.30% and 55.8% (see Additional Information 2:1). The sequencing data are submitted to the NCBI Sequence Read Archive (PRJNA257992). Applying basic trimming (described in detail in the "preprocessing sequencing data" section) resulted in a mock community dataset consisting of 145,245 reads with an average length of 480 and a raw error rate of 0.0031.

2.3.2. Error calculation and chimera detection

To obtain erroneous bases (i.e. "the ground truth") in the TrainingDB dataset, each read was BLASTed (Altschul et al. 1990) against a database containing all the reference sequences of the corresponding mock community. After finding the potential reference sequence, an accurate alignment was produced using ClustalW (Thompson et al. 1994) adjusting the parameters as recommended in Gilles et al. (2011), to get the highest accuracy for the identification of insertions, deletions and mismatches. This computationally costly approach was needed to get the highest sensitivity in identifying

those errors that are used as training data to build the classifier, and to obtain the positional information of the sequencing errors needed to train the classifier.

In the comparative study between different tools, the global error rate was used to evaluate the accuracy of each approach, reflecting the number of erroneous nucleotides over the total number of nucleotides. For this benchmarking step the mothur command *seq.error* as implemented in Schloss et al. (2011) was used to calculate the over-all error rate. Importantly, this algorithm also implements a highly efficient approach to detect chimeric sequences in mock communities. In a first step, both the reference sequences and sequencing reads were aligned using the NAST algorithm to the SILVA reference alignment. In the second step, chimera were detected via calculating the number of mismatches between each 454 Pyrosequencing read and all possible two-parent chimera that could be generated by the reference sequences. In those cases where a sequence is at least three bases more similar to a multi-reference chimera than to a single reference sequence it is considered a chimera, and thus excluded from error calculations. The percentage of chimera was found to be 8.6% and 2.0% for MOCK1 and MOCK2 reads. Additionally to the chimera detection step, the error rate is calculated via counting the distance between the 454 reads and closest reference sequence. It was applied on both MOCK1 and MOCK2 for the non-denoised data and after each different denoising tool.

2.3.3. Denoising algorithms

Our newly introduced denoising algorithm was benchmarked with four other commonly used denoising algorithms: single linkage preclustering (SLP) (Huse et al. 2010), AmpliconNoise (Quince et al. 2011), Acacia (Bragg et al. 2012) and Denoiser (Reeder & Knight 2010). For SLP we used the implementation *pre.cluster* as available in mothur (Schloss et al. 2009), using version 1.33.3. We will refer to this algorithm as

Pre-cluster in the text below. Similarly, AmpliconNoise (PyroNoise and SeqNoise) was run using the mothur commands shhh.flow and shhh.seqs. For Denoiser we used Mac-QIIME implementation via the denoise_wrapper.py (version 1.5.0) script. For Acacia we used the original implementation (version 1.52) as available online. All algorithms were run using their default parameters.

2.3.4. Preprocessing sequencing data

As shown in the introduction, trimming of Pyrosequencing reads is a common preprocessing step in 16S rRNA amplicon sequencing (Kunin et al. 2010; Huse et al. 2007; Schloss et al. 2011). In order to allow a fair comparison between different denoising algorithms, the same input data should be used for all tools, thereby preventing the confounding effect of using different pre-processing pipelines. Therefore, the same basic preprocessing approach was applied as proposed in Schloss et al. (2011), i.e. culling reads with one or more ambiguities, removing too short reads (< 200 bp), and filtering out reads with homo-polymers longer than 8 bp. This approach – what we will refer to as "basic trimming" – is applied on all datasets discussed in this work.

Following the basic trimming step, optionally a more stringent trimming approach – further referred to in the text as "strict trimming" – can be applied. For preprocessing the data used as input for Pre-cluster, Acacia, Denoiser and our newly developed approach (NoDe), a sliding window approach was used to trim reads until the position where the average quality of this window drops below a cut-off Phred score (*trim.seqs* command in mothur (Schloss et al. 2009)). As mentioned in the introduction, the aim of this work is to develop a denoising algorithm resulting in an acceptable error rate, while preserving longer read lengths. These longer read lengths can be guaranteed by using a sliding window of 100 nt and a cut-off on the average Phred score of 30. For

AmpliconNoise, a similar effect could be achieved by trimming the sequencing data according to the guidelines stated in the original paper describing the tool, (Quince et al. 2011), by retaining only those reads with minimum flow length of 360 and maximum flow length of 720 which were later processed using exactly steps to emphasis the effect of the denoising tools solely to allow a fair comparison (see below). For both approaches (basic and strict trimming) reads were aligned to a 50,000-column wide SILVA-based reference alignment (Quast et al. 2013), using a NAST-based aligner (DeSantis, Hugenholtz, Keller, et al. 2006), as available in mothur (Schloss 2009) and filtered (*align.seqs* and *filter.seqs* commands in mothur respectively), and subsequently subjected to error calculation (using *seq.error*) for comparative analysis. For assessing the impact of the error correcting algorithms on the OTU clustering, reads were clustered using the clustering algorithm as integrated in UPARSE (a 0.97 cut-off without singletons removal) using the UPARSE command with the following options: *sortbysize*, *cluster_otus*, and *usearch_global* (Edgar 2013). Next, reads were classified using the RDP classifier (Wang et al. 2007) with 80% cutoff by applying mothur *classify.seqs* command. First, we applied this clustering approach (i.e. clustering with cut-off 0.03 and classified using the RDP classifier) on the selected V1-V3 region of the correct reference sequences to assure that a correct taxonomic classification could be obtained theoretically (i.e. by working on the correct reference sequences), see Additional Information 2:6. The same algorithmic approach was used for taxonomic classification of the MOCK2 reads after applying the pre-processing as mentioned above.

2.4. Results

2.4.1. NoDe (Noise Detector) algorithm

Our algorithm consists of two steps. First, a pre-trained classifier is used to identify those positions in the reads that are conceivable to be erroneous nucleotides based on a list of features potentially acting as a predictor for sequencing errors. In a second step the SLP algorithm (Huse et al. 2010) as implemented in mothur (Schloss et al. 2009) is adapted in such a way that those nucleotides being marked as potentially erroneous are not penalized in the mismatch counting used to cluster similar reads. Both steps are explained more in detail below.

For training the NoDe classifier, an exhaustive list of features potentially able to predict sequencing errors was derived based on conclusions presented in Schloss et al . (2011), Gilles et al. (2011) and Huse et al. (2010): i) the position in the read, ii) PhreD score, iii) the presence and exact location within a homopolymer, iv), the possibility whether this position is sensitive to carry forward events, and v) the flowgram signal intensity. Additionally we also examined the predictive effect of characteristics derived from neighbouring nucleotides (one position before and after the investigated nucleotide): i) the Phred Score, ii) the presence and exact location within a homopolymer, iii) flowgram signal intensity and iv) the highest flowgram signal intensity score that did not result in a base call, measured over all signal intensity levels in the flowgram between the position studied and the neighbouring nucleotide. Combining those features lead to a list of 13 attributes, five related to the position itself and eight linked to the neighbouring positions. Those 13 attributes were extracted from the Standard Flowgram Format (SFF) files. To properly describe the homopolymer status (presence and exact location within a homopolymer), following annotation was used: "N" for non-homopolymer, "A" and "Z" for the first and last

position in the homopolymer respectively, and for the second, third, fourth, etc. position (if any) we use "B", "C", "D", etc. Since the exact microbial composition is known for the TrainingDB, we were able to know for each position whether it was an error (insertion, deletion or substitution) or not based upon the original genome sequence of the organism at hand. Principal Component Analysis showed that all proposed features were needed to explain 95% of the variation in the training data (see Additional Information 2:2).

For training the model, several classifiers were considered (multilayer perceptron, support vector machine (SVM), decision tree, naive Bayes, nearest neighborhood, logistic regression) as implemented in WEKA (Hall et al. 2009). In order to select the best performing classifier the training dataset was split into two subsets, subset A for training, and subset B for initial testing. For constructing these subsets, the training dataset was dereplicated and distributed over subsets A and B with a ratio 1:9 in a stratified way. Afterwards, normalization of the ratio between different error types (Insertions, Deletions, Substitutions) was applied to equally train and assess each type of error (applied on subset A and B). Within subset A the native ratio between erroneous versus non-erroneous instances was 1:40, which would bias the classifier towards the more abundant one (i.e.. non-erroneous instances). Therefore subset A was reduced to have a count of 132 for each error type as well as 700 non-erroneous instances via random selection.

As evaluation criteria for assessing the best performing classifier (tested on subset B), we used the sensitivity (i.e. the proportion of actual erroneous bases that was detected as such: true positives/(true positives + false negatives)), and specificity (i.e. the proportion of actual non-erroneous bases that was detected as such: true negatives/(true negatives + false positives)). The best performing classifier was found to be an SVM with a Pearson VII Universal Kernel (PUK) (Üstün et al. 2006) and sequential minimal

optimization algorithm (Platt 1998), which achieved a sensitivity of 0.62 and specificity of 0.95. The relative influence of each feature was assessed using the feature weights of each of them. However, as the PUK kernel is a non-linear kernel, it does not directly allow to calculate feature weights. Therefore, the feature weights were illustrated by training a linear SVM for classification, showing again that all features were essential for the optimal performance of the classifier (see **Table 2:2** in Additional Information 2:2). The selected classifier (SVM with PUK kernel) optimally integrating these 13 features was used as error predicting tool in the NoDe algorithm. For any position predicted by this classifier in NoDe to be erroneous, the nucleotide will be marked as such.

In the second step, a modified version of the SLP algorithm (Huse et al. 2010) as implemented via *pre.cluster* in mothur (Schloss et al. 2009) was developed. The *pre.cluster* implementation merges the less redundant reads with no more than 2% mismatches with the more redundant reads (parameter setting recommended in Schloss et al. (2011)). In NoDe, the pre-cluster like algorithm is proceeded by a machine learning approach identifying potentially erroneous nucleotides, and those positions are masked. Accordingly, we adapted the pre-cluster algorithm implemented in NoDe in such a way that is able to ignore those masked positions in the difference calculations of pre-cluster, which means that those positions will not lead to an increase in differences. After the clustering, the remaining masked positions (i.e. positions in reads that are not merged with a more abundant one) are converted back to their original nucleotide upon rechecking the original version of the read (pre-NoDe version). A schematic representation of this approach is given in Additional Information 2:3. The source code and binaries for NoDe are freely available at http://science.sckcen.be/en/Institutes/EHS/MCB/MIC/Bioinformatics/NoDe.

It is important to stress that albeit the fact that NoDe was trained using a certain mock community amplified targeting specific region of the gene, its training is neither dependent on the mock community nor the region thus it can be used for any 454 Pyrosequencing errors.

2.4.2. Benchmarking of NoDe

The MOCK1 dataset was used to benchmark NoDe to other state-of-the-art denoising tools: Pre-cluster (Schloss et al. 2009), Acacia (Bragg et al. 2012), Denoiser (Reeder & Knight 2010) and AmpliconNoise (starting with PyroNoise followed by SeqNoise (Quince et al. 2011)). Two evaluation factors were used at this stage: the quality of those sequences in terms of error rates retained after error correction and the computational cost.

To assess the capability of each algorithm we tested them on datasets after basic and strict trimming respectively. The error rate of the MOCK1 dataset after basic trimming and without performing any error correction step was found to be 0.0050, with an average read length of 509 (888,635 total number of reads). NoDe resulted in the lowest error rate (0.0012) while AmpliconNoise, SLP, Acacia and Denoiser had an error rate of 0.0019, 0.0028, 0.0040 and 0.0045 respectively (

Table 2:1). Although different denoising algorithms processed an equal amount of data (comparable average lengths and number of sequence reads), the required computational time varied dramatically. NoDe was found to yield an almost 40-fold speed improvement over AmpliconNoise (9.5 hours for NoDe versus 370 hours for AmpliconNoise on a single Intel Xeon E5-2640 2.50 GHz CPU). On the other hand, NoDe requires more computational time compared with Acacia and Pre-cluster. However, this relatively small increase in running time over the full preprocessing pipeline is largely compensated with a significant decrease of the error rate. One

sample was used for additional analysis, delivering a detailed overview of the computational costs of the different preprocessing steps (trimming, aligning, filtering, denoising, etc.). An overview is given in Additional Information 2:4. These computational costs can be further reduced via utilizing multiple cores simultaneously, which is an option available for NoDe, AmpliconNoise, Denoiser and Pre-cluster. For Acacia this parallelization can be done manually by the end user via staring up different runs in parallel. Inherent to the mode-of-action of NoDe is the linear increase of the computational time with the number of unique reads. It should be noted that this does not mean that NoDe will increase linearly with the input data, as the number of unique reads will reach to an asymptotic value upon increasing the coverage of the sample.

The NoDe classifier consist of three processes: 1) extracting and preparing the data, 2) running the classifier, and 3) the masking phase. The first and last process (using Perl-scripting) required a maximum of 400 MBs RAM memory using MOCK2 samples, and this during 80% of the execution time. During the residual 20%, the classifier component using the WEKA software as implemented in JAVA, required a maximum of 1,800 MBs of RAM memory. For the other denoising tools, the maximum RAM requirement was found to be 1,600 MBs, 2300 MBs, 100 MBs and 1,600 MBs for AmpliconNoise, Denoiser, Pre-cluster and Acacia, respectively.

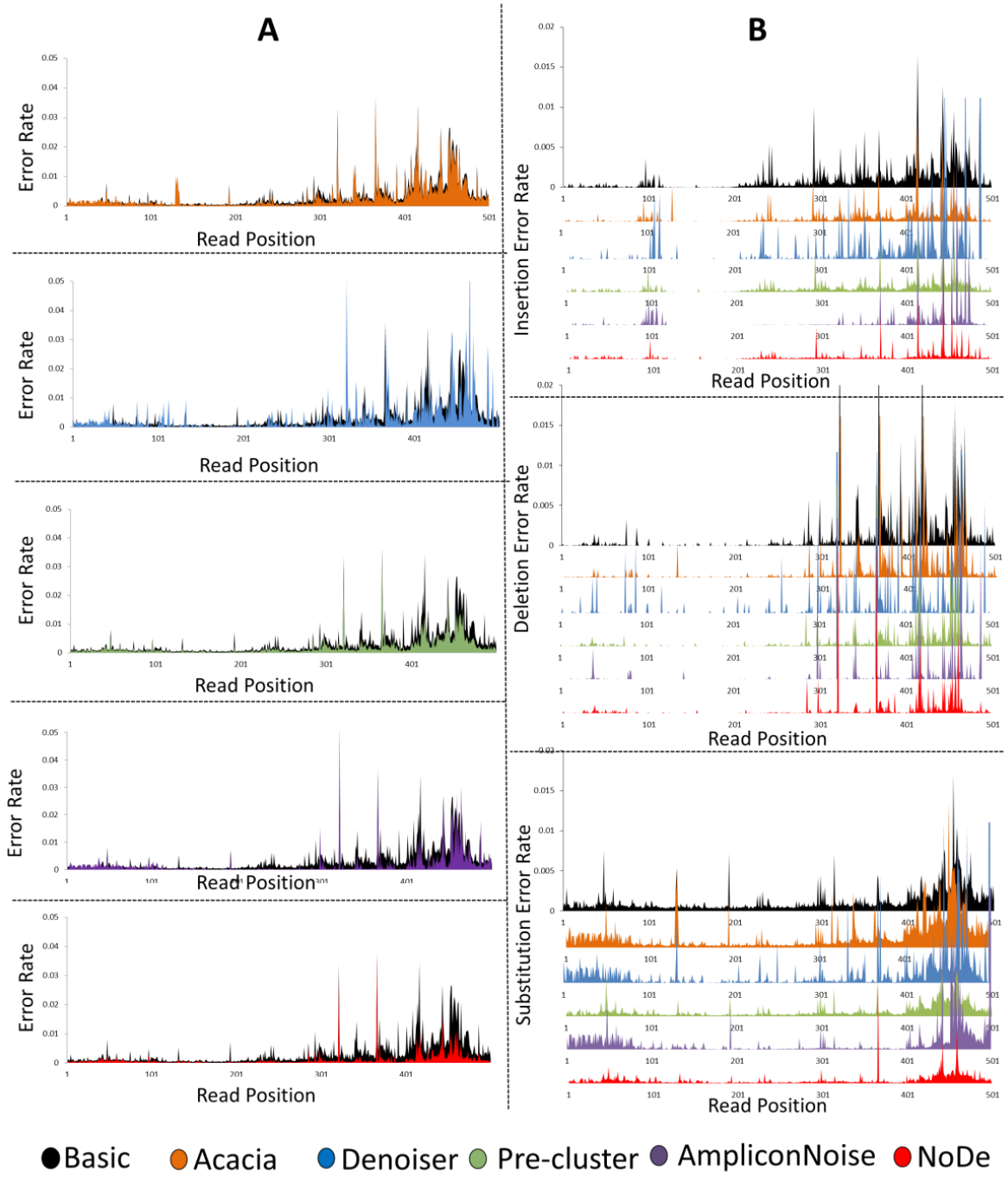


Figure 2:1. Effect of denoising algorithms with respect to position in read A) Plot showing the error rate versus the position in the read after being treated with different denoising algorithms, including: Acacia (orange), Denoiser (blue), SLP (Green), AmpliconNoise (violet) and NoDe (red), with the raw error rate in black. B) Plots showing the insertion (upper), deletion (middle) and substitution (lower) error rates produced in the raw reads (black), as well as after being treated by different approaches, versus the position in the read.

Table 2:1. Benchmarking of different denoising algorithms using the MOCK1 dataset. The comparison covers the final error rate as well as the computational cost (on a single CPU - Intel Xeon E5-2640 2.50 GHz) for the analysis pipelines including all tested denoising algorithms (Acacia, Denoiser, Pre-cluster, AmpliconNoise, NoDe). Also the number of reads and average read length returned by the different algorithms is displayed.

Basic Trimming	Average length	Error	CPU Cost	Number of seq
Denoiser	504	0.0045	112 hr	862,279
Acacia	482	0.004	8.8hr	845,513
Pre-cluster	482	0.0028	8 hr	845,513
AmpliconNoise	499	0.0019	370 hr	860,273
NoDe	481	0.0012	9.5 hr	845,513
Strict Trimming	**Average length**	**Error**	**CPU Cost**	**Number of seq**
Denoiser	439	0.0024	96 hr	785,115
Acacia	424	0.0021	7.7hr	827,123
Pre-cluster	424	0.0014	7 hr	827,123
AmpliconNoise	424	0.0013	312 hr	818,421
NoDe	425	0.0008	8.3 hr	827,123

Similarly, after applying the strict trimming step, the error rate without applying any denoising algorithm was found to be 0.0026 with an average read length of 441. Applying NoDe on these sequencing data resulted in an error rate of 0.0008, which is significantly lower than the error rates obtained with Denoiser (0.0024), Acacia (0.0021), Pre-cluster (0.0014) and AmpliconNoise (0.0013), see **Table 2:1.** A graphical representation of the effect of the different denoising tools on the sequencing data with respect to the position of the error in the read is given in **FIGURE 2:1**. As expected, these plots show that the total error rate (i.e. sum of the fraction of insertions, deletions and substitutions) is mainly increasing towards the end of the read. In the second plot, the performance of each denoising tool on different types of errors (average numbers are given in Additional Information 2:5) is illustrated. It is important to notice that peaks seen at the beginning of the read and those consistently high (albeit the various denoising tools used) could be explained by PCR-errors, missed chimera (of closely

related parents), or missing (non-identical) paralogs from the reference sequences. However, as those factors are the same for all tools, they is not expected to affect the fairness of the comparative analysis. Also for the strictly trimmed sequencing data, NoDe outperformed the second best benchmarked tool (AmpliconNoise) in computing time, as NoDe processed the same dataset in 8.3 hours versus 312 hours for AmpliconNoise. An overview of the computational cost of each step in the preprocessing pipeline – including the different denoising algorithms –as in **Table 2:2**.

Table 2:2. Tabular overview of the computational cost of the different denoising algorithms. To have an idea about the computational cost for each step, the complete pipeline was subdivided in different steps to illustrate its running time, as described above. From the table, it can be observed that the computational burden added to the complete preprocessing pipeline was relatively small, and it was largely compensated with a significant improvement in the error rate, exceeded the second best performing (computationally intensive) AmpliconNoise.

Denoising Approach	SFF Extraction	Trim Reads	Denoising Algorithm (average memory)	Aligning	Filter Alignment	Total time
NoDe	00:00:12	00:00:02	00:02:16(760 MBs)	00:06:40	00:01:00	00:10:05
Pre-cluster	00:00:12	00:00:02	00:00:13(100 MBs)	00:06:40	00:01:00	00:08:02
AmpliconNoise	00:00:12	00:00:01	08:25:17(1,900 MBs)	00:03:40	00:01:00	08:30:27
Denoiser	00:00:12	00:00:01	00:38:17(2,300 MBs)	00:02:30	00:01:00	00:42:00
Acacia	00:00:12	00:00:02	00:00:55(1,600 MBs)	00:06:40	00:01:00	00:08:49

2.4.3. Impact of error correction methods on OTU clustering

The final step in amplicon sequencing-based community profiling is the clustering of reads into Operational Taxonomic Units (OTUs), which are believed to reflect a well-delineated taxonomic group. However, different proposed amplicon sequencing processing pipelines tested on artificial communities all lead to an inflation of the number of OTUs reported, often multiple times higher than the number of bacterial species present in the tested mock community (Huber et al. 2007; Sogin et al. 2006; Kunin et al. 2010).

A second mock community (MOCK2, in-house produced) was used to assess the influence of different denoising and preprocessing methods on the OTU distribution. As the exact concentration of each species is known in this uneven mock community, the accuracy of the OTU clustering process can be followed up (e.g. a species assigned to more than one OTU – i.e. over-splitting – or absence of specific species). This MOCK2 community consists of 17 species spread over a wide taxonomic range and sequenced in triplicate as described in the methods section. The difference in error rates between different denoising algorithms for MOCK2 showed the same trend as for MOCK 1, i.e. an error rate of 0.0021 for AmpliconNoise, 0.0037 for Denoiser, 0.0025 for Acacia, 0.0024 for Pre-cluster while the lowest error rate of 0.0009 was achieved by NoDe.

We applied the OTU clustering algorithm using UPARSE (Edgar 2013) as described in the Methods section. As a validation step we first applied this algorithm on the 17 reference sequences only, resulting in 17 distinct OTUs each representing a species in the MOCK2 dataset. In the ideal case, such 17 distinct OTUs would also be returned upon analysing the 16S amplicon Pyrosequencing data. To check to which extent this could achieve using any of the denoising tools, UPARSE was applied on each of the denoised sequencing datasets obtained after applying AmpliconNoise, Pre-cluster, Acacia, Denoiser and NoDe, and the number of OTUs are reported as average number over the three replicates. NoDe had the smallest number of OTUs (22 OTUs on average, which is the closest to the optimal number of 17), while AmpliconNoise, Denoiser, Pre-cluster and Acacia had 29, 24, 46 and 46 OTUs on average respectively. When omitting the error correcting algorithms (i.e. when doing OTU clustering on the data directly after basic trimming) the number of OTUs even further inflated to 58 OTUs, pointing out the importance of integrating a denoising algorithm in a 16S rRNA amplicon sequencing pipeline.

All the OTUs obtained were evaluated qualitatively (checking the taxonomic classification of each OTU) and quantitatively (checking the abundance of the OTUs). For the qualitative analysis, we counted the number of "correct OTUs" (classified as one of the species present in the mock community), "noisy OTUs" (classified as one of species in the mock but unclassified at the Genus, Family or/and Order level), "missed OTUs" (species present in the mock community but totally absent in the OTU classification), "over-split OTUs" (correct yet redundant classification), "contaminant OTUs" (classified as species which should not be in the mock community) and "other OTUs" (OTUs unclassified at the Class level or higher).

All of the different preprocessing pipelines determined the correct relative percentage for each mock species. However, from a qualitative point of view some of the species suffered from OTU over-splitting (i.e. one species is split up over different OTUs), and is observed with all error-correcting algorithms (**Table 2:3** and Additional Information 2:6) however at different levels. The number of over-split OTUs was 4, 4, 3, 24 and 23 for NoDe, AmpliconNoise, Denoiser, Pre-cluster and Acacia, respectively. Additionally, the number of unclassified OTUs (the sum of 'noisy' or 'others' OTUs) could also be used as a quality criterion, since all organisms present in MOCK2 are well-known species present in all standard reference databases (e.g. SILVA). As such this number of unclassified OTUs should be as low as possible. For this mock community 1, 7, 5, 4 and 5 unclassified OTUs were detected for NoDe, AmpliconNoise, Denoiser, Pre-cluster, and Acacia respectively. For both aberrant types of OTUs NoDe showed the lowest number compared to other tested denoising algorithms, showing the beneficial influence of an accurate error correction tool. Additionally, over-clustering was assessed by checking the number of missed OTUs. Upon checking the closely related species (two species of Staphylococcus and three of Streptococcus) within MOCK1, we could see that on average both Denoiser and AmpliconNoise suffer from a

missing OTU due to over-clustering, while NoDe, Acacia an Pre-cluster do not have this problem. It is important to emphasise that any differences in the contaminants' count were a consequences of the variations in the pre-processing steps.

Table 2:3. OTUs produced after treating the data with different noise removal approaches. The left side of the table displays the qualitative OTU assessment and the right side displays the quantitative analysis. For the qualitative analysis, we counted the number of "correct OTUs" (classified as one of the mock species), "noisy OTUs" (classified as one of mock species but only classified until Class, Order or Family level), "missed OTUs" (number of undetected mock species), "over-split OTUs" (correct yet redundant classification), "contaminant OTUs" (classified as species no belonging to mock) and "other OTUs" (OTUs unclassified at the Class level or higher). In the quantitative analysis, the number of OTUs with an abundance below 0.1% (rare OTUs) and the ones with an abundance redundancy above 0.1% (Redundant OTUs) were counted.

	Qualitative							Quantitative		
Approaches	Total OTUs	Correct OTUs	Over-Split OTUs	Missed OTUs	Noisy	Contaminants	Others	Approaches	Rare-OTUs	Redundant OTUs
NoDe	22	17	4	0	1	0	0	NoDe	4	18
AmpliconNoise	29	16	4	1	2	1	5	AmpliconNoise	11	18
Denoiser	24	16	3	1	1	0	4	Denoiser	7	17
Pre-cluster	46	17	24	0	4	0	0	Pre-cluster	22	24
Acacia	46	17	23	0	5	1	0	Acacia	21	25
Non Denoised	58	17	29	0	5	1	7	Non Denoised	35	17

In the quantitative analysis, we count the number of OTUs with an abundance below 0.1% (rare OTUs) and the ones with an abundance above 0.1% (redundant OTUs). If we used the number of rare OTUs as an indicator for a better error correction step (the lower the better), the noise removal step was more accurate with NoDe resulting in only 4 rare OTUs, while AmpliconNoise, Denoiser, Pre-cluster and Acacia had 11, 7, 22 and 21 rare OTUs respectively. Indeed, the number of rare OTUs is – with exception of Denoiser – proportional to the error rate produced. Although, as Denoiser is returning the highest error rate yet resulting in a low number of rare OTUs, an extra analysis was performed plotting the expected percentage of a species versus the

observed percentage after OTU clustering (Additional Information 2:7). From these plots it can be derived that the percentages of the different species in the MOCK communities obtained using different denoising strategies correlate better with the actual percentages with decreasing error rate. This is largely reflected by the R-square goodness of fit value derived for each denoising algorithm, resulting in the highest value for NoDe (0.978), while observing the lowest R-square value for Denoiser (0.928) showing the highest error rate. The same conclusion could be drawn when using the method as described in Bragg et al. (2012) for assessing the correspondence between the theoretical and observed proportions of the species in the mock community. Also here, the average relative deviation of the observed concentration of a species after analysing the data versus the theoretical concentration was the lowest for NoDe, (Additional Information 2:7)

2.5. Discussion

New sequencing technologies have revolutionized the way microbial communities are characterized, but still suffer from amplification and sequencing artefacts. In this work we proposed a new denoising methodology NoDe which is able to significantly reduce the sequencing error rates at a low computational cost (CPU). In general our method consists of a two-step approach. First an artificial intelligence based classifier is trained to identify those positions in 454 sequencing reads having a high sequencing error probability. These positions are identified via a set of features that are able to predict less reliable sequencing regions. By marking those positions, a quality-driven clustering of reads is made possible via a modified version of the Pre-cluster command in mothur in the second step.

When comparing our algorithm with other denoising algorithms, we could show a significant improvement at the level of error rate reduction (i.e. 37% to 73% more

errors that could be corrected compared with the second best performing algorithm). As such, NoDe manages to bring the error rate of 454 Pyrosequencing reads to an acceptable level in MOCK1 (0.0008 with strict trimming and 0.0012 with basic trimming) while retaining a large proportion of the sequencing data. The latter is reflected in a long read length (> 400 bp) which will result in a more precise taxonomic classification (Klindworth et al. 2013). Moreover, when comparing the required computational resources of NoDe with the second best denoising algorithm (AmpliconNoise), more than an order of magnitude reduction in computational cost could be obtained. The computational burden added by NoDe in the complete preprocessing pipeline as implemented in mothur is very limited, and largely compensated by the improvement in the error rate (e.g. 0.0012 versus 0.0028 in MOCK1 after basic trimming, and 0.0009 versus 0.0024 in MOCK2 for NoDe and Pre-cluster respectively).

Additionally, we could show that denoising using NoDe has a beneficial effect on the number of OTUs returned after clustering, reaching almost a one-to-one relationship between the number of OTUs and the number of species that are present in our artificial community. Moreover this process could be completed at a reasonably low computational cost. Such a close correlation between the number of OTUs and the number of present bacteria could not be achieved on the studied mock community using any of the other error correcting algorithms. Moreover, also applying the UPARSE pipeline without integrating the error correction step leads to a larger deviation from the one to one relationship between OTUs and species present in the tested communities. However, it should be noted that obtaining such a one-to-one relationship with NoDe was obtained using mock communities, and caution should be taken when extrapolating those results to real biological samples.

In this work we focused on applying error correction on 16S rRNA amplicon sequencing data. However, in principle this method is applicable to all amplicon sequencing data obtained via the Roche 454 Pyrosequencing technology. As a proof of concept, we successfully tested our algorithm on the sequencing data presented in Gilles et al. (2011) containing control DNA fragment type I sequences as provided with 454 sequencing kits (data not shown). Similarly, NoDe implementation is trained and benchmarked using different 454 GS-FLX titanium sequencing dataset. However a highly similar approach could also work with sequencing data obtained via the recent GS-FLX+ technology, producing reads exceeding 1000 bp.

2.6. Conclusion

We have developed a new denoising algorithm NoDe that produces lower error rates compared with other existing denoising algorithms. Moreover, using the MOCK2 community we could show that error correcting algorithms are a necessary and powerful step to come to biologically relevant numbers of OTUs, which were hard to obtain without any denoising step. NoDe is able to perform this error correcting step in a computational realistic time frame, without being a bottleneck in the preprocessing pipeline.

Chapter 3

IPED: a Highly Efficient Denoising Tool for Illumina MiSeq Paired-end 16S rRNA Amplicon Sequencing Data

Mohamed Mysara[1,2,3,4], Natalie Leys[1], Jeroen Raes2,[3,4] and Pieter Monsieurs[1]

[1] Unit of Microbiology, Belgian Nuclear Research Centre (SCK-CEN), Mol, Belgium.
[2] Department of Bio-engineering Sciences, Vrije Universiteit Brussel, Brussels, Belgium.
[3] VIB Center for the Biology of Disease, VIB, Leuven, Belgium.
[4] Department of Microbiology and Immunology, REGA institute, KU Leuven, Belgium.

This chapter is based on the following publication:

Mysara M, Leys N, Raes J, Monsieurs P. (2016). IPED: a highly efficient denoising tool for Illumina MiSeq Paired-end 16S rRNA gene amplicon sequencing data. BMC Bioinformatics 17:192.

3.1. Abstract

The development of high-throughput sequencing technologies has revolutionized the field of microbial ecology via the sequencing of phylogenetic marker genes (e.g. 16S rRNA amplicon sequencing). Denoising, the removal of sequencing errors, is an important step in preprocessing amplicon sequencing data. The increasing popularity of the Illumina MiSeq platform for these applications requires the development of appropriate denoising methods. Here, a new denoising algorithm –named IPED- is proposed including a machine learning method which predicts potentially erroneous bases in sequencing reads based on a combination of quality metrics. Subsequently, this information is used to group those error-containing reads with correct reads, resulting in error-free consensus reads. This is achieved by masking potentially erroneous bases during this clustering step. Compared to the second best algorithm available, IPED detects double the amount of errors. Reducing the error rate had a positive effect on the clustering of reads in operational taxonomic units, with an almost perfect correspondence between the number of clusters and the theoretical number of species present in the mock communities. Our algorithm IPED is a powerful denoising tool for correcting sequencing errors in Illumina MiSeq 16S rRNA amplicon sequencing data. Apart from significantly reducing the error rate of the sequencing reads, it has also a beneficial effect on their clustering into operational taxonomic units.

3.2. Introduction

The development of high-throughput sequencing technologies has revolutionized the field of microbial ecology by offering a cost-efficient method taking microbial diversity at a new level of accuracy. Initial ecological applications mainly relied on the usage of the 454 Pyrosequencing platforms, resulting in an impressive repository of bioinformatics analysis tools for processing this kind of data, as used for example in 16S rRNA amplicon sequencing data. Linking different tools developed for the preprocessing of amplicon sequencing data has resulted in frequently used analysis pipelines such as mothur (Schloss et al. 2009), QIIME (Caporaso et al. 2010) and UPARSE (Edgar 2013).

Due to the recent advances in other high-throughput sequencing technologies regarding throughput and read length, and the announcement of Roche to shut down its 454 services by 2016, sequencing platforms provided for example by Pacific Biosciences and Illumina gain importance for assessing microbial diversity using amplicon sequencing. However, analysis pipelines developed for 454 Pyrosequencing data cannot be translated into an Illumina MiSeq specific pipeline in a straightforward way due to fundamental differences between both sequencing technologies.

Indeed, the 454 Pyrosequencing technology has difficulties in predicting the exact length of homopolymers, as such mainly leading to indel errors (Loman et al. 2012; Kozich et al. 2013). Illumina sequencing data do not suffer from indel errors to the same extent, but rather from nucleotide substitutions (miscalling), mainly originating from two effects: 1) high correlation of the intensities of A and C as well as G and T due to similar emission spectra of the fluorophores (Kircher et al. 2009; Bentley et al. 2008; Rougemont et al. 2008), and 2) dependency of the signal of each cycle on the signal before and after this cycle, caused by inadequate flushing of fluorophores, incomplete removal of the 3' terminators, or integration of nucleotides without

effective 3' terminators (Kircher et al. 2009), known as phasing and pre-phasing. Additionally, it has been shown that such substitutions are often linked to the presence of the GGC motif, or more in general, to the GC-richness of the amplified region (Allhoff et al. 2013; Tsai et al. 2014; Schirmer et al. 2015).

Different approaches have been developed for reducing sequencing errors originating from the Illumina MiSeq sequencing platform. These approaches can be categorized into three types: 1) denoising tools which dedicated to actively resolve sequencing errors, 2) paired-end assemblers that merge overlapping reads into one contig represented by a consensus sequence (specifically for Illumina MiSeq amplicon paired-end sequencing data), and 3) quality filtering approaches which remove poor-quality reads or regions.

A large number of denoising algorithms has already been developed for 454 Pyrosequencing reads, for example Denoiser (Reeder & Knight 2010), AmpliconNoise (Quince et al. 2011), SLP (Huse et al. 2010), Acacia (Bragg et al. 2012) and NoDe (Mysara, Leys, et al. 2015). However, due to fundamental differences in the nature of both sequencing technologies, denoising algorithms developed for 454 Pyrosequencing data are likely to perform suboptimal when applied to Illumina sequencing data in a naive way. At this moment, a few algorithms for denoising Illumina MiSeq paired-end amplicon sequencing data have been developed. One of the 454 Pyrosequencing denoising algorithms, i.e. SLP (Huse et al. 2010) -implemented as *pre.cluster* in mothur- and the more recently released algorithm called UNOISE (Edgar & Flyvbjerg 2015), have been shown to be applicable to Illumina MiSeq specific analysis pipelines (Kozich et al. 2013).

However, next to denoising tools, a plethora of tools has been developed for assembling paired-end reads into one amplicon contig, by merging both the forward and reverse reads into one consensus sequence. Apart from the assembly tools

integrated in the amplicon sequencing pipelines like mothur, QIIME and USEARCH, other more general paired-end assembly algorithms have been developed such as FLASH (Magoč & Salzberg 2011) PANDAseq (Masella et al. 2012), COPE (Liu et al. 2012) and PEAR (Zhang et al. 2014). Additionally, several quality filtering approaches were implemented to identify and remove or trim reads with poor quality, according to specific criteria defined within each tool, as implemented in mothur, QIIME and USEARCH. Despite the fact that paired-end assemblers and quality filtering approaches cannot be seen as genuine denoising tools, they will have an effect on the error rate, and should be included in a benchmark when assessing denoising tools.

In this work we propose the Illumina MiSeq Paired-End Denoiser (IPED) algorithm, an error correction algorithm specifically developed for denoising Illumina MiSeq 16S rRNA amplicon sequencing data. Our machine learning methodology was benchmarked using four different mock datasets, each of them containing sequencing data of different hypervariable regions in the 16S rRNA gene, including completely as well as partially overlapping paired-end reads. Working with mock communities consisting of a known set of species had the major benefit that we can use the error rates as most prominent evaluation criterion. Since the amount of data produced within one sequencing run is steadily increasing for the Illumina MiSeq technology, we evaluated the additional computational cost associated with our algorithm.

3.3. Materials and methods

3.3.1. Mock Communities

Four publicly available Illumina MiSeq sequencing datasets of mock communities were used within this work. The first mock community – called MOCK1 – is composed of 21 species added in equimolar concentrations (5 ng/µl), as illustrated in Kozich et al. (2013). The second mock community, termed MOCK2, has almost the same

composition as MOCK1, however omitting one species, resulting in 20 different organisms (Nelson et al. 2014). The DNA of the mock communities can be obtained from BEI Resources (catalog number HM-278D). Both Illumina MiSeq libraries were prepared using primers as described in the work of Kozich et al. (2013) for the amplification of the V3-V4, V4 and V4-V5 hyper-variable regions of the 16S rRNA gene for MOCK1, and in the work of Nelson et al. (2014) for the amplification of the V4 and V4-V5 region for MOCK2. Both mock communities were sequenced on the Illumina MiSeq platform using the 2×250 bp paired-end protocol. Merging both reads into one contig resulted in different contig lengths for each primer pair: contigs resulting from the V4 primer pairs in MOCK1 and MOCK2 resulted in a length of 251-253 bp (completely overlapping paired-end reads), V4-V5 contigs in MOCK1 and MOCK2 in lengths of 375 nt and 390 nt respectively (overlapping regions of 125 and 110 bp respectively) and V3-V4 contigs in MOCK1 in a length of 430 nt (overlapping region of 70 bp). In MOCK1, four sequencing runs were performed with various cluster densities (ID's: 130401,130403, 130417 and 130422). In MOCK2, samples for each region (V4 and V4-V5) were run in duplicate (named v4.I.1, v4.I.05, v4.v5.I.1 and v4.v5.I.11 respectively).

The third mock community – called MOCK3 – consists of three samples (named M1, M2, M3), each consisting of 12 species (Sequencing Read Archive accession number SRP066114). The MOCK3 community was sequenced on the Illumina MiSeq platform using the 2×300 bp paired-end protocol. Merging both reads into one contig resulted in a length of 422-428 bp after clipping the primers (overlapping regions of 141 bp). **Table 3:1** provides a detailed description of the MOCK1, MOCK2 and MOCK3 sequencing datasets.

The fourth mock community - called MOCK4 - consists of 73 samples and was recently published in Schirmer et al. (2015). Their microbial composition ranges from a single

species to diverse mock communities (49 bacteria and 10 Archaea), mixed either in even or uneven concentrations. Five different Illumina MiSeq library preparation methods were used to amplify the V4 and the V3-V4 region. Contigs constructed by merging both reads resulted in two different lengths, ranging from 253 nucleotides (i.e. almost completely overlapping reads) to 450 nucleotides (partially overlapping reads). A detailed description of MOCK4 and the sequencing protocols used can be found in the original publication (Schirmer et al. 2015).

Table 3:1. Overview of the mock sequencing data discussed in this work. It contains information on the amplified regions, samples ID's, number of paired-end reads (i.e. contigs), average contig length (i.e. length after merging both paired-end reads), and average length of the overlapping part between both paired-end reads.

Name	Region	length	Overlap	ID	#contigs
MOCK1 (Kozich et al. 2013)	V3-V4	430	70	130401	184216
				130403	131241
				130417	102547
	V4	250	250	130422	79701
				130401	1217529
				130403	1191998
	V4-V5	375	125	130417	1015673
				130422	871118
				130401	826262
MOCK2 (Nelson et al. 2014)	V4	250	250	v4.I.1	213043
				v4.I.05	240682
	V4-V5	390	110	v4.v5.I.1	2484
				v4.v5.I.11	90126
MOCK3 (SRP066114)	V3-V4	421	140	M1	35168
				M2	60488
				M3	21723

3.3.2. Pre-Processing Steps

For all datasets (MOCK1, MOCK2, MOCK3 and MOCK4) contigs were created by merging the paired-end reads using a heuristic based on the difference in Phred quality scores of both reads as implemented via the *make.contigs* command in mothur (Kozich et al. 2013). Contigs were culled if they had an ambiguous base or if they were not properly merged. All sequencing data were trimmed, aligned, screened, filtered and

dereplicated using the mothur software package (v.1.33.3), thereby following the SOP as described on the mothur website (http://www.mothur.org/wiki/MiSeq_SOP d.d. 2015-05-18). Afterwards, reads were denoised (by any of the tested denoising algorithms) and chimeras were identified using *seq.error* command in mothur (Schloss et al. 2009) using the full length 16S rRNA genes of the mock community species as reference. This *seq.error* command was also used to identify sequencing errors. To assess the performance of our newly introduced algorithm, IPED was benchmarked against the Pre-cluster and UNOISE algorithms, both using the recommended parameter settings as proposed by the initial developers (described in the mothur MiSeq SOP and the UNOISE publication respectively (Edgar & Flyvbjerg 2015)).

Despite the fact that paired-end assembly algorithms cannot be considered as denoising algorithms in the strict sense, their potential influence on the error rate required a comparison with the error rates obtained using IPED. Therefore, different paired-end assembly algorithms used for amplicon sequencing were tested on the MOCK1, MOCK2 and MOCK3 datasets. Algorithms included are the standalone tool PEAR as well as the assembly steps as included in mothur, QIIME and USEARCH, together with their proposed quality filtering. An overview of the commands used for those tools is given in Additional Information 3:4.

For MOCK1, MOCK2 and MOCK3 the sequencing reads were clustered in operational taxonomic units (OTU) using the mothur recommended clustering approach (shown in Additional Information 3:8) as well as UPARSE (Edgar 2013) with default parameters, with the exception of singleton removal. The default setting of singleton removal was deactivated to accurately assess the effect of sequencing errors on all the OTUs produced, including singletons. The exact commands are given in Additional Information 3:3.

3.3.3. Training Data

An important component of the IPED algorithm is the machine learning method developed to predict potentially erroneous bases. A dedicated dataset for training and testing this machine learning method was created by randomly selecting reads from sample 130401 of MOCK1 (1,000 reads from the V3-V4, V-V4 and V4-V5 region respectively), resulting in a dataset of 3,000 reads. Important to notice is that all three samples used to construct the training data were completely disregarded in the subsequent benchmarking analysis. Each nucleotide in those reads was evaluated as being either erroneous (mismatch, insertion or deletion) or correct, identified as such by aligning those reads against the reference genomes. In order to obtain highly reliable training instances, the same parameter setting was applied as used in Gilles et al (2011) for running BLAST (Altschul et al. 1990) and subsequently ClustalW (Thompson et al. 1994) in order to obtain high-quality alignments. This led to a dataset consisting of 1,031,625 instances (i.e. all nucleotide positions in the 3,000 reads). These data were cleaned as follows: dereplication, randomization and simplification via selecting a subset of the features (see further for details on the feature selection step). Next, the data were split into three subsets, thereby respecting the initial ratio between erroneous versus non-erroneous instances throughout the three subsets: (a) a learning dataset for training the classifier, (b) a validation set for selecting the most optimal kernel and (c) a test dataset for testing the accuracy of the classifier. Subset (a) and (b) were further modified by adjusting the ratio between erroneous and non-erroneous instances: for subset (a), several ratios between erroneous and non-erroneous instances were applied to select the one resulting in the best performance when training the classifier, while for subset (b) we used an equal ratio between both classes. Extra information on the feature-selection step and selecting the correct ratio in subset (a) is

given in Additional Information 3:1. All machine learning methods for training and testing IPED were used as implemented in the WEKA software version 3.7.11 (Hall et al. 2009).

3.3.4. Evaluation Parameters

For the evaluation of IPED, we calculated the number of true positives (TP), false negatives (FN), true negatives (TN) and false positives (FP) as follows: if an erroneous nucleotide was correctly detected as such, it is a TP, if it was not it is a FN, if an non-erroneous nucleotide was correctly detected as such it is a TN, if not it is a FP. We used the Mathews Correlation Coefficient (MCC) (Matthews 1975): $(TP \times TN - FP \times FN)/\sqrt{((TP+FP)(TP+FN)(TN+FP)(TN+FN))}$, sensitivity (i.e. the proportion of actual erroneous bases that was detected as such: $TP/(TP + FN)$), specificity (i.e. the proportion of actual non-erroneous bases that was detected as such: $TN/(TN + FP)$), and Receiver operating characteristics (ROC). The latter analysis combines both sensitivity and specificity by plotting the sensitivity (Y axis) against one minus the specificity (X axis)). ROC curves were produced by swiping the threshold cut-off of the probability estimated by each classifier, and plotting the sensitivity versus one minus the specificity value.

3.4. Results

Our newly developed algorithm for denoising Illumina MiSeq amplicon sequencing data was developed in two steps. First an artificial intelligence classifier was trained to detect potentially erroneous bases in the sequencing reads. Secondly, a modified version of the previously published Pre-cluster algorithm (Schloss et al. 2011), which is the mothur implementation of the Single Linkage Preclustering algorithm (Huse et al. 2010), was adapted in such a way that it does not penalize those potentially erroneous bases during clustering. The development of both the classifier and

clustering component of the IPED algorithm is discussed in the first part of this results section. Once the setup and training of IPED has been finalized, the algorithm was tested on a wide range of datasets against Pre-cluster and UNOISE, and this at the level of error rate, computational cost and the accuracy of the OTU clustering.

3.4.1. IPED Development

IPED Classifier

The performance of the IPED algorithm is largely depending on the ability of its classifier component to correctly identify erroneous bases. The machine learning approach followed to develop this classifier is based on supervised learning, where we trained the classifier to identify such positions by training it on a dedicated dataset (i.e. a dataset containing correct and erroneous bases). Training of the IPED classifier consisted of four consecutive stages: 1) identify those parameters that are potential predictors of sequencing errors (i.e. feature identification), 2) select the most informative parameters (i.e. feature selection), 3) train the classifier to identify sequencing errors based on those parameters (i.e. model training), and 4) check whether the classifier correctly predicts sequencing errors on unseen sequencing data (i.e. validation).

The first stage consisted of extracting a list of features potentially predicting erroneous bases. Different sequencing characteristics have been taken into consideration such as the position in the read, the homopolymer status and Phred quality score, the presence of the GGC motif in front of the position in question, the homopolymer status ("0" in case of no homopolymer, "-1" when the nucleotide is ambiguous, and in case of a homopolymer an ascending number indicating the position within the homopolymer) and the Phred quality score of the preceding and succeeding position for both the forward and reverse reads, totalling up to 16 features. An additional feature was added

indicating whether the nucleotide was situated in the overlapping region of both paired-end reads, and if so to indicate whether they have no conflict, a conflicting base call (mismatch, deletion or insertion), or unknown overlap (if at least one of both nucleotides is ambiguous).

Importantly, integrating too many uninformative features would have led to an inflation of the computational cost. Accordingly, reducing the total number of available features (17 in total) while retaining the predictive power of our algorithm has a beneficial effect on the performance of the classifier as it increases its accuracy and reduces the computational cost. Therefore, in a second stage, a feature-selection step was applied using a three-fold cross-validation on the training data. This approach allowed us to identify those features having a high predictive power for recognizing erroneous bases (i.e. having a high correlation with the class 'error' or 'non-error'), while having a low correlation with other predictive features. Performing this step resulted in a subset of six features i.e. for the forward read: the position in the read, homopolymer status and Phred quality score; for the reverse read: the position in the read and Phred quality score; and as last feature the overlap status between the forward and reverse read. Important to notice within this context is the absence of the GGC motif after the feature selection step. However, this might not be surprising as we could clearly see a drop in the Phred quality score in the positions succeeding this motif (data not shown). As the Phred quality score was retained after applying the feature-selection step, the GGC motif was removed from the features list. This could be explained either by the weak evidence of error incidences related to the presence of the GGC motif (Nakamura et al. 2011) or due to the high correlation between this motif and the Phred quality score, making this feature superfluous.

In the third stage, these six features were used to train a wide range of classifiers available in WEKA based on subset (a) of the training data. The goal of this step was

to select the type of classifier that obtains the highest accuracy in predicting erroneous bases. In order to optimize the balance between specificity and sensitivity, a range of ratios balancing the number of erroneous versus non-erroneous instances was tested. The highest sensitivity while maintaining an acceptable specificity was obtained using a ratio of 1:3 (error : non-error), an overview of different ratios is available within Additional Information 3:1. The training process was further evaluated by plotting the learning curves for each of the classifiers, which confirmed that a training dataset size consisting of 5,000 erroneous and 15,000 non-erroneous instances (respecting the 1:3 ratio) gave the best performance (see Additional Information 3:2). Subset (b) of the training data was used to evaluate each of the individual classifiers using sensitivity, specificity, MCC measurements and ROC analysis as performance parameters. Additional analyses tested the performance obtained when different sets of classifiers (ranging from two up to five classifiers) were combined using plurality voting. In a plurality voting approach, each of the considered classifiers outputs a confidence score for the classification made and the class with the highest confidence is selected as output for the respective instance. Comparing the performance of the individual classifiers as well as the different voting combinations (see Additional Information 3:2), the best performance was achieved via plurality voting combining Multilayer Perceptron (MLP – with a learning rate of 0.3, momentum of 0.2, and using 6 perceptrons in one hidden layer) and Random Forest (using an ensemble of unpruned decision trees). This combination achieved a sensitivity of 0.57, specificity of 0.93, MCC of 0.53 and ROC area under the curve (AUC) of 0.87 on subset (b) of the training data, thereby outcompeting the other machine learning approaches. The additional computational burden resulting from this plurality voting approach was minimal, as the best performing single classifier (random forest) required 2.2 sec compared to 2.3 sec

for the plurality voting approach (including MLP and random forest), tested on subset (c) of the training data, containing around 200,000 instances.

IPED Clustering

As mentioned above, the first step in the development of IPED consisted of training a classifier able to predict potentially position-dependent erroneous bases with high accuracy. In the second step a modified version of the SLP algorithm (Huse et al. 2010) as implemented via the Pre-cluster command in mothur (Schloss et al. 2011) has been developed (Mysara, Leys, et al. 2015). In the original Pre-cluster implementation, sequences are sorted based on their abundance level in descending order. When a rare sequence is at maximum 1 nt per 100 nt differing from a more abundant one, it is merged with the more abundant one and its abundance is added to the latter one. In the first step of IPED, the classifier has marked some of the positions as potentially erroneous. We have developed a modified version of the mothur Pre-cluster algorithm that will not penalize those marked positions when calculating the amount of conflicting positions between two reads. This means that any position in the alignment containing a nucleotide which is marked as potentially erroneous, will not increase the distance score (i.e. the score used as cut-off to either merge two reads, or leave them ungrouped). After the clustering step, those masked positions are reverted to their original nucleotide as they were before running IPED. A schematic representation of this approach is given in **Figure 3:1**. The IPED software can be downloaded via http://science.sckcen.be/en/Institutes/EHS/MCB/MIC/Bioinformatics/ or https://github.com/M-Mysara/IPED.

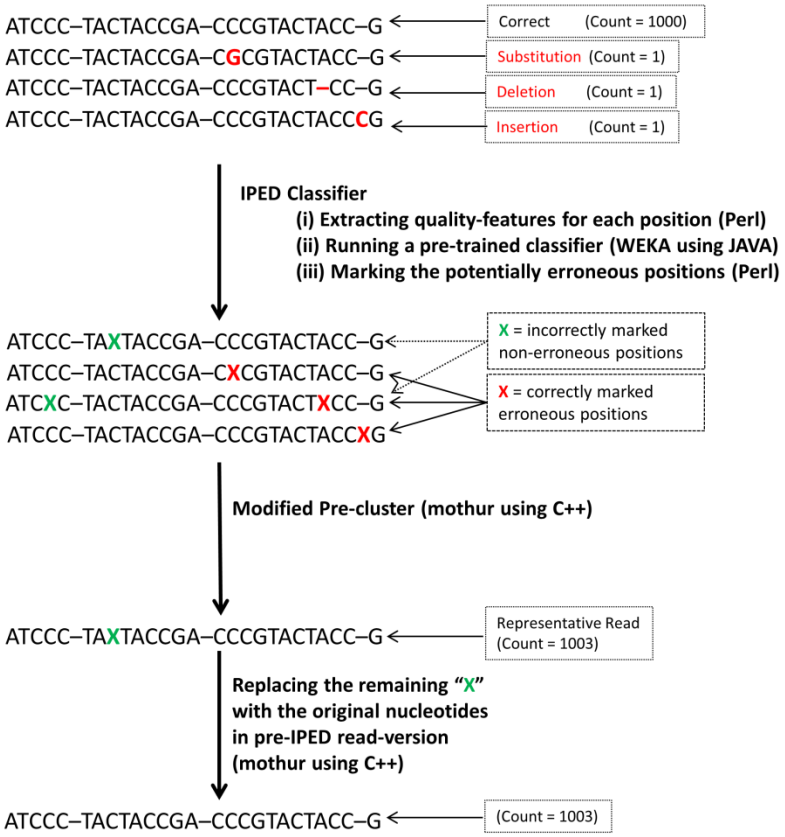


Figure 3:1. Schematic overview showing the different steps of the IPED algorithm.

3.4.2. Benchmarking of IPED

Impact of denoising algorithms on the error rate

Once the development of the IPED algorithm has been finalized, its performance was compared with those of the Pre-cluster and UNOISE algorithms. Both those tools are the only denoising algorithms currently applicable for Illumina amplicon sequencing data. Using the reference 16S rRNA gene sequences from the organisms present in the MOCK1, MOCK2 and MOCK3 communities, we calculated the error rates before applying the denoising algorithms, which were subsequently compared with the error rates obtained after applying IPED, Pre-cluster and UNOISE on all three mock datasets. The error rate was calculated using the *seq.error* command by taking the ratio of the number of deletions, insertions and substitutions over the total number of bases.

The average error rate before denoising (i.e. after using the *make.contigs* command in mothur) was 0.0005 and 0.0006 for V4 (MOCK1 and MOCK2 respectively), 0.0071 and 0.0050 for V4-V5 (MOCK1 and MOCK2 respectively) 0.0026 and 0.0015 for V3-V4 (MOCK1 and MOCK3 respectively). The fact that the error rate of V4 was up to an order of magnitude lower than both other regions is not surprising since the V4 amplicon fragment consists of two completely overlapping reads, as such assuring a two-fold prediction for each nucleotide.

Table 3:2. Overview table comparing error rates of the samples various preprocessing either treated with UNOISE (after USEARCH preprocessing), without applying a denoising algorithm, or after applying Pre-cluster or after applying IPED (after mothur preprocessing). Due to the difference preprocessing steps applied in USEARCH and mothur, the number of reads removed differ, where around 53% and 39% of reads are removed in respective order.

Variable region	Sample ID	Error rates			
		USEARCH + UNOISE	Mothur (make.contigs)	Mothur + Pre-cluster	Mothur + IPED
V3-V4	130403	0.0003	0.0026	0.0013	0.0002
	130417	0.0004	0.0023	0.001	0.0003
	130422	0.0008	0.0028	0.0017	0.0008
	M1	0.0006	0.00149	0.0007	0.0004
	M2	0.0007	0.0015	0.0008	0.0006
	M3	0.0005	0.0014	0.0007	0.0005
V4	130403	0.00011	0.00056	0.00013	0.0001
	130417	0.00009	0.00051	0.0001	0.00008
	130422	0.00009	0.00049	0.0001	0.00008
	v4.l.1	0.00002	0.00061	0.00008	0.00004
	v4.l.05	0.00002	0.00068	0.0001	0.00004
V4-V5	130403	0.003	0.0084	0.0055	0.0022
	130417	0.0029	0.0069	0.0041	0.002
	130422	0.0026	0.006	0.0033	0.0016
	v4.v5.l.1	0.0082	0.0066	0.0061	0.0041
	v4.v5.l.11	0.0084	0.0033	0.0034	0.0031
Average	All samples	0.0018	0.0029	0.0018	0.001

When comparing the output of IPED with the raw error rates, our algorithm was able to reduce on average the error rate with 72% (individual values for different regions and

runs are varying between 28% and 94%). When benchmarking those results with UNOISE and Pre-cluster, UNOISE was able to reduce the error rate on average by 52% (individual values varying between an increase of the error rate with 66% and a decrease of 97%) while Pre-cluster was able to reduce it by 51% (individual values varying from 4% to 86%), see **Table 3:2** for all details. On average (averaged over all mock communities), IPED diminished the error rate to 0.0010, while UNOISE and Pre-cluster reduced the overall error rate to the same value of 0.0018. However, compared to other denoising algorithms, the effect of IPED is more pronounced for those regions with no complete overlap between both paired-end reads (i.e. region V3-V4 and V4-V5). Importantly, it should be noted that UNOISE (as implemented in USEARCH) results in an additional loss of on average 13% of sequencing data due to its more stringent pre-processing steps, as illustrated in Additional Information 3:4.

The same trend in lowering the error rate was observed when running IPED on the MOCK4 dataset. Indeed, when both reads are almost completely overlapping (contig lengths ranging between 253 and 292), IPED was able to reduce the error rate from 0.0041 to 0.0032. This effect was more prominent when dealing with contigs with a smaller overlap between both paired-end reads (contig length ranging between 330 and 450), showing a decrease in the error rate from 0.0065 to 0.0033 (Additional Information 3:5). However, the error rate reported for this dataset is significantly higher than what is expected for Illumina MiSeq sequencing data. This can be explained via presence of contamination(s), as such, caution should be taken when extrapolating those results.

Plotting the error rates for the MOCK1 and MOCK2 datasets versus their position in the amplicon, indicated that the beneficial effect of IPED is mainly situated in the uniquely covered region of the second read (i.e. those positions not overlapping with the first read), and to a lesser extent also the overlapping part (i.e. those positions

sequenced twice: a first time via read one, and a second time via read two), see **Figure 3:2** for the V3-V4 and V4-V5 region of MOCK1, and Additional Information 3:6 for more details.

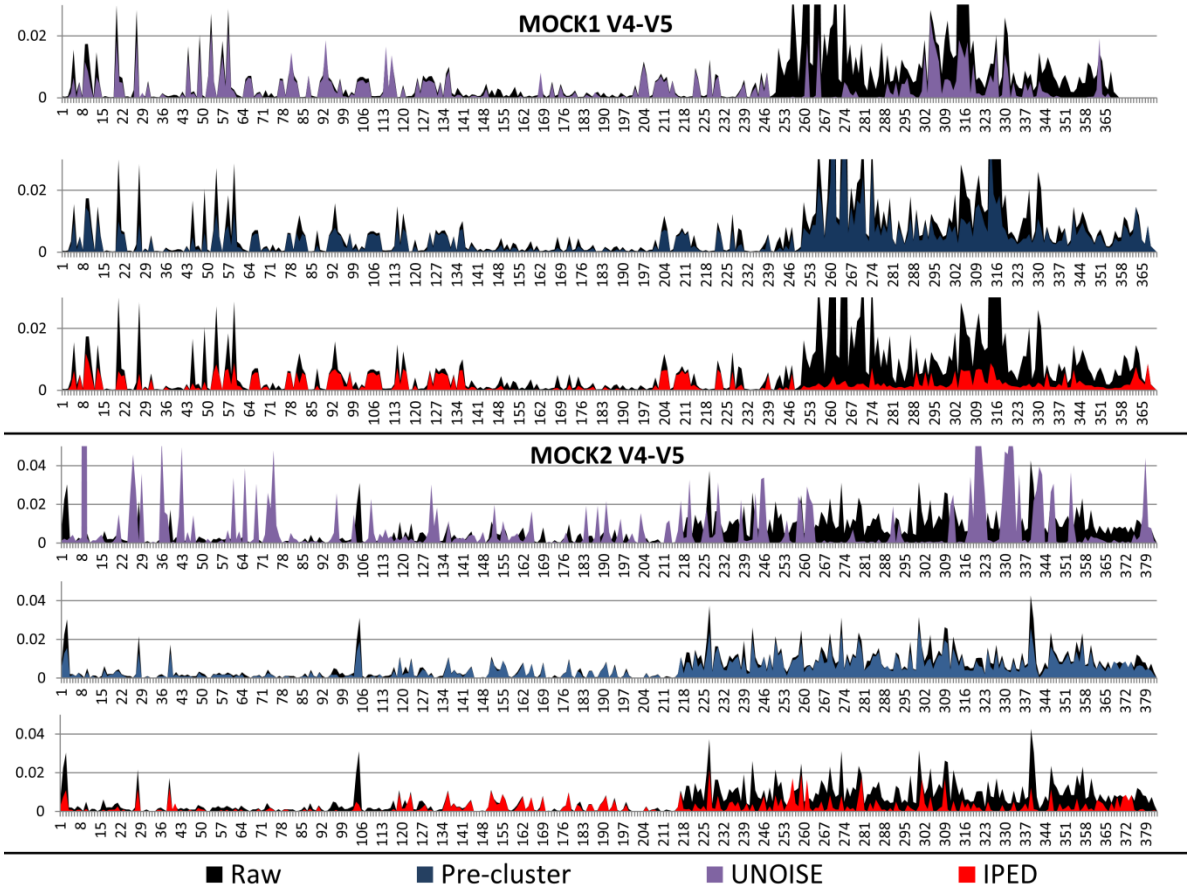


Figure 3:2. Plot showing the error rate versus the position in the read after being treated with Pre-cluster, UNOISE, and IPED. The raw error rates (i.e. without applying a denoising algorithm) are coloured black.

As stated in the introduction, a plethora of algorithms is available for assembling paired-end reads into one amplicon fragment. Even though those algorithms are not denoising algorithms in the strict sense, they can have an impact on the error rate of the resulting fragment. For this comparison, error rates were calculated for MOCK1, MOCK2 and MOCK3 after running different assembly algorithms, being *fastq_mergepairs* (USEARCH), *make.contigs* (mothur), *join_paired_ends* (QIIME) and *PEAR*, which resulted in error rates of 0.0027, 0.0029, 0.0031 and 0.0097 respectively

(see Additional Information 3:4). Important to notice is that the number of reads retained is correlated with the error rate: PEAR returned the highest error rate, however it managed to retain 86% of the reads, while USEARCH reached the lowest error rate but removed more than 53% of the data. Anyhow, it is clear from those data that the error rates obtained using those assembly algorithms did not come close to the error rates obtained with IPED (run on the output of the mothur *make.contigs* step) i.e. 0.0010. Similar effects were obtained by including IPED after the QIIME assembly step, leading to an error rate of 0.0014, which was also significantly lower than the error rate obtained with the assembly steps solely (data not shown). Those data suggest that whatever currently available assembly algorithm is used, running IPED afterwards will still have a beneficial effect on the error rate.

To investigate the extra computational cost related to IPED, the calculation time was registered for all three samples of the MOCK1 dataset covering the V4 region, where each sample was subsampled to 6000 unique reads. When one processor (single Intel Xeon E5-2640 2.50 GHz CPU) was used for each sample (i.e. a total of three processors), IPED required 70 seconds for running all three samples, while Pre-cluster and UNOISE could end the analysis in 14 seconds and 12 seconds respectively. Similar relative differences in calculation time were also observed for other MOCK1, MOCK2 and MOCK3 samples (see Additional Information 3:7).

Impact of denoising algorithms on the OTU clustering

As the negative effect of sequencing errors has an influence on the amount of spurious OTUs, ideally an improvement at the level of denoising step should be reflected in a decrease of the number of OTUs. Although the OTU clustering step is influenced by the number of reads and level of complexity in the mock samples, (Kozich et al. 2013), it has been used by others as a metric for sequence quality (Schloss et al. 2011; Kozich et al. 2013; Huse et al. 2010; Reeder & Knight 2010; Mysara, Leys, et al. 2015; Edgar

& Flyvbjerg 2015; Quince et al. 2011; Kunin et al. 2010). In order to get an idea to which extent IPED, UNOISE and Pre-cluster have a beneficial effect on the OTU clustering step, sequencing data denoised by either one of the three approaches were clustered using the average neighbourhood hierarchical clustering algorithm via the "cluster" command (as implemented in mothur), and subsequently compared with the number of OTUs obtained when no denoising algorithm was applied. As the amount of species present in both mock communities is known, in the most ideal scenario the amount of OTUs returned should be 20, 19 and 12 for MOCK1, MOCK2 and MOCK3 respectively. It is important to emphasize that any undetected chimera or possible contamination would lead to an inflation of the number of the OTUs. However, as their effect is the same for all tools, we assume that the number of OTUs still provides a good indication of the performance.

The average number of OTUs produced when the denoising step was omitted returned on average 109, 64 and 127 OTUs for the V4, V3-V4 and V4-V5 regions respectively (combined results of MOCK1 and MOCK2, except for the V4-V5 region of MOCK1 – see further). IPED was able to reduce these numbers to an average of 94, 48 and 81 while Pre-cluster resulted in 110, 66 and 118 OTUs respectively and UNOISE resulted in 120, 15 and 363 OTUs respectively. Again, it is important to highlight the impact of the strict pre-processing approach followed by UNOISE, resulting in a removal of almost all of the reads in MOCK1 V3-V4 samples in this pipeline, and therefore returning a very low number of OTUs for the UNOISE approach. Similarly, for MOCK3, the number of OTUs for the non-denoised data was 90, while integrating IPED, UNOISE or Pre-cluster in the preprocessing pipelines led to 84, 89 and 107 OTUs respectively. Altogether, this analysis showed a more beneficial effect of IPED on the OTU clustering step than Pre-cluster and UNOISE (see Additional Information 3:8). Concerning the number of OTUs for the V4-V5 region of MOCK1, it was not

possible to calculate the number of OTUs due to high memory requirements, leading to the exclusion of these datasets from the OTU analysis.

It should be noted that all mock samples analysed in this work contain a high number of reads per sample (on average more than 500,000), which is significantly higher than the number of sequences obtained for most real-life microbial diversity studies. In order to work with more realistic numbers, we rarefied the datasets to 5,000 – as proposed in Kozich et al. (2013)– and 25,000 reads per sample. Again, IPED outperformed UNOISE and Pre-cluster when applied on the rarefied MOCK1, MOCK2 and MOCK3 samples (see Additional Information 3:8). Moreover, similar results were obtained upon using the UPARSE clustering algorithm on both complete and rarefied datasets (see Additional Information 3:8).

However, the analysis performed above starts from an ideal situation, since all chimeras can accurately be removed using the reference sequences from the species present within the mock community. Additionally, the species present in the mock communities are well-known species, incorporated in the reference alignment database, as such resulting in an accurate alignment. To get an idea on the effect of the different denoising tools in case of a more realistic scenario, i.e. using a regular chimera removal algorithm and the presence of species not represented in the reference alignment database, we applied a *de novo* chimera detection tool CATCh (Mysara, Saeys, et al. 2015) on the MOCK3 dataset in order to remove chimeric sequences from the mock community. Additionally we removed the corresponding sequences of the species represented in the mock community from the 16S rRNA gene reference alignment database, together with any other sequence showing an identity higher than 97% to any of those twelve species. As reported in Additional Information 3:10, IPED was able to outperform Pre-cluster and UNOISE with a reduction of the error rate in the same range as reported earlier, and additionally led to the lowest number of OTUs.

In order to check the effect of the denoising algorithm on the OTU clustering quality, sequencing data were analysed using a preprocessing pipeline where the only varying factor was the denoising algorithm (IPED, Pre-cluster or UNOISE). This way we could assess whether the anticipated species of the mock community could be retrieved. Applying Pre-cluster and UNOISE led to less accurate clustering results as reads originating from the same species where more frequently scattered over different OTUs. In general we can conclude that applying IPED has a beneficial effect on the OTU clustering step when compared with the Pre-cluster and UNOISE results, since for all mock samples the number of OTUs produced with IPED was the closest to the actual number of species. Details on the number of OTUs are given in Additional Information 3:8.

As a proof of principle, IPED was applied on a real dataset (i.e. a non-mock dataset) to emphasize the effect on a more complex dataset. However, it is important to stress that unfortunately for those real-life datasets no error rates could be calculated, forcing us to revert to the number of OTUs as evaluation criterion. Despite the fact that this criterion is inferior to the error rate, it has been used in previous publications to evaluate the performance of denoising tools (Huber et al. 2007; Sogin et al. 2006; Kunin et al. 2010). In this dataset, presented in Kozich et al. (2013), murine fecal samples of mice were used to assess the shifts of the microbial community after weaning at two different stages: early (0-9 days) versus late (141-150 days) after weaning. IPED was able to reduce the number of spurious OTUs, as illustrated by the rarefaction curves, and produced a more clear separation of clusters of late versus early stage samples when visualized in principle coordinate analysis (PCoA) biplots (see Additional Information 3:9).

3.5. Discussion

New sequencing technologies have revolutionized the assessment of microbial diversity via amplicon sequencing. However, each of the currently available high-throughput sequencing platforms suffers from sequencing errors originating from the sequencing technology itself (which is different from PCR point errors). In order to prevent the inflation of artificial OTUs due to these sequencing errors, different algorithms have already been developed for the correction of sequencing errors in 454 Pyrosequencing data, for example SLP (Huse et al. 2010), AmpliconNoise (Quince et al. 2011) and Denoiser (Reeder & Knight 2010). However, assessment of bacterial diversity using the Illumina MiSeq technology is now the standard, as it offers high throughput in combination with an acceptable read length. Recently, the SLP-based algorithm Pre-cluster (available as *pre.cluster* in mothur) and UNOISE (available in USEARCH) have been proposed as denoising tools in Illumina MiSeq specific pipelines (Kozich et al. 2013; Edgar & Flyvbjerg 2015). In this work, we introduced IPED as the best denoising tool specifically oriented towards Illumina MiSeq 16S rRNA gene paired-end reads.

IPED was shown to outperform Pre-cluster and UNOISE, as observed on the three mock datasets where our newly introduced algorithm could on average correct double the amount of sequencing errors. This effect seems to be less pronounced in those paired-end reads having a complete overlap between both reads, as every nucleotide position in the amplicon is covered twice, once by the first read, and once by the second read. Therefore, the added effect of IPED is smaller in those latter cases.

Moreover, reducing the error rate has a significant effect on the quality of the reads in the OTUs. Adding an error-correction step before running the OTU clustering algorithm, led to a very close correspondence between the number of OTUs returned, and the true number of species known to be present in the mock communities. Such a significant correspondence could not be obtained when omitting the denoising

algorithm in the amplicon sequencing preprocessing pipeline, or via using any other denoising tool. However, caution should be given when extrapolating this to real-life environmental communities, since the diversity linked to the latter samples will be significantly higher than in the tested mock communities. Despite this increased complexity, running IPED on real biological samples still showed a clear improvement, which was visualized using rarefaction curves showing a clear decrease in the number of OTUs. Moreover, a more accurate correlation was found between biologically related samples when comparing the OTU tables produced where IPED was integrated into the workflow, as shown in the results by producing denser clusters distinguishing two different biological conditions.

Where Pre-cluster and UNOISE have an impressive speed, IPED needs more calculation time due to the machine learning classifier required in the first step of IPED. However, as seen in the results, IPED led to a more pronounced improvement in accuracy compared to both of them.

At this stage we only tested our IPED algorithm on mock datasets containing paired end reads that are at least partially overlapping, or in some cases completely overlapping. Recent papers suggest the usage of primer pairs for amplicon sequencing producing paired-end reads which are not overlapping at all, as this approach allows flexibility in development of PCR primers and selection of the hypervariable regions. This way those primers can be selected that allow the most optimal distinction for a specific type of sample (Jeraldo et al. 2014). Within this area IPED can have a more pronounced effect on the final results as our algorithm was shown to be most effective in the non-overlapping part of the second read, which in such a case would mean the complete second read. IPED has only been tested on 16S rRNA amplicon sequencing data. In principle our tool can be used for any amplicon sequencing dataset, such as 18S rRNA, 23S rRNA or 28S rRNA, whenever a reliable reference alignment dataset is

available. IPED was developed to be applied after the mothur *make.contigs* command; yet, further adjustments are needed to make it compatible with other paired-end assemblers. Preliminary data showed that IPED was able to reduce the error rate of QIIME to the same extent.

3.6. Conclusion

We have presented in this work the denoising algorithm IPED specifically developed for Illumina MiSeq 16S rRNA amplicon sequencing data. IPED obtains a better performance on mock datasets compared with the available alternatives Pre-cluster and UNOISE, and on average can correct double the amount of errors compared to both algorithms. The beneficial effect of this improved denoising was reflected in more accurate OTU clustering results.

Chapter 4

CATCh: an Ensemble Classifier for Chimera Detection in 16S rRNA Sequencing Studies

Mohamed Mysara[1,2,3,4], Yvan Saeys[4,5], Natalie Leys[1], Jeroen Raes2,[3,6], Pieter Monsieurs[1]

[1] *Unit of Microbiology, Belgian Nuclear Research Centre (SCK•CEN), Mol, Belgium*
[2] *Department of Bio-engineering Sciences, Vrije Universiteit Brussel, Brussels, Belgium*
[3] *VIB Center for the Biology of Disease, Leuven, Belgium*
[4] *Data Mining and Modeling group, VIB Inflammation Research Center*
[5] *Department of Respiratory Medicine, Ghent University Hospital, Ghent, Belgium*
[6] *Department of Microbiology and Immunology, REGA institute, KU Leuven*

This chapter is based on the following publication:

Mysara M, Saeys Y, Leys N, Raes J, Monsieurs P. (2015). CATCh, an ensemble classifier for chimera detection in 16S rRNA sequencing studies. Appl. Environ. Microbiol. 81:1573–84

4.1. Abstract

In ecological studies microbial diversity is nowadays mostly assessed via the detection of phylogenetic marker genes such as 16S ribosomal RNA. However, PCR amplification of these marker genes produces a significant amount of artificial sequences often referred to as chimeras. Different algorithms have been developed to remove these chimeras, but efforts to combine different methodologies are limited. Therefore, two machine learning classifiers (CATCh reference and CATCh *de novo*) were developed by integrating the output of existing chimera detection tools into a new, more powerful method. When comparing our classifiers with existing tools either in reference based or *de novo* mode, a higher performance of our ensemble method is observed on a wide range of sequencing data, including simulated, 454 Pyrosequencing and Illumina MiSeq datasets. Since our algorithm combines the advantages of different individual chimera detection tools, our approach produces more robust results when challenged with chimeric sequences having a low parent divergence, short length of the chimeric range and a varying number of parents. Additionally, it could be shown that integrating CATCh in the preprocessing pipeline has a beneficial effect on the quality of the clustering in operational taxonomic units.

4.2. Introduction

A powerful method to assess microbial diversity of diverse environments is the identification of specific phylogenetic marker genes like the 16S ribosomal RNA gene (rRNA) for Bacteria and Archaea, or the internal transcribed spacer (ITS) for Fungi. However, because of the limitations of most current sequencing technologies, PCR amplification of the targeted DNA is an unavoidable step in this approach. During this PCR amplification, chimeras might be created due to incomplete extension. If this incomplete PCR fragment acts as a primer in the next round of the PCR (by binding to the template DNA of a different species), a chimeric sequence originates which is consisting of two or more fragments amplified from the DNA of distinct species (Ashelford et al. 2005; Lahr & Katz 2009; Porazinska et al. 2012; Thompson et al. 2002; Wang & Wang 1996; Wang & Wang 1997).

Chimeras introduced via this mechanism will be propagated through the PCR reaction as any other DNA sequence (Ashelford et al. 2005; Judo et al. 1998; Odelberg et al. 1995; Smyth et al. 2010). Consequently, these chimeras will result in an artificial inflation of the microbial diversity in a biodiversity analysis using next-generation sequencing approaches on marker genes like the 16S or 18S rRNA gene or fungal ITS (Edgar et al. 2011). Indeed, since chimeras are PCR errors and not sequencing errors, they cannot be recovered using regular data denoising approaches (Logares et al. 2012; Quince et al. 2011), and can falsely be interpreted as unique sequences representing novel species.

A significant number of chimeras has been identified in curated databases in a proportion that can reach up to 46% (Ashelford et al. 2005). Even after being treated with traditional chimera detection tools, chimeras are continuously being detected in highly valuable and frequently consulted databases like RDP (Cole et al. 2014), SILVA

(Quast et al. 2013), and Greengenes (DeSantis, Hugenholtz, Larsen, et al. 2006). Likewise, the percentage of chimeric sequences in the unique amplicon pool of PCR amplified samples might reach up to values higher than 70% (Lahr & Katz 2009; Quince et al. 2011).

In general, two classes of chimera detection tools can be distinguished: reference based and the more recent *de novo* methods. Reference based methods basically screen the sequences potentially containing chimeras against a curated reference database with chimera-free sequences. This approach has been implemented in the first generation of chimera detection tools such as Pintail (Ashelford et al. 2005) and Bellerophon (Huber et al. 2004). A major improvement of the reference based approaches was achieved via the introduction of ChimeraSlayer (Haas et al. 2011), that uses 30% of each end as a seed for searching a reference dataset, finding the closest parent (if any), performing alignments, and scoring to the candidate parents. Although ChimeraSlayer was found to outperform any of the previous tools, it was not able to detect chimeras with a small chimeric range (i.e. the shortest region produced by one of the parents, Wright et al. 2012). The reference based mode of a new chimera detection tool called UCHIME was built upon the implementation of ChimeraSlayer and outperformed it on almost all investigated datasets (Edgar et al. 2011). In the reference based UCHIME, query sequences are divided into four non-overlapping segments and searched against a reference database. Both of the three-way alignment tools (ChimeraSlayer and UCHIME reference based) were reported to have a lower accuracy when compared to DECIPHER (Wright et al. 2012) in cases where the algorithms were challenged with a dataset containing chimeric sequences with short chimeric range and long sequence lengths. The DECIPHER algorithm is a search-based algorithm that splits the query sequence into different fragments, and analyses whether those fragments are uncommon

in the reference phylogenetic group where the query sequence is classified. If a significant amount of fragments is assigned to a phylogenetic group different from the complete query sequence, the sequence is classified as being chimeric. However, in general an objective comparison between different chimera detection tools is difficult as each of the algorithms is biased toward its own test data and has been proven to outperform other tools when applied to their own dedicated data.

De novo methodologies are generally based on the fact that parents of any chimeric sequence have gone through at least one more PCR cycle than chimeric sequences. Three such tools, Perseus (Quince et al. 2011), UCHIME *de novo* (Edgar et al. 2011) and the *de novo* ChimeraSlayer implementation are integrated in mothur (Schloss et al. 2009). Perseus was trained using real 454 Pyrosequencing data while UCHIME *de novo* used simulated data. Recently, the UPARSE pipeline was released (Edgar 2013), combining in one step chimera detection with clustering of sequencing reads into operational taxonomic units (OTUs).

Both reference based and *de novo* methodologies have their advantages and disadvantages. In situations dealing with well-studied environments, the reference based approaches were found to be very effective in distinguishing between chimeras and chimera-free (parents) sequences. This efficiency is assumed to be lower when dealing with less well known environments, which is where the need for *de novo* approaches originated. However, most of the *de novo* approaches depend on redundancy differences between chimeras and parents, assuming that the number of parent sequences has to be at least one time more redundant than their corresponding chimeric sequences. This requires that data abundances have been reported with high accuracy. Practically this might be problematic as some of the sequences are removed due to noise removal, score filtration or any other preprocessing step. As each approach

has its pros and cons, reference based as well as *de novo* approaches were taken into account in the analyses described below. A schematic overview comparing the various chimera detection tools is available in Additional Information 4:8.

In this work we present a chimera detection tool consisting of a machine learning classifier called CATCh (Combining Algorithms to Track Chimeras), which is able to discriminate between chimeric and non-chimeric sequences based on a specific set of input data (called features in the context of machine learning). In this work, we do not use as input data the sequence read characteristics, but rather the results (e.g. scores) of different individual chimera detection tools mentioned above, and integrate them into one prediction. All different tools are run separately, and their output values are combined and processed by the classifier in order to give a prediction whether a read is a chimera or not. Developing this machine learning method consists of three stages. First, the necessary input features (i.e. output values of the different chimera detection tools) are identified. In the second step, the classifier is trained via a supervised learning approach. In this training step, the classifier learns to make a correct prediction based on example input data which is in our case training data consisting of the output features of a set of sequences reads obtained from different chimera detection tools, together with their correct classification (i.e. whether this read is a chimeric sequence or not). In the third step, the trained classifier can be used to predict chimeric sequences in new, previously unseen data (i.e. data that did not belong to the training data). By feeding the outputs of the different individual chimera detection tools into the classifier, CATCh is able to classify them into chimeric and chimera-free subsets. As two different types of chimera detection tools exist, either reference-based or *de novo*, we also developed two different versions of CATCh. In order to illustrate

its performance, CATCh (reference based as well as *de novo*) was benchmarked against other chimera detection tools using various publically available benchmark data.

4.3. Materials and methods

4.3.1. Included chimera detection algorithms

As mentioned in the introduction, the strategy of CATCh consists of consolidating the output of different chimera detection tools into one highly reliable prediction. As such, a wide range of different reference based and *de novo* tools has been integrated into CATCh, and each of the tools used is discussed below.

When running the reference based algorithms, the implementations of the tools as available in mothur -version 1.28.0- (Schloss et al. 2009) were used for UCHIME -reference mode- (Edgar et al. 2011), ChimeraSlayer -reference mode- (Haas et al. 2011) and Pintail (Ashelford et al 2005), each of them using the default parameters. For DECIPHER (version 1.10.0), that does not support parallelization, we increased its speed by implementing a forking approach – as already suggested in the original paper – making it possible to split up the run of DECIPHER over an arbitrary number of cores (Wright et al. 2012). All these reference based algorithms use a database of chimera free reads as a reference: UCHIME, ChimeraSlayer and Pintail use the Gold reference database (available at the Broad Microbiome Utilities website version microbiomeutil-r20110519) and DECIPHER uses a "good quality" filtered and chimera curated version of RDP [release 10, update 22] (Wright et al. 2012). DECIPHER has two modes, depending on the length of the sequences, either full length (fs_decipher) or short length (ss_decipher). In our experiments we adjusted these parameters according to the length of the input sequences of each database.

Similarly, for the *de novo* based algorithms, the implementation of the algorithms as available in mothur (Schloss et al. 2009) were used for UCHIME (Edgar et al. 2011) -*de novo*-, ChimeraSlayer -*de novo*- (Haas et al. 2011) and Perseus (Quince et al. 2011). For integrating the different *de novo* based algorithms into the CATCh *de novo* classifier, the default parameter settings for each of the three tools were used.

4.3.2. Building CATCh

Two different CATCh classifiers were developed – one reference and one *de novo* model – each of them integrating the results (called features below) of different individual chimera detection tools into a chimera prediction tool with higher performance.

For the reference based CATCh classifier, we included the following input features: 1) the calculated score and the final decision (i.e. chimeric or non-chimeric) for UCHIME, 2) the calculated score, and final decision for ChimeraSlayer, 3) the score, standard deviation and final decision for Pintail, and 4) final decision for DECIPHER. The final decision (i.e. whether a read is predicted as chimeric or not) for the individual chimera detection tools, is depending on the chosen cut-off value for which we took the default value. For the *de novo* CATCh classifier, both the score and the final decision for 1) UCHIME *de novo*, 2) ChimeraSlayer *de novo* and 3) Perseus were selected as input parameters. Based on the chimera prediction results for each of these individual tools, two classifiers were built (reference and *de novo*) integrating the different output values produced by these tools into one final score.

Different mathematical functions – called kernels – have already been developed for solving classification problems. To develop the CATCh classifier, we had to select the most optimal kernel based on their performance on the training data. For this purpose,

the training data – consisting of the output of different chimera detection tools, together with the chimeric or non-chimeric label – was split into two subsets: a subset to train and a subset to test the classifier. The first one is used during the learning step applied for each of the kernels to build a classifier, the second dataset is used for evaluating the performance of the different kernels. An overview of the tested kernels available in WEKA (Hall et al. 2009) is given in Additional Information 4:9 for the reference based and the *de novo* implementations.

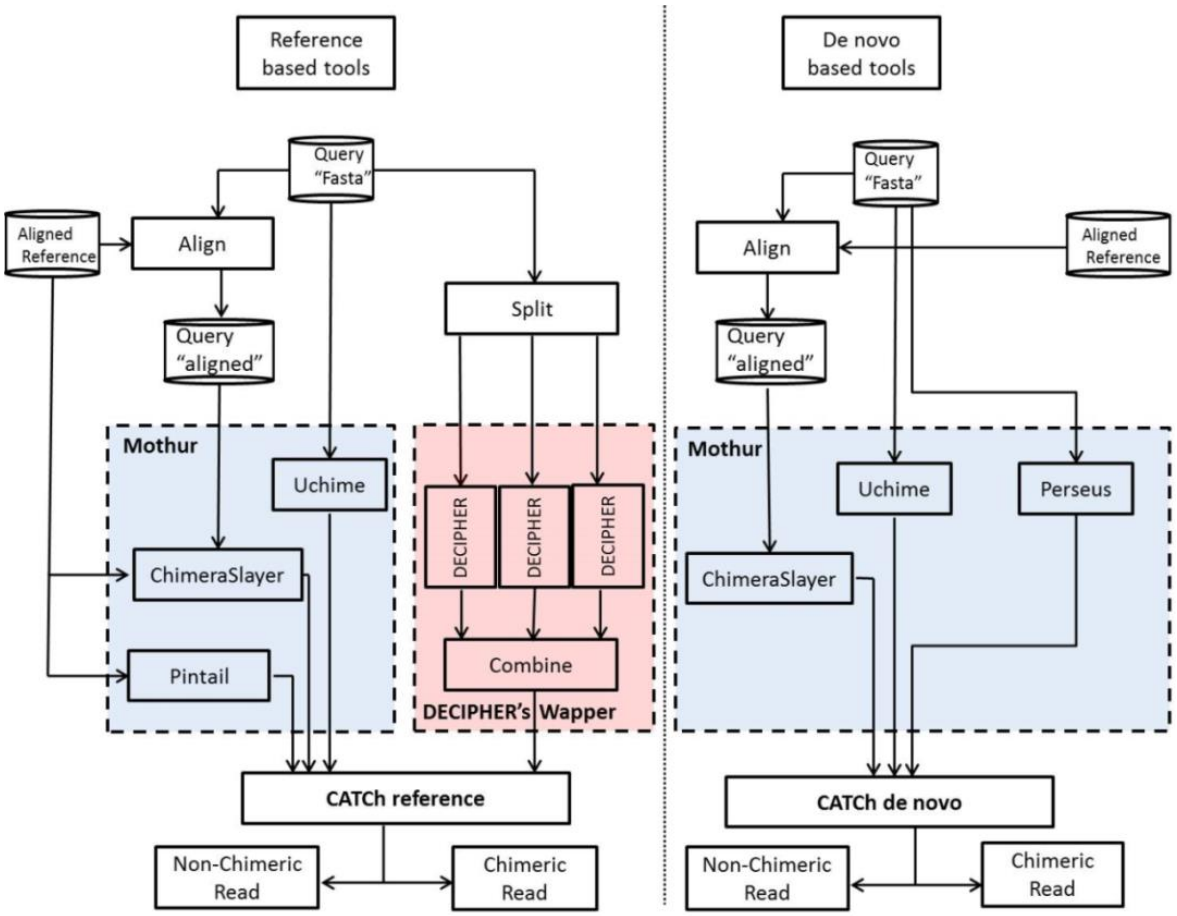


Figure 4:1. Illustration of different steps and tools included in CATCh reference and *de novo*.

For all model parameters, the WEKA standard values were used, version 3.7.1 (Hall et al 2009). A schematic overview of the developed methodology (reference based and *de novo*) is given in **FIGURE 4:1**. The CATCh software and its documentation is available

via http://science.sckcen.be/en/Institutes/EHS/MCB/MIC/Bioinformatics/CATCh. It has been tested on Mac and Linux (RHEL-derived distributions). On the CATCh website a manual together with the training data can be downloaded, explaining how CATCh was trained and tested using the WEKA software. This software is distributed under the terms of the GNU General Public License as published by the Free Software Foundation.

4.3.3. Training and testing datasets

As a training dataset to build the CATCh classifiers, the denoised output of the Titanium dataset as published in Quince et al. (2011) was used. These data are grouped into 91 parent sequences (chimera-free) and 176 chimeric sequences (obtained directly from C. Quince). For the reference based model, we used the Titanium dataset (see below) to train as well as test different classifiers (splitting to a 3:1 ratio). For the *de novo* classifier the whole Titanium dataset was used for training and a separate dataset (F01QS4Z01_rep2_v13, a subset of MOCK2-b, see below) was used for testing, as complete information on the redundancy of the reads is essential for all *de novo* algorithms.

4.3.4. Validation datasets

For benchmarking our tool against other available chimera detection tools, two types of data were used, i) simulated data containing artificial chimeric sequences generated using tools like CHSIM (Edgar et al. 2011), and ii) publically available real sequencing data (454 Pyrosequencing, Illumina MiSeq and PacBio SMRT sequencing).

For the first dataset, we randomly selected sequences from the GOLD database (described in (Haas et al. 2011)), and used them as parent sequences to artificially produce chimeras using the CHSIM tool (Edgar et al. 2011). By trimming these parent

and chimeric sequences to a randomly selected length within 50-99% range of the actual length, we obtained a dataset consisting of 1532 chimera-free (parents) and 1308 chimeric sequences, referred to as *Simu1*.

Additionally, three publically available datasets were used for the evaluation of different previously published reference based chimera detection tools: SIMM (Edgar et al. 2011), Decipher (Wright et al. 2012) and *SIM2* (Haas et al. 2011) respectively called *Simu2, Simu3* and *Simu4* in the rest of the paper, see **Table 4:1**. To challenge the chimera detection tools to predict chimeras in more experimentally relevant situations obtained via high-throughput sequencing technologies (sequencing errors, which can rise up to 1%) as well as to simulate biological variation of reads corresponding to novel species, we used the previously published *SIMM* datasets with 1% and 5% insertion and deletion, and *SIM2* datasets with 1% and 5% insertion, deletions, indels and substitutions (referred to as *Simu*2_Mutation and *Simu*4_Mutation respectively).

For benchmarking our *de novo* classifier, the uneven mock dataset as reported in Quince et al. (2011) and initially introduced in Turnbaugh et al. (2010) was used. This dataset – covering the V2 region of the 16S rRNA gene – is sequenced in triplicate and consists of 3036 sequences, from which those sequences labelled as "anomalies" (n=225) were removed, leaving 2565 chimeric sequences and 246 non-chimeric sequences in this dataset, called MOCK1.

The HMP-MOCK dataset, part of Project SRP002397 in NCBI Sequence Read Archive (SRA), contains 454 Pyrosequencing data of 16S rRNA amplicons of 21 bacteria, consisting of 10 runs, covering 3 regions (V1-V3, V3-V5, V6-V9) with 3 replicates per region per run resulting in a total of 90 samples (Schloss et al. 2011). This dataset was treated using mothur (Schloss et al. 2009) (version 1.28.0) with two different denoising strategies – quality trimming (Schloss et al. 2011) and flow denoising (AmpliconNoise,

(Quince et al. 2011)) – producing two separate datasets (P. Schloss, personal communication). These two datasets were named MOCK2-a and MOCK2-b for quality trimmed and AmpliconNoise treated datasets respectively.

The MiSeq mock dataset (called MOCK3) consists of the 16S rRNA amplicons of a mock community containing 21 bacterial isolates, and covering the V3-V4 and V4 region (each region sequenced in quadruplicate), which were extracted from the Sequence Read Archive (SRA082708) as presented by Kozich et al. (2013). The sequencing data were preprocessed following the mothur MiSeq SOP (accessed August 3th 2014).

For assessing the computational resources required for different sequencing platforms, also a (non-mock) PacBio SMRT sequencing dataset was analysed. This dataset covers the V1-V3 and V2-V3 region of the 16S rRNA gene and was extracted from the Sequence Read Archive (SRA056302) as presented by Marshall et al. (2012). All 16 samples were preprocessed using mothur (Schloss et al. 2009), using version 1.28.0, as described in the corresponding paper (Marshall et al. 2012), i.e. removing sequences with an average quality score of lower than 25, an anomalous length (<300 or >615 bp), an ambiguous base, homopolymers with a length higher than 8 or more than one mismatch in the barcode or primer.

Additional to testing the performance of chimera detections tools on 16S rRNA amplicon sequencing data, the performance on two datasets covering regions in the eukaryotic 18S rRNA gene were included in this work. The first dataset – called MOCK4 – consists of a mock community of either 12, 24 or 48 species from closely or distantly related nematodes, as described in Fonseca et al. (2012). Each sample was sequenced in five replicates, resulting in a total of 30 samples (SRA accession number

SRA043810). The data were preprocessed as described in the initial paper (Fonseca et al. 2012) using AmpliconNoise.

A second 18S rRNA mock community (MOCK5) – covering the V1-V3 and V4-V6 region – was obtained from the publication of Morgan et al. (2013). The data downloaded from http://research.csiro.au/software/amplicon-pyrosequencing-denoising-program/ consist of 30 sequencing datasets and were preprocessed by removing reads with one or more ambiguous nucleotides and homopolymers longer than 8 nt. An overview of all simulated and mock datasets used in this study is given in **Table 4:1**.

4.3.5. Evaluation parameters

For evaluation of the performance of the different chimera detection algorithms, we adopted four parameters: sensitivity $\left(\frac{TP}{TP+FN}\right)$, specificity $\left(\frac{TN}{TN+FP}\right)$, Matthews Correlation Coefficient (MCC) $\frac{TP\times TN-FP\times TN}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}}$, and accuracy $\left(\frac{TP+TN}{TP+FN+TN+FP}\right)$ where false positives (FP) and false negatives (FN) represent the number of sequences falsely predicted as chimeric and non-chimeric respectively, while true positives (TP) and true negatives (TN) stand for the number of sequences correctly predicted as chimeric and non-chimeric respectively. MCC is a correlation coefficient between the observed and predicted binary classifications giving an idea on the balance between the specificity and sensitivity balance. It returns a value between -1 (negative correlation i.e. performing worse than a random prediction), 0 (no correlation i.e. performing as random predictor) and +1 (positive correlation i.e. perfect prediction).

Table 4:1. Detailed description of each of the datasets used in this work either for training or testing.

Name	Non-chimeric	Chimeric	platform	Samples	Genomes	Region	Ref.
Training	91	176	454 Titanium	1	91	V4-V5	(Quince et al. 2011)
Simu1	1532	1308	simulated	1	-	-	(Mysara et al. 2015)
Simu2	86	900	simulated	1	-	-	(Edgar et al. 2011)
Simu3	11000	4000	simulated	1	-	-	(Wright et al. 2012)
Simu4	4769	2500	Simulated	1	-	-	(Haas et al. 2011)
MOCK1	246	2565	454 GS-FLX	3	67	V2	(Quince et al. 2011)
MOCK2-a	5447	13205	454 Titanium	90	21	V1-V3/V3-V5/V6-V9	(Schloss et al. 2011)
MOCK2-b	8302	22131	454 Titanium	90	21	V1-V3/V3-V5/V6-V9	(Schloss et al. 2011)
MOCK3	11958	13612	Illumina MiSeq	8	21	V3-V4/V4	(Kozich et al. 2013)
MOCK4	2050	5326	454 GS-FLX	30	48	V1-V3	(Fonseca et al. 2012)
MOCK5	82426	16317	454	30	16	V1-V3 / V4-V6	(Morgan et al. 2013)

4.4. Results

In this work, two classifiers were developed, one for reference based approaches and a second one integrating all *de novo* approaches. For both CATCh classifiers, we used the output of the individual chimera detection tools as input features for the calculation of a combined chimera prediction score. Constructing a machine learning classifier generally consists of three stages: 1) identifying the necessary input data, called feature selection, 2) training and testing the classifier, and 3) evaluating the performance of the classifier. In the first two sections of the Results, we describe respectively the feature selection, and training and testing the classifier respectively. The remaining sections are dedicated to the evaluation of the CATCh classifier.

4.4.1. Feature selection

The feature selection step is required to identify the optimal set of features needed for the development of a robust classifier. The question answered here is whether all input features as described in the Material and Methods section (and by extension all individual chimera prediction tools) are needed to obtain a good classifier. This step is important as a reduction of the number of input features might have a beneficial effect

on the accuracy of the classifier as well as the computational calculation time. However, based on the Principal Component Analysis (PCA) on the training data (Additional Information 4:1 and Additional Information 4:10), all input features were needed to explain 95% of the variation in the data. This also implies that the results of all individual chimera detection tools need to be retained as input date for the classifier. This observation was further confirmed by a dedicated experiment where we left out one of the chimera prediction models from CATCh (Additional Information 4:11). Even removing a tool with a low individual performance had a major impact on the performance of the classifier (e.g. drop in sensitivity of CATCh reference when tested on MOCK2-a from 0.82 when including Pintail to 0.69 when excluding Pintail in the classifier). Similarly, when calculating the overlap of correctly predicted chimeric sequences between different individual algorithms, the added effect between the different individual tools is highlighted, e.g. 1,060 out of the 13,205 chimera in the MOCK2-a dataset could exclusively be predicted by Pintail (Additional Information 4:2).

4.4.2. Training and testing CATCh

As mentioned in the material methods, a wide range of mathematical functions (i.e. kernels) has been developed to address classification problems. Different kernel functions were compared by training and testing them on the training data (i.e. data containing the features extracted from the results (e.g. scores) of the different individual chimera detections tools together with the information whether the read is chimeric or non-chimeric. As mentioned in the material and methods section, the training data were split into two parts, a first part used for training classifiers using different kernel functions, and a second set to measure the performance of different kernels. This performance of the different trained classifiers on the test data was

assessed by applying a cut-off score on the output of the classifier which guarantees a pre-defined level of specificity (0.90 for the reference and 0.93 for *de novo*), and subsequently comparing the sensitivities for each of them. For both the reference based as well as the *de novo* implementation of CATCh, the Support Vector Machine with Pearson VII Universal Kernel (SVM-PUK) obtained the highest accuracy, with sensitivities of 0.85 and 0.92 and specificities of 0.90 and 0.93 respectively on the test data (Additional Information 4:9). Those results were noticeably better than the performance of each individual tool as well as the union of chimeric predictions of all individual tools (Additional Information 4:12).

To test the dependency of the CATCh classifier on the accuracy of the training data (i.e. whether a read is correctly identified as being chimeric or not in the training data), we artificially introduced a mislabelling of the chimeric reads in the training data. As such, randomly switching the label of a read from chimeric to non-chimeric or vice versa, was tolerated up to a level of 5% for the reference based classifier, and up to 10% for the *de novo* implementation. Exceeding these thresholds lead to a dramatic decrease in sensitivity. However, since we restricted the training data to high-quality predictions – all reads where the chimeric prediction was not clear, were removed from the data – the fraction of mislabelled reads will be very low.

To test whether it was possible to integrate both approaches in one powerful classifier, a classifier combining both *de novo* and reference based algorithms was built, and tested on MOCK2-a and MOCK2-b. However, compared with the separate *de novo* or reference based classifier, no improvement (MOCK2-a) or only a marginal improvement (MOCK2-b) in the sensitivity with the same cost in the specificity was observed (Additional Information 4:13), and a major increase in the running time. However this might be due to the fact that the mock communities used are not

challenging enough. Thus, most of the chimeras were already detected by either one of the approaches, allowing no room to demonstrate any improvement.

4.4.3. Performance of reference based algorithms

For benchmarking the different reference based chimera detection algorithms, all available tools were run on different datasets, each of them already used to optimize one of the existing chimera detection tools. When comparing our ensemble algorithm with the best performing alternative, the CATCh classifier obtains a higher sensitivity on *Simu1, Simu2* and *Simu3* data (respectively 14%, 12 and 11% higher), and an equally high sensitivity on *Simu4* data (95% for both UCHIME and CATCh), while showing a small drop in specificity (respectively 4%, 2%, 1% and 3%), see **TABLE 4:2**.

In order to simulate real-life situations, where sequence reads are divergent from sequences in the reference database due to factors like natural variation (i.e. predict chimera that are produced by 16S rRNA sequences which are not present in the currently used reference databases), or sequencing errors (sequences containing insertions, deletions or mismatches can rise up to 1%), we challenged the chimera detection algorithms with adapted versions of the *Simu2* and *Simu4* datasets where 1% and 5% indels and mutations were added (Additional Information 4:14). Where some of the individual tools like UCHIME could tolerate 1% of mutations, the decrease in sensitivity was much more pronounced at 5% level compared with CATCh. Indeed, the drop in sensitivity at 5% mutations varied between 10% and 20% for UCHIME (i.e. the best performing individual tool in this test) for the different type of mutations, while CATCh showed a drop in sensitivity between 4% and 9%). Overall, CATCh shows the highest sensitivity over *Simu2* as well as *Simu4* datasets, with the clearest difference when the average percentage of indels and mutations at 5%.

As an additional test, the chimeras in *Simu2* and *Simu4* were separated according to their level of divergence as reported in the original files (*Simu2* from 1 to 10% and *Simu4* data from 1 to 25%). As shown in **FIGURE 4:2**, CATCh appeared to endure those challenges with the best sensitivity reported compared with other tools. Indeed, it combines the advantages of ChimeraSlayer – having a high sensitivity at low divergence – and maintains a good performance at higher divergence as observed with UCHIME. This also confirms the finding that DECIPHER performs at its best on high divergence, however the sensitivity is still lower than UCHIME and ChimeraSlayer.

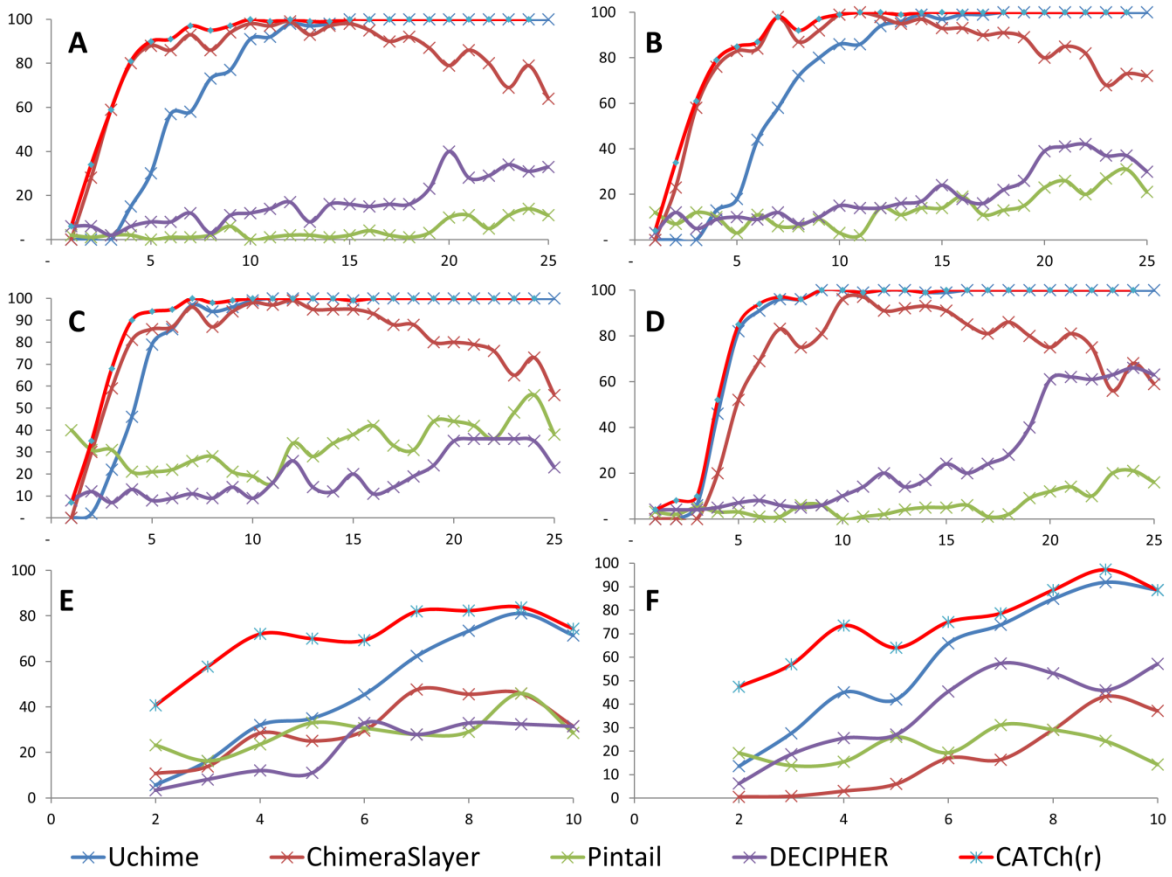


Figure 4:2. Illustration of the effect of chimeric sequences divergence on the sensitivity of various reference-based chimera detection tools. Several datasets were used with deletions (A), indels (B,E), insertions (C) and mismatch (D,F) plotting the divergence on the x-axis and the sensitivity on the y-axis. Both Simu4 (A,B,C,D) and Simu2 (E,F) datasets were used, having chimeras with a divergence ranging from 1 to 25% and from 1 to 10% in the Simu4 and Simu2 data respectively. Five tools where involved in this comparative study: UCHIME, ChimeraSlayer, Pintail, DECIPHER and CATCh.

In order to test the robustness of the different tools against the number of segments in chimeric sequences (di-, tri- and tetramera) in sequencing data, a comparison of the performances on *Simu2* and *Simu3* data (respectively 300 and 1000 chimeras for each chimera type) was performed. In general, CATCh appeared to have the highest sensitivity in all of the classes of chimeras in both datasets (Additional Information 4:3).

Similarly, the evolution of the performance of the reference based algorithms was tested in relation to the chimeric range (i.e. representing the shortest region originating from one of the parents) in a similar way as described by Wright et al. (2012). Plotting the sensitivity versus the chimeric range (Additional Information 4:4) emphasizes that the most challenging cases to detect are those where one of the parents participated with less than 200 nt, a condition where DECIPHER obtained very good results. This results is on its turn reflected in a high sensitivity of the CATCh classifier. CATCh achieved the highest average sensitivity over all ranges of 0.95 while DECIPHER, UCHIME, ChimeraSlayer and Pintail achieved 0.89, 0.88, 0.80 and 0.26 respectively.

4.4.4. Performance of de novo based algorithms

For the *de novo* algorithms, a similar comparative study was performed combining UCHIME *de novo*, ChimeraSlayer *de novo* and Perseus, testing them against CATCh *de novo* using the MOCK1 dataset, as described in the methods section. CATCh outperformed Perseus and UCHIME *de novo* mode, by achieving a sensitivity of 0.97 compared to 0.94 and 0.92 for the others tools respectively without any reduction in the specificity (**TABLE 4:2**). A second analysis took into account the different levels of divergence in the MOCK1 dataset ranging from 1 to 15%. Just as the three other tools, CATCh *de novo* mode showed sustainable performance, with the most clear increase in sensitivity at lower divergence levels (Additional Information 4:5).

Table 4:2. Illustration of the performance difference between CATCh (default score) and each individual tool after applying a cut-off which results in the same specificity as the corresponding CATCh output.The only exception is DECIPHER as it does not allows the usage of a cut-off score. The difference is the increase in sensitivity obtained when using the CATCh classifier compared to each individual tool.

	Reference					***de novo***				
MOCK2-a	**UCHIME**	**Chimera Slayer**	**Pintail**	**DECIPHER**	**CATCh**	**UCHIME**	**Chimera Slayer**	**Perseus**	**CATCh**
Sensitivity	0.77	0.77	0.03	0.61	0.82	0.78	0.78	0.80	0.85
Specificity	0.92	0.92	0.92	0.93	0.92	0.91	0.91	0.91	0.91
MCC	0.64	0.65	-0.12	0.49	0.70	0.64	0.64	0.66	0.72
Accuracy	0.82	0.83	0.30	0.72	0.86	0.83	0.82	0.83	0.87
Cut-off	0.44	72.5	31.5	0.50	0.62	0.21	74.0	0.40	0.70
Difference	5	5	79	19	-	7	7	5	-
MOCK2-b	**UCHIME**	**Chimera Slayer**	**Pintail**	**DECIPHER**	**CATCh**	**UCHIME**	**Chimera Slayer**	**Perseus**	**CATCh**
Sensitivity	0.90	0.89	0.02	0.70	0.93	0.90	0.90	0.93	0.95
Specificity	0.92	0.92	0.91*	0.93	0.92	0.91	0.91	0.91	0.91
MCC	0.79	0.77	-0.18	0.56	0.82	0.78	0.76	0.82	0.84
Accuracy	0.91	0.90	0.26	0.76	0.92	0.90	0.90	0.93	0.94
Cut-off	0.3	44	30	0.5	0.62	0.17	73.0	0.23	0.70
Difference	3	4	91	13	-	5	5	2	-

*Pintail could not reach specificity of 0.92 with any of the cut-off scores.

Similarly as for reference based algorithms, the robustness of the different tools against the number of chimeras (di-, tri- and tetramera) was compared for *de novo* chimera detection tools. The number of parents in the MOCK1 data were identified as described in detail in the original paper by Quince (2011): 2328 bimeras, 234 trimeras, and 3

tetrameras. As seen from these outputs a high sensitivity is obtained for all algorithms for detection of bimeras, and our classifier showed an increase in bimera detection (0.97 compared to 0.91, 0.70, 0.94 for UCHIME, ChimeraSlayer and Perseus, respectively).

4.4.5. Benchmarking on 454 Pyrosequencing data

To illustrate in a single experiment the differences in performance between algorithms (reference and *de novo* based), recently published mock sequencing data (MOCK2) were used (Schloss et al. 2011). The data were preprocessed using two different pipelines as described in the Materials and Methods, resulting in two different datasets (MOCK2-a and MOCK2-b), each of those preprocessing methods differing in both their computational cost and accuracy (Schloss et al. 2011). CATCh reference was found to have the highest sensitivity and maintaining a similar specificity as the best performing individual reference based chimera detection tool on both datasets. Moreover, when comparing CATCh *de novo* to the best performing individual method, a major increase in the sensitivity (11% and 8%) on the MOCK2-a and MOCK2-b datasets respectively was found, with a reduction at the specificity of 3% for both datasets (**TABLE 4:2**).

As an extra evidence for the increased performance of CATCh, the cut-off score of each individual chimera detection algorithm was tuned in such way that the same specificity as obtained with CATCh was reached. Subsequently the corresponding sensitivities for the different algorithms were compared at this fixed specificity level. Note that DECIPHER was excluded from this analysis as it does not produce a score which can be used as cut-off. Using MOCK2-a and MOCK2-b we found that CATCh reference has the highest sensitivity with an improvement of 4% (on average) over the best performing individual tool. The same behaviour was observed with CATCh *de novo* with an average of 3.5% improvement over the best *de novo* tool (**Table 4:3**). It is

important to stress that despite the differences in the ratio of chimeric and non-chimeric reads, the performance (in term of sensitivity and specificity) was unaffected.

To further illustrate the improvement brought by both CATCh classifiers, and to show that this improvement is independent of the threshold score applied to any of the other individual tools, a parameter sweep using different cut-off scores was applied for the different chimera detection tools (except for DECIPHER that does not provide a cut-off score). It is important to emphasise that in practice, changing the cut-off score is not advisable, as it has been adjusted to achieve the best balance between sensitivity and specificity. For each parameter setting the Matthews Correlation Coefficient (MCC) was calculated for all algorithms and both CATCh classifiers. This approach was applied on both MOCK-2 datasets (MOCK2-a and MOCK2-b) where in each dataset reads were randomly sampled in such a way that the number of chimeric sequences equals the non-chimeric sequences. This was not to bias the plot towards the sensitivity or the specificity (as the number of chimeric sequences more abundant than non-chimeric ones in MOCK2). This led to 10894 and 16604 reads for MOCK2-a and MOCK2-b respectively, each consisting of an equal ratio of chimeric and non-chimeric.

From the graphs it can be derived that CATCh has the highest MCC score over a wide range of different cut-off scores (**FIGURE 4:3**). For the *de novo* implementations, the graphs for CATCh *de novo* and Perseus are in some regions showing a similar trend. However, when tracing back the sensitivity and specificity values at these points, we noticed that these peaks are situated in the left region of the graph (i.e. corresponding to a low cut-off score), resulting as such in a rather exceptional low specificity. E.g., for MOCK2-a these high MCC values are produced by gaining a high sensitivity (e.g. 0.93 for CATCh and 0.91 for Perseus on MOCK2-a) however at the cost of a considerable reduction of the specificity (0.87 for both tools on MOCK2-a), denoting a

dramatic increase in the number of false positive chimera. Moreover, using the default cut-off score of each tool (indicated with a red cross in **FIGURE 4:3**), CATCh automatically returns those predictions that reflect the most optimal performance, while other tools might be performing sub-optimal with its default cut-off score (e.g. Perseus). This behaviour of CATCh is a major advantage when dealing with real-life data, as it is far from trivial to change the cut-off score to obtain an optimal performance when dealing with non-mock data. The drop in MCC values observed below 0.45 and above 0.8 is a consequence of having a major reduction in the specificity and the sensitivity valuesrespectively (classifying all sequences as chimeric or non-chimeric).

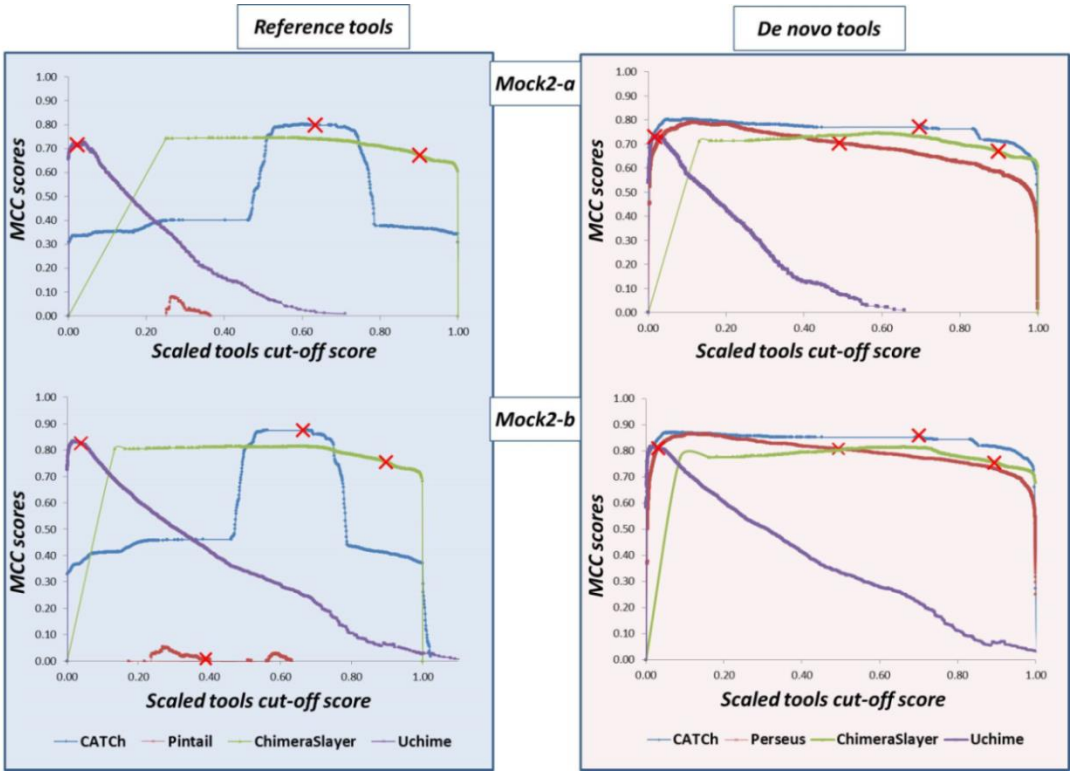


Figure 4:3. Plots illustrating the robustness of each tool (reference and *de novo*) to changing their (scaled) cut-off parameters via plotting the Matthews Correlation Coefficient (MCC) in Y-axis and cut-off parameters in the X-axis, using MOCK2-a and MOCK2-b. The scores are scaled to fit the same range between [0.00-1.00]. The plot shows that the performance of CATCh (reference and *de novo*) is the highest compared to results obtained for different cut-off scores for each individual tool.

Table 4:3. Sensitivity and specificity values of all tested chimera prediction tools. The first three datasets are tested with both reference and *de novo* based tools. The second part of the table contains the datasets either analysed using reference or *de novo* tools. The last three rows give respectively the average sensitivity and specificity, and the average increase or decrease in sensitivity / specificity when compared with CATCh.

		Reference tools						***De novo* tools**			
Evaluation	**Datasets**	**UCHIME**	**ChimeraSlayer**	**Pintail**	**DECIPHER**	**CATCh**	**Datasets**	**UCHIME**	**ChimeraSlayer**	**Perseus**	**CATCh**
Sensitivity	MOCK2-A	0.79	0.66	0.58	0.61	0.82	MOCK2-A	0.74	0.66	0.76	0.85
Specificity		0.94	0.95	0.42	0.93	0.92		0.94	0.94	0.92	0.91
Sensitivity	MOCK2-B	0.90	0.77	0.58	0.70	0.93	MOCK2-B	0.87	0.80	0.87	0.95
Specificity		0.94	0.96	0.46	0.93	0.92		0.94	0.94	0.93	0.91
Sensitivity	MOCK3	0.75	0.63	0.06	0.61	0.81	MOCK3	0.64	0.65	0.73	0.81
Specificity		0.95	0.99	1.00	0.98	0.94		0.98	0.97	0.98	0.96
Sensitivity	*Simu1*	0.56	0.51	0.15	0.45	0.70	MOCK1	0.92	0.71	0.94	0.97
Specificity		1.00	1.00	0.88	1.00	0.96		1.00	1.00	1.00	1.00
Sensitivity	*Simu2*	0.69	0.53	0.21	0.48	0.81	MOCK4	0.37	0.27	0.27	0.47
Specificity		1.00	0.99	0.80	0.99	0.98		0.96	0.98	0.94	0.91
Sensitivity	*Simu3*	0.83	0.66	0.27	0.87	0.94	MOCK5	0.08	0.11	0.14	0.16
Specificity		0.99	1.00	0.86	0.98	0.97		1.00	0.95	0.99	0.98
Sensitivity	*Simu4*	0.95	0.90	0.17	0.27	0.95					
Specificity		1.00	0.99	0.86	1.00	0.97					
Sensitivity	**Average**	**0.78**	**0.67**	**0.29**	**0.57**	**0.85**	**Average**	**0.60**	**0.53**	**0.62**	**0.70**
Specificity		**0.97**	**0.98**	**0.75**	**0.97**	**0.96**		**0.97**	**0.96**	**0.96**	**0.95**
Difference		**7 / -2**	**18 / -2**	**56 / 21**	**28 / -1**	**-**		**10 / -2**	**17 / -1**	**8 / -1**	**-**

4.4.6. Benchmarking on Illumina MiSeq data

Next to 454 Pyrosequencing data, the performance of chimera detection algorithms was also tested on Illumina MiSeq data using the MOCK3 community. In general the same trend could be observed regarding the performance at the level of sensitivity and specificity. For the reference based implementations, an increase in sensitivity of 6% was observed compared with the best performing individual tool (UCHIME), with a drop in the specificity of 1%. Similarly, for *de novo* implementations, the 2% drop in specificity of CATCh compared to Perseus, was largely compensated by an increase in sensitivity of 8% (**TABLE 4:2**).

4.4.7. Benchmarking on 18S rRNA amplicon sequencing data

The main results described in this manuscript are obtained for 16S rRNA amplicon sequencing data. However, the *de novo* implementations have been suggested to work also for other marker genes (Edgar et al. 2011). Comparing all *de novo* implementations on MOCK4 and MOCK5 –containing 18S rRNA amplicon sequencing data – resulted in an increase in sensitivity combined with a drop in specificity (**TABLE 4:2** and Additional Information 4:15). More importantly, the sensitivity values were in general much lower than the performance obtained on 16S rRNA gene sequencing data. The best performing algorithm (UCHIME) only obtained sensitivity of 37% and 8% on MOCK4 and MOCK5, while CATCh obtained sensitivity of 47% and 16% respectively.

4.4.8. Impact of CATCh on OTU clustering

Apart from showing the improved accuracy of CATCh on the sensitivity and specificity statistics, the effect of our algorithm on the OTU clustering was assessed. Indeed, even when working with mock communities, the number of OTUs in most cases exceeds the theoretical number of species in the mock communities. This inflation of the number of

OTUs can partially be explained by the presence of undetected chimeric sequences. As such, chimera detection tools can reduce the number of OTUs by removing those chimera before performing OTU clustering. When assessing this feature on the MOCK1 community, CATCh clearly had the most beneficial effect, as it reduced the number of OTUs with 23% and 35% compared to the best available tool for reference (UCHIME) and *de novo* (Perseus) applications respectively. This effect can also been visualized by plotting the rarefaction curves after applying all tested chimera detection tools (see Additional Information 4:6). The same trend was observed for MOCK2-a and MOCK2-b (samples rarefied to 10,000 reads, see Additional Information 4:7), leading to a drop of 8% and 11% respectively in the number of OTUs using the reference-based CATCh implementation. Running *de novo* CATCh on MOCK2-a and MOCK2-b had an even more dramatic effect by reducing the number of OTUs with 22% and 18% respectively, compared with the second best tool. It should be noted within this context that the diversity of the MOCK2 community is much lower than MOCK1 (20 species versus 91 species), which might explain the smaller improvement observed for MOCK2. However, in general, by using CATCh the number of OTUs returned is closer to the expected number of species in the mock communities. This reduction in number of OTUs is also confirmed when running CATCh on real biological data, like the eleven samples used in the SOP as proposed in Schloss et al. (2011), downloaded from http://www.mothur.org/w/images/a/a1/SOPData.zip. Indeed, integrating CATCh (reference-based or *de novo*) into the processing pipeline led to a reduction of 23% and 30% of the OTUs compared with UCHIME reference or Perseus respectively.

4.4.9. Computational resources

The computational cost of the different chimera detection algorithms was assessed by examining the running time for all chimera detection algorithms on three different

datasets, each of them produced by a separate sequencing technology i.e. the MOCK2-a dataset obtained via 454 Pyrosequencing (Schloss et al. 2011), MOCK3 via Illumina MiSeq sequencing technology (Kozich et al. 2013) and an additional dataset via PacBio SMRT sequencing (Marshall et al. 2012).

Table 4:4. Execution times of different reference based and *de novo* chimera detection tools, tested on 16S rRNA amplicon sequencing data obtained from three different sequencing platforms. For CATCh the execution time of the classifier is mentioned (CATCh classifier), as well as the total runtime required for the CATCh pipeline, being the sum of the execution times of different individual tools and the classifier.

Dataset Name		MOCK2-a	MOCK3	PacBio
Sequencing platform	Sequencing platform	454 Titanium	MiSeq	PacBio
Total Dataset Statistics	Number of Reads	1433524	2676388	136327
	Number of Unique Reads	18652	25570	100154
	Number of Samples/Replicates	90	8	16
Average Per Sample/Replicate	Length	197	259	423
	Number of Reads	15928	334549	8520
	Number of Unique Reads	207	3196	6260
Reference	Uchime	00:02:10	00:34:39	02:07:05
	ChimeraSlayer	00:00:41	00:11:35	00:32:15
	Pintail	00:05:50	01:15:20	03:40:56
	DECIPHER	00:20:39	00:35:30	01:04:10
	CATCh (classifier)	00:00:02	00:00:23	00:00:49
	CATCh (total)	00:29:22	02:37:27	07:25:15
De novo	Uchime	00:00:01	00:01:25	00:21:11
	ChimeraSlayer	00:00:40	00:11:39	00:28:19
	Perseus	00:00:10	00:34:19	03:12:48
	CATCh (classifier)	00:00:01	00:00:20	00:00:38
	CATCh (total)	00:00:52	00:47:43	04:02:56

The execution time – consisting of CPU time and input / output operations – is differing significantly between the different tools, and is also largely depending on the number of unique reads and average read length produced per sample (**TABLE 4:4**). In general *de novo* tools outperform reference based tools at the level of execution time. However, as the execution time for *de novo* tools increases exponentially with the number of input sequences while for reference based algorithms this is only linear, the difference between both approaches is diminishing with an increasing number of

unique reads. For example, UCHIME *de novo* is performing 130 times faster on MOCK2-a than UCHIME reference, while this is reduced to 6 times faster with the PacBio dataset. For ChimeraSlayer, execution times are almost the same for reference and *de novo* implementation on the PacBio dataset.

Looking at reference based tools, UCHIME reference based is performing very fast on small dataset (i.e. low number of unique reads) in comparison with DECPIHER. The longer execution time for DECIPHER for small datasets is largely due to reading in the large reference database supporting the algorithm. However, once this database is read in, DECIPHER can quite easily withstand an increase in sequencing data, only leading to a relatively small increase in execution time when running on large datasets, clearly outperforming UCHIME.

For *de novo* based algorithms, UCHIME is outperforming Perseus, and to a lesser extent also ChimeraSlayer. However, important to notice within this context is the exceptional performance of Perseus on larger datasets like MOCK3 – being the best performing individual chimera detection tool (10% higher sensitivity than UCHIME), largely making up for the longer execution time.

The added computation burden of the CATCh classifier is rather limited, not exceeding one minute even for the largest dataset. However, as all individual tools need to be run separately before CATCh can be applied, the execution time of our proposed pipeline is dependent to the slowest performing algorithm (Pintail in reference mode and Perseus in *de novo* mode). For datasets containing a low number of unique reads (like MOCK2-a), the total execution time of CATCh reference and *de novo* is significantly higher than the best performing individual tool (UCHIME reference and Perseus respectively) i.e. 29 minutes versus 2 minutes for the reference based implementation and 52 seconds versus 10 seconds for the *de novo* implementation. However, on large datasets this is

effect is less pronounced, where the reference mode of CATCh takes around 3.5 times more time than UCHIME. For *de novo* mode, this effect is even more modest, extending the execution time with 25% compared to the best individual tool Perseus.

4.5. Discussion

The first goal of this work was to perform a comparative study between different chimera detection algorithms. For each of the implementations, an obvious bias was observed towards the datasets that were used to fine-tune their respective algorithms on, as almost each algorithm outperformed the other tools on its own subset of data. Nevertheless, important trends in chimera detection tools could be derived: UCHIME showed robustness against the presence of mutations, ChimeraSlayer clearly outperformed other tools when dealing with sequences containing indels at low divergence level, and DECIPHER performed very efficient on chimera with a short chimeric range.

In order to combine the advantages of all of these tools, we used a machine learning approach to develop a classifier that combines the output of all of these chimera prediction tools in an intelligent way. Our reference based classifier obtained in almost all cases a better sensitivity. The same trend has been observed with *de novo* chimera detection algorithms. Where the improvement in sensitivity of our classifier was less pronounced on the MOCK1 dataset, a significant increase in the sensitivity was observed for the MOCK2-a and MOCK2-b 454 Pyrosequencing reads datasets as well as the MOCK3 Illumina MiSeq dataset. On the other hand, CATCh frequently reports a higher number of false positives, resulting in a lower specificity. However, when forcing all algorithms to produce the same specificity by tuning the cut-off scores, both CATCh implementations (reference and *de novo*) showed a significant increase in sensitivity when compared with the best scoring individual algorithm. In summary,

when comparing the results of CATCh with the second best performing tool (UCHIME for reference and Perseus for *de novo* tools) for all possible mock datasets tested in this manuscript, CATCh detects on average 7% and 8% more chimeras for the reference and *de novo* implementations respectively. This goes at the cost of only 1% at the specificity level in both cases. This beneficial effect on the detection chimera is translated into a more accurate results after OTU clustering (up to 35% less OTUs predicted), since data processed by CATCh are for MOCK1 and MOCK2 consistently closer to the theoretical number of OTUs expected in the community (i.e. the number of species in the mock community). Thus, whether one is interested in beta diversity analysis, where specificity is crucially important, or in alpha diversity necessitating high sensitivity, CATCh the best available tool. Caution should be taken when extrapolating those conclusions to real biological samples since they will have a greater diversity than most mock communities. However, mock communities are the only way to reliably assess the difference when benchmarking various algorithms.

Where in the initial publication, (Schloss et al. 2011), the database-dependent algorithms slightly outperformed the *de novo* tools (sensitivity of 79% versus 74%), the opposite was observed when comparing our reference based classifier with the *de novo* classifier (84% versus 94%). Regardless the fact that in their study the reference based algorithms outperformed the *de novo* ones, it was already suggested using the *de novo* (database-independent) approach over the reference based ones, firstly due to the independence on incomplete reference databases, and secondly because of a faster execution time (Schloss et al. 2011). Furthermore, we could show now that the *de novo* approach developed in this study is also at the performance level the preferred approach over reference based methodologies. Indeed, even in an area where it would be expected that reference based algorithms would be favoured – since we are dealing

with mock communities containing well studied bacterial isolates known to be present in the reference databases – the *de novo* classifier has a better performance than reference based approached. This conclusion corresponds with the observation in Edgar et al. (2011) calculating a higher performance of UCHIME *de novo* on a mock dataset than UCHIME reference.

Where the *de novo* approach also results in faster execution times when dealing with small datasets, this effect is less clear with datasets containing a large number of unique reads (> 3000 sequences i.e. samples with a high sequencing coverage or highly diverse samples). The best performing individual algorithms – UCHIME and Perseus for reference based and *de novo* respectively – require comparable execution times in such a large-scale setup. When extending this to CATCh, the additional computational burden is larger in reference based mode, resulting in a significant increase in execution times for small and large sequencing datasets respectively. However, for *de novo* mode this effect is much more modest, only extending the runtime 25% for large sequencing datasets.

Important to notice is that this significant increase in performance of our *de novo* classifier is obtained by combining only three *de novo* chimera detection tools (UCHIME *de novo*, ChimeraSlayer *de novo* and Perseus). Our classifier would benefit from integrating more *de novo* algorithms once being available. This also highlights the power of such ensemble algorithms. When new algorithms will become available in the future they can easily be integrated in the classifier most probably leading to even better chimera prediction results than the ones presented here.

Additionally, when combining the output of different algorithms, a machine learning approach is preferred over a rudimentary approach such as taking the union or intersection of the predictions of different algorithms. Indeed, taking the union of

ChimeraSlayer, UCHIME and DECIPHER would result in sensitivity values comparable to results obtained with CATCh, however with a dramatic decrease of the specificity (88%, 78%, 83% and 84% for the *Simu1, Simu2, Simu3* and *Simu4* data respectively versus 96%, 98%, 97% and 97% for CATCh). These data agree with the observation that the number of false positive chimera predictions is minimal between DECIPHER and UCHIME, leading to a decrease in specificity when simply combining both outputs in a straightforward way (Wright et al. 2012). For the *de novo* implementation, the opposite trend is observed, as the sensitivity of our classifier is higher than the union while maintaining a similar specificity, suggesting an overlap of Perseus and UCHIME *de novo* in the false positive predictions, while having an added effect in the detection of true chimeras (data not shown).

The majority of the conclusions discussed in this work are derived based on 16S rRNA gene sequencing data. Certainly for the *de novo* implementations, it is tempting to assume that the area of application could easily be extended to other biodiversity marker genes like 18S rRNA, 23S rRNA or ITS regions. However a small-scale test case using two different 18S rRNA mock communities suggests that additional analyses might lead to unexpected results (probably due to the hard coded parameters adjusted within each tool for 16S rRNA gene application) and result in a re-thinking of the chimera detection methods currently used in eukaryotic biodiversity assessments. Where the basic concept of *de novo* tools should also be applicable for assessing chimeric sequencing in eukaryotic biodiversity studies, precaution is needed when shifting away from the 16S rRNA marker gene to eukaryotic marker genes, as a straightforward application of *de novo* chimera detection tools on 18S rRNA results in significantly lower sensitivity values.

4.6. Conclusion

In conclusion, a comparison between different chimera prediction tools was performed, pointing out each tool's strengths and weaknesses. Based on this information an ensemble classifier was developed for reference based as well as *de novo* chimera detection tools, which is able to produce stable results over various mock datasets. Moreover, since the classifier combines the strengths of various individual chimera detection tools, it shows an increased robustness against different confounding factors like a low parent divergence, short length of the chimeric range and a varying number of parents. The beneficial effect of CATCh is highlighted by improved OTU clustering results in mock datasets, returning OTU numbers closer to the true number of species in the mock community.

Chapter 5

Reconciliation Between Operational Taxonomic Units and Species Boundaries

Mohamed Mysara[1,2,3,4], Peter Vandamme[5], Ruben Props[6], Frederiek-Maarten Kerckhof[6], Natalie Leys[1], Nico Boon[6], Jeroen Raes2,[3,4] and Pieter Monsieurs[1]

[1] Unit of Microbiology, Belgian Nuclear Research Centre (SCK-CEN), Mol, Belgium.
[2] Department of Bio-engineering Sciences, Vrije Universiteit Brussel, Brussels, Belgium.
[3] VIB Center for the Biology of Disease, VIB, Leuven, Belgium.
[4] Department of Microbiology and Immunology, REGA institute, KU Leuven, Belgium.
[5] Department of Biochemistry and microbiology, Ghent University, Belgium.
[6] Department of Biochemical and microbial technology, Ghent University, Belgium.

This chapter is based on the following submitted publication:

Mysara M, Vandamme P, Leys N, Raes J, Monsieurs P. (2016). Reconciliation between Operational Taxonomic Units and Species Boundaries. FEMS Microbiology Ecology (submitted).

5.1. Abstract

The development of high-throughput sequencing technologies has revolutionized the field of microbial ecology via 16S RNA gene amplicon sequencing approaches. Clustering those amplicon sequencing reads into Operational Taxonomic Units (OTUs) is the most commonly used approach to estimate microbial diversity. However, this is done using a fixed cut-off on the sequence identity, mostly 97%, and this independent of the amplified region of the 16S rRNA gene. This 97% threshold was chosen with the intended purpose that resulting OTUs could be interpreted as a proxy for a bacterial species. Our results show that the robustness of such a generalized cut-off is questionable when applied to short amplicons that are only covering one or two variable regions of the 16S rRNA gene. It will lead to biases in diversity metrics and makes it hard to compare results obtained with different amplicons. The method developed within this work takes into account the differential evolutional rates of taxonomic lineages in order to define a dynamic and taxonomic dependent OTU clustering cut-off score. For a taxonomic family consisting of species that show high evolutionary conservation in the amplified variable regions, the cut-off will be more stringent than 97%. By taking into consideration the amplified variable regions as well as the taxonomic family when defining this cut-off, such a threshold will lead to more robust results. Additionally, as the cut-off is defined taking into account the bacterial taxonomy, this methodology results in a more accurate correspondence between OTUs and species.

5.2. Introduction

A major breakthrough in microbial ecology has been realized by the usage of PCR based amplification of phylogenetic marker genes, such as the 16S or 18S rRNA gene for the assessment of microbial diversity in a specific environment, thereby omitting the time-consuming and challenging culturing approach. The usage of this culture-independent approach has been accelerated via the introduction of high-throughput sequencing technologies, leading to a dramatic increase of marker gene sequencing studies for the assessment of microbial communities.

However, downstream processing of those large amounts of sequencing data is not forthright. In the most straightforward approach, the reads resulting from the high-throughput sequencing platforms are grouped and classified based on their sequence identity to a reference taxonomic dataset, a methodology most often referred to as phylotyping. Yet, such a binning procedure will be biased towards the existing taxonomic classification – including its classification errors – which is largely based on cultivable organisms (Amann et al. 1995; Rosselló-Móra 2012; Wang et al. 2007). Therefore, the most widely used bioinformatics pipelines like mothur (Schloss et al. 2009), QIIME (Caporaso et al. 2010) and UPARSE (Edgar 2013) implemented an alternative approach, where sequencing reads are grouped together based on their sequence identity to each other rather than a reference taxonomic dataset, resulting in cluster of reads, referred to as Operational Taxonomic Units (OTUs).

For clustering those reads into OTUs, an identity threshold of 97% is frequently used for pragmatic reasons, as it offers a compromise between the potential inflation of the number of OTUs due to sequencing errors and the cut-off used for taxonomic classification.

Indeed, on the one side, sequencing errors might introduce artificial OTUs, thereby leading to an overestimation of the microbial diversity (Huse et al. 2007; Quince et al. 2009; A Murat Eren et al. 2013; Edgar 2013). On the other side, it has already been suggested that this cut-off of 97% underestimates the total microbial diversity (Koeppel & Wu 2013; A. Murat Eren et al. 2013; Pedrós-Alió 2006; Chen et al. 2013), as bacterial taxonomists nowadays use a more stringent cut-off of 98.65% to delineate species (Kim et al. 2014; Janda & Abbott 2007; Stackebrandt & Ebers 2006). It should however be stressed that organisms that share more than 98.65% of their full 16S rRNA gene sequences may or may not represent the same species (Fox et al. 1992), so the potential for underestimating bacterial diversity through 16S rRNA amplicon sequencing is intrinsic to the molecule studied and cannot be avoided.

Moreover, the currently most widely used sequencing platform for microbial community assessment – being the Illumina MiSeq platform – is only able to sequence a small sub-region (nowadays mostly covering the V3-V4 regions) of the 16S rRNA gene thereby ignoring the fact that the sequence identity threshold scores are derived for full-length sequences. Given the fact that the degree of conservation for each variable region might vary between taxonomic lineages, the region(s) chosen for amplification can have a large impact on the microbial diversity observed (Schmalenberger et al. 2001; Yu & Morrison 2004; Clarridge 2004). Moreover, applying a general cut-off for OTU clustering is based on the naïve assumption that 16S rRNA genes evolve at the same rate regardless of their taxonomic affiliation (Schmidt et al. 2014; Schloss & Westcott 2011; Yarza et al. 2014). This might result in merging two distinct taxonomic species into one OTU when the conservation is higher than expected, or a splitting up of one species over different OTUs when more sequence variation is observed. Since the recommended amplicon targeted for 16S

rRNA gene sequencing has been varying due to frequent updates of the sequencing technologies and corresponding selection of primer pairs, this will lead to biases in the assessment of microbial diversity, hampering direct comparison of diversity indices between studies targeting different regions of the 16S rRNA gene (Youssef et al. 2009; Schloss & Westcott 2011).

Several efforts have been made to address the limits of one or a limited number of variable regions by quantifying the loss of information, as well as the effect of using different (combinations of) variable regions on richness and phylogeny (Youssef et al. 2009; Chakravorty et al. 2007; Liu et al. 2008; Jeraldo et al. 2011; Kumar et al. 2011; Soergel et al. 2012; Ghyselinck et al. 2013; Yarza et al. 2014).

As such, performing the above mentioned OTU clustering on sequencing data obtained for amplicons covering one or two variable regions within the 16S rRNA gene will lead to important shortcomings concerning the comparability and interpretation of bacterial diversities. Therefore, within this work, we first show that applying a 97% cut-off during the clustering step results in a biased interpretation of the OTU clustering results, as the variability observed for a specific variable region within a taxonomic family might vary dramatically within the bacterial kingdom. Next, we propose a new approach for OTU clustering using a dynamic threshold score, instead of a fixed one, adapted for each bacterial family. Such an approach would compensate for the difference in evolutionary rates within the different taxonomic lineages throughout their various variable regions, as such resulting in OTUs reflecting more accurately the species level. Finally, this family-dependent threshold approach is validated using a 16S rRNA sequencing dataset obtained from synthetic and real microbial communities.

5.3. Materials and methods

Analyses performed within this work rely on type strains as stored in the high-quality and curated SILVA database – release 119 (part of the Living Tree Project (LTP) (Yarza et al. 2008). All species were grouped based on their taxonomic family, omitting those families containing overall less than three species or those with pending classification (see **TABLE 5:1**).

Table 5:1. Illustration the number of Bacteria and Archaea within the LTP database release 119: raw, cleaned data (omitting sequences with unclear classification), and filtered data (omitting sequences belonging to families consist of one or two sequences).

	Raw		**Cleaned**		**Filtered**	
Domain	**Bacteria**	**Archaea**	**Bacteria**	**Archaea**	**Bacteria**	**Archaea**
Phyllum	33	2	30	2	22	2
Class	67	9	57	9	39	8
Order	162	18	119	16	89	12
Family	327	31	280	28	216	17
Genus	2111	118	1937	115	1884	103
Total	10850	419	10511	411	10462	395

To study the level of variability within each taxonomic family, the selected 16S rRNA sequences were aligned using the SSU-ALIGN software package (Nawrocki 2009) which uses the Infernal software package (Nawrocki & Eddy 2013) for generating alignments, thereby taking into account the secondary structure of the 16S rRNA gene. This software allows determining the degree of evolutionary conservation for each position within the 16S rRNA per family, which can be quantified by assigning a conservation bit-score (ranging between zero to two representing complete dissimilarity or variability respectively) .

Nine hypervariable regions together with the intervening conserved regions were defined for all 16S rRNA sequences using the E. coli 16S rRNA gene as reference (Ashelford et al. 2005). Throughout the analysis described in the Results section, two

different approaches were used: 1) different amplicons of the 16S rRNA gene were used, covering the spam of one, till nine hypervariable regions (and adjacent conserved regions) and 2) a 50 base pair sliding window approach selecting sequences of 300 or 500 nucleotides. The latter approach tries to mimic the effect of working with short sequencing reads, currently producing amplicon sequences roughly varying between 300 and 500 base pairs.

Either starting from complete 16S rRNA gene sequences directly obtained from the reference database or starting from partial sequences of these genes (as mentioned in the paragraph above), OTUs were obtained for both approaches using the mothur commands *align.seq, dist.seqs* and cluster for alignment, distance calculation and clustering respectively. For distance calculation, the default one-gap option in mothur was selected, implementing the method proposed by Sogin et al. (2006), while for clustering the average neighborhood approach was selected (Schloss & Westcott 2011).

For the analysis of the real-life dataset, Illumina MiSeq sequencing data were obtained from the published mock samples of 56 species in 50 samples (Schirmer et al. 2015) with amplicons spanning the V4 or V3-V4 regions. The data were pre-processed using mothur, including the command *make.contigs, align.seqs, screen.seqs, filter.seqs, dist.seqs* and cluster. Chimeric sequences were identified via the mothur command *seq.error*, before performing the distance calculation. Due to dramatic differences in sequencing depth (i.e., varying between 3,000 and 3,500,000 reads), all samples were subsampled to 25,000 reads, thereby removing fourteen samples that do not reach this number of reads, as well as two samples that targeted the complete 16S rRNA gene rather than (a subset of) variable regions. The resulting OTUs were filtered by removing rare-OTUs (those representing < 0.1% of the number of subsampled reads).

Finally, to illustrate the real-life performance of our newly proposed methodology we used a dataset generated in the context of the Human Microbiome Project (HMP), consisting of 16S rRNA gene amplicon sequencing data obtained for 15 to 18 body sites from 300 healthy human individuals. The mothur-treated version of this dataset is available via http://www.hmpdacc.org/HMMCP/healthy/. Those samples for which sequencing data were obtained for all three different amplicons (V1-V3, V3-V5 and V6-V9) were selected. Filtering out those samples with less than 1000 reads for any of those amplicons led to a total of 64 samples. The number of sequences for each sample was normalized throughout the three amplicons, via randomly subsampling them to the smallest one.

5.4. Results

5.4.1. Divergent evolution of 16S rRNA variable regions

In the most ideal scenario, each OTU would represent a taxonomic species. However, as stated above, the current OTU clustering approaches using a fixed 97% threshold will not be able to achieve this goal. To further assess to which extent the current approach is deviating from the optimal scenario, all sequences selected from the SILVA database were aligned using the SSU-align algorithm. Subsequently, for each taxonomic family separately, the conservation of each position within the 16S rRNA gene is translated into a bit score, which was plotted via a heatmap (**FIGURE 5:1** and Additional Information 5:1). From this heatmap, it is clear that there is a distinction in evolutionary conservation of the variable regions between different taxonomic families. Additional examples are shown in Additional Information 5:2.

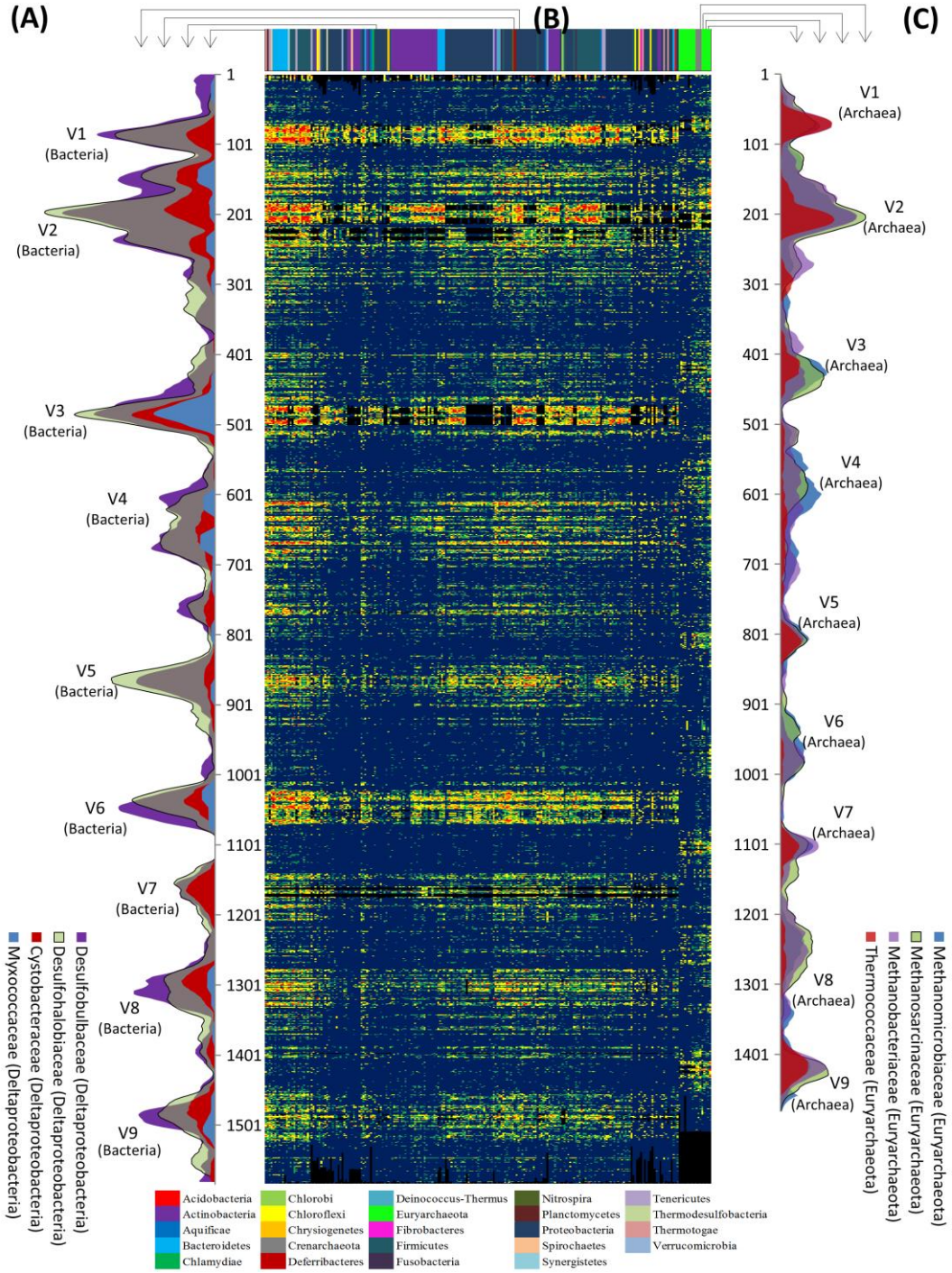


Figure 5:1. Different levels of conservation within the 16S rRNA gene compared between all bacterial and archaeal families. Within the heatmap each column represents a taxonomic family, and each row represents the position within the 16S rRNA gene, starting from the 5' start site (top) until the 3' end of the gene (bottom). The colour code of the cells reflects the alignment bit-score (as derived using the SSU-ALIGN algorithm), with red indicating the most variable and blue the most conserved positions as shown in section (B). To emphasize the variation within taxonomically related groups, four families belonging to the same bacterial class (Deltaproteobacteria) and four belonging to the same archaeal class (Euryarchaeota) were selected and the normalized bit-score (bit-score averaged over a 30-nucleotide window) was plotted against the position within the 16S rRNA gene , as shown section A and C.

Moreover, as the families do not contain the same number of species, we assessed whether the same observation could be made when comparing families containing a comparable number of species. Families were divided into three groups: families containing a low number of species (3-10 species per family), an intermediate number (11-40 species) or high number (41 or more species). Upon investigating the plots, it could be derived that also families with the same size differ in the level of conservation within the conserved and variable regions (data not shown).

5.4.2. Robustness of OTU clustering when combining different 16S rRNA hypervariable and conserved regions

To examine the discriminative power of the various variable regions, we combined one up to all nine variable regions into hypothetical amplicons of different lengths and subsequently clustered those into OTUs (see **FIGURE 5:2**). The aim of this analysis is to assess the potential erroneous merging of different species within the same OTU. In the most ideal scenario, each species would within this analysis be represented by one OTU, implying that the total number of OTUs would equal the number of tested species.

However, when applying the commonly used 97% identity as cut-off for the OTU clustering step, depending on the combination of hypervariable regions tested, all species are clustered into a total number of OTUs between 2586 and 7948 (i.e. 41% and 77% of the expected number of OTUs, respectively). Even increasing the OTU clustering cut-off value to 98% and 99% still uncovered a significant portion of erroneously merged species, i.e. the analysis returned between 47% and 82% of the expected number of OTUs for 98%, and between 56% and 89% of the expected numbers of OTUs for the 99% cut-off, respectively (**FIGURE 5:2**).

Importantly, even when using the complete 16S rRNA gene, an erroneous merging of distinct species into one OTU was observed for 35% of the species upon using 97% threshold. Also for the higher cut-off score of 98%, 98.65% and 99%, discrepancies could be found between the number of OTUs and the number of species, leading to over-merging up to 23%, 15% and 12% of the species respectively.

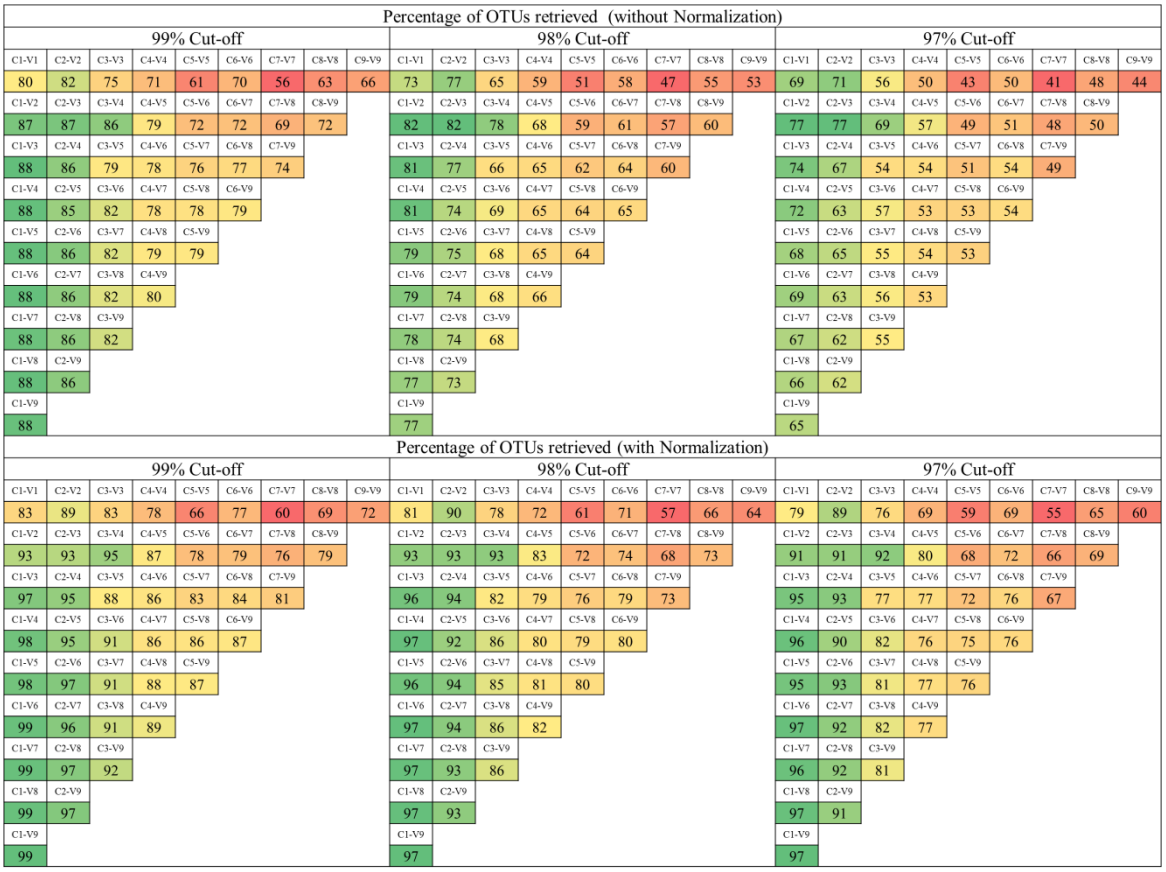


Figure 5:2. Heatmap illustrating the percentage of retrieved OTUs upon applying the various cut-offs, thereby using various combinations of variable regions, ranging from one region (e.g. covering the spam of V1) to nine variable regions (C1V9, i.e. the full length 16S rRNA gene). The analysis was repeated twice, the first time (upper part) performed on the total number of species within each family, while the second time (lower part) performed after cleaning up the database by only retaining one representative species in case two or more had a similarity higher than the applied cuff-off. The heatmap colour corresponds to the percentage of the OTUs retrieved, where a red colour is set in such a way that it represents the maximum number of over-merged OTUs and a green colour has been assigned to the minimum number of over-merged OTUs, per region within each cut-off.

In order to compensate for this bias, the database was cleaned up by only retaining one representative species in case a sequence identity higher than the applied cuff-off, as such resulting in a curated database complying with an identity cut-off score of 97%, 98% or 99% respectively. We repeated the analysis mentioned above on this curated database for different cut-off thresholds and different combinations of variable regions. However, even when working with this curated database, our analysis still resulted in significant erroneous merging reaching up to 40%, 43% and 45% respectively for 99%, 98% and 97% cut-offs.

5.4.3. Robustness of OTU clustering using short read lengths

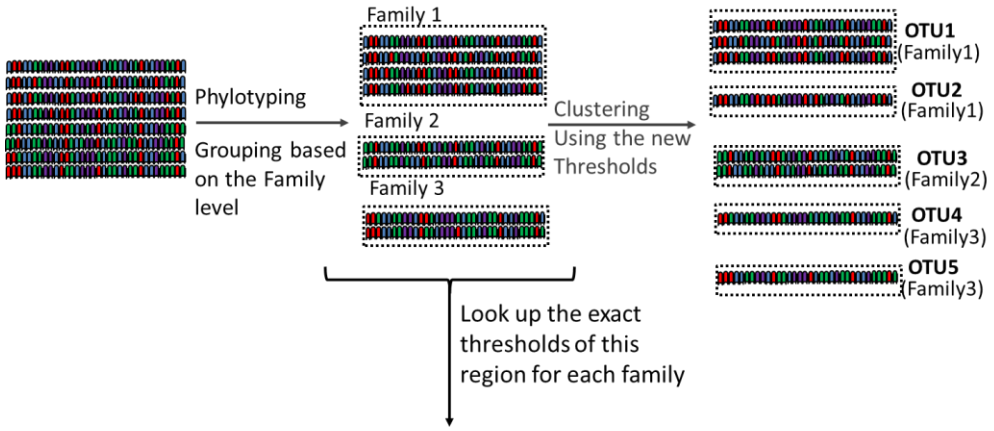
Where the previous analysis allows to gain insight into the discriminative power of combining different 16S rRNA gene regions, the next analysis deals with a real-life situation where a sequencing technology permits to sequence amplicons of a specific length (300 and 500 nucleotides in the hypothetical examples described below). The question to be answered is which part of the 16S rRNA gene would result in the most accurate OTU clustering i.e. OTUs reflecting most accurately the bacterial taxonomic classification. Using a 50 base pair sliding window, we ended up with 24 artificial amplicons with a length of 300 nucleotides and 20 amplicons with a length of 500 nucleotides. For each of those artificial amplicons, all pairwise distances between the species within one family were calculated.

For each of the windows, we sorted per family the distances obtained by pairwise aligning all sequences, allowing us to identify the minimum, the lower 2.5%, the lower 10%, the lower 25%, the median, the higher 25%, and the maximum distances observed between each two sequences originating from the same family. These different distance distributions demonstrate the differences between all species within a family, in a

heatmap plot (see Additional Information 5:3). As for the window covering the nowadays commonly amplified V3-V4 region on the Illumina MiSeq platform, a loss of 45% of the species on average was observed.

5.4.4. Dynamic family-dependent cut-off for OTU clustering

Based on the observations outlined above (Additional Information 5:3), it is clear that defining one generally applicable cut-off for OTU clustering is very difficult, as the selected amplicon as well as the taxonomic family are important parameters influencing an appropriate cut-off. Therefore we developed a new approach which dynamically defines the cut-offs applied when performing the OTU clustering step. Basically, instead of using a rigid cut-off of e.g. 97%, we propose a dynamic way for defining these cut-off scores, which will vary depending on the taxonomic family as well as the targeted amplicon. For a specified amplicon, this cut-off score within one specific family is calculated as follows: 1) for all species within this family – based on the LTP taxonomic database – the sequence fragment corresponding to this amplicon is derived 2) all those fragments are aligned and the distance scores are ordered 3) as we considered 2.5% of these distances to be outliers, the distance corresponding to the 2.5 percentile is used as OTU clustering cut-off. 4) If this value is lower than 1% or higher than 3%, this cut-off value is set at 1% or 3% respectively. Repeating those four steps for all taxonomic families results in a lookup table defining the OTU clustering cut-off per family. In practice, this implies that first all reads are assigned to a specific taxonomic family, and next per family an OTU clustering is performed using the cut-off derived from the lookup table (see **FIGURE 5:3**). For reads with uncertain taxonomic classification at the family level (e.g., due to sequencing errors or due to the presence of less-well studied bacterial species), we propose using the default cut-off (97%).



	Families cut-off			
Region	Family 1	Family 2	Family 3	Family 4
1-300	0.99	0.99	0.98	0.97
50-350	0.98	0.99	0.99	0.99
100-400	0.97	0.97	0.99	0.97

Figure 5:3. Illustration of the proposed lookup table approach for defining a dynamic threshold for OTU clustering of 16S rRNA gene sequencing reads

As a proof of concept, we applied this approach to the LTP reference database. Lookup tables where produced for each of the sliding window approaches mentioned above (windows of 300 and 500 bp, using a shift of 50 bp). Erroneous merging of different species within the same OTU was reduced with 45% compared to the default clustering cut-off of 3% (**TABLE 5:2**). As the average cut-off score calculated over the complete lookup table was 2.2 and 2.4% for 300 and 500 bp, respectively, we also compared our approach with a traditional OTU clustering approach using a 2% distance score as cut-off (instead of the default 3% used previously). Also in this case, we noticed an improvement of 30% with our method compared to the 2% rigid cut-off approach (15% of the species wrongly merged with our approach compared to 22% using the rigid approach). Similarly, applying a cut-off score of 98.65% – a value nowadays frequently used to distinguish different species in bacterial taxonomy (Kim et al. 2014)– still results in an improvement of 15% with our method.

Table 5:2. Illustration of the percentage of over-merged OTUs upon applying both the 97% default cut-off and our proposed lookup table,for two amplicon lengths (300 and 500 bp) and a sliding window of 50 bp. The lookup table was calculated using either the complete number of species for each family (column "lookup") or subsampled families containing at maximum of 7 species per family (column "lookup*").

Position		% of over-merged OTUs			Position		% of over-merged OTUs		
Start	**End**	**default**	**Lookup**	**Lookup***	**Start**	**End**	**default**	**Lookup**	**Lookup***
1	300	22	15	17	1	500	25	16	19
51	350	27	18	20	51	550	29	18	21
101	400	34	20	25	101	600	31	19	23
151	450	29	18	21	151	650	31	19	22
201	500	45	24	29	201	700	45	24	28
251	550	51	28	33	251	750	48	25	31
301	600	44	25	30	301	800	48	25	31
351	650	38	23	27	351	850	43	24	28
401	700	40	24	27	401	900	45	24	28
451	750	53	28	33	451	950	55	27	33
501	800	55	29	34	501	1000	47	25	31
551	850	48	26	32	551	1050	44	24	29
601	900	54	28	34	601	1100	49	25	30
651	950	63	32	36	651	1150	49	25	30
701	1000	48	26	32	701	1200	49	25	31
751	1050	48	26	31	751	1250	46	24	29
801	1100	47	26	30	801	1300	44	24	29
851	1150	49	26	32	851	1350	48	25	30
901	1200	48	25	32	901	1400	48	24	29
951	1250	41	24	28	951	1450	41	22	26
1001	1300	50	26	31					
1051	1350	53	28	34					
1101	1400	53	27	33					
1151	1450	49	25	30					
Average		44	24	29	**Average**		43	23	28

However, the question might be raised whether our lookup table method is sensitive to the size of the taxonomic family, i.e. can we also determine a reliable cut-off if the taxonomic family only contains a limited number of species, or do we only obtain reliable cut-off with sufficiently covered taxonomic families. Therefore, to mimic such

a situation we randomly subsampled each family to have at most seven species and subsequently recalculated the lookup table. This resulted in a similar level of improvement as previously reported i.e. an improvement of 36% over the 3% cut-off for the 500 bp region (compared with 46% when using the complete table, **TABLE 5:2**).

5.4.5. Comparative analysis using mock data

Next, we tested our lookup table method on a mock dataset consisting of 34 samples, each of them comprising 56 species (3 species represented by 2 strain each). After pre-processing, either our lookup table or the commonly used 97% cut-off was applied to cluster those reads into OTUs. As not all species were detectable in all 34 samples, in the most optimal case this clustering should result in a number of OTUs corresponding to the number of detectable species (i.e., a varying number of species, displayed in Additional Information 5:4) present in the mock community. From the analysis, our proposed lookup table approach lead to an average of 44.2 OTUs per sample compared to 41.6 OTUs achieved by the default 97% approach. Upon inspecting the percentage of lost species due to over-merging, our proposed approach lead to a loss of 8% of OTUs compared to 12% when applying the default cut-off of 97%.

However, by applying a too stringent cut-off, our method might also lead to the opposite effect, thereby assigning reads originating from the same species to different OTUs (called over-splitting below). We noticed an over-splitting in 4% of all OTUs when applying the default 97% cut-off, while our lookup table approach only resulted in a minor increase to 5%. Most of the over-splitting for both approaches was observed within the Spirochaetaceae family and could be explained by the presence of extremely erroneous reads and the presence of multiple non-identical paralogs of the 16S rRNA genes.

5.4.6. Application on real-life samples

To illustrate the performance of our lookup table methodology we applied both approaches on 64 human microbiome samples, for which three different amplicons were sequenced (V1-V3, V3-V5, and V6-V9). Our proposed lookup table methodology was able to produce a number of OTUs with less variability between the different amplicons i.e. the microbial diversity predicted with our methodology was more robust than the traditional approach (p-value < 0.01 based on a paired t-test comparing the standard deviations on the number of OTUs returned for the three regions between both approaches). Upon plotting numbers of OTUs obtained for each region against one of the other two regions, i.e. V1-V3 against V3-V5, V1-V3 against V6-V9, and V3-V5 against V6-V9, a closer correspondence can be observed between the number of OTUs returned for the different amplicons when applying our proposed lookup table (see **FIGURE 5:4**).

Finally, to illustrate to what extent applying our proposed approach would affect the species richness, we re-assessed the richness of the complete dataset of V3-V5 of human microbiome project dataset for healthy individuals (consisting of 28 million sequences from around 5,000 samples) using both approaches, producing 36,000 OTUs and 29,000 respectively for the lookup table and default approach (data not shown), i.e. an increase of 24%. These results suggest that default approach tends underestimate the species richness.

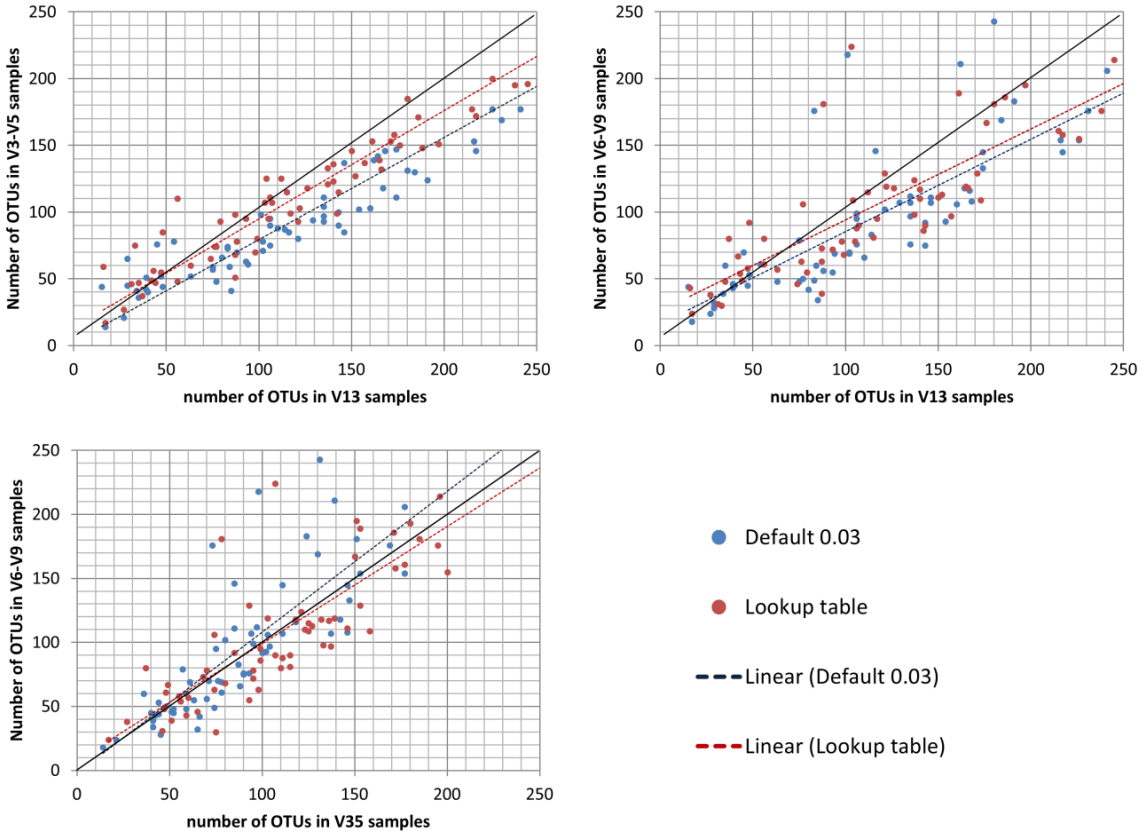


Figure 5:4. Plot illustrating the correspondence between the number of OTUs derived based on different amplicons (V1-V3, V3-V5, and V6-V9) as calculated for the HMP samples. The numbers of OTUs obtained for each amplicon were plotted against each other for all pairwise combinations. Results produced using the default approach (fixed cut-off of 3%) are coloured blue, results produced using our lookup table methodology in red. The corresponding linear fit (coloured in red or blue for both approaches respectively) summarizes the relation between the number of OTUs returned for different amplicons for both approaches; the black line represents the ideal scenario i.e., a perfect correspondence between the number of OTUs returned for both amplicons.

5.5. Discussion

In prokaryotic taxonomy, bacterial species are delineated on the basis of both genotypic and phenotypic criteria. Sequence analysis of (near) entire 16S rRNA genes cannot be used as a sole criterion to delineate species because this gene lacks resolution to distinguish between closely related species. Yet, organisms that share less than 98.65% of their (near) entire 16S rRNA genes consistently represent different

species (Kim et al. 2014). In traditional bacterial systematics species were defined as phenotypically coherent groups of strains sharing a certain percentage of DNA-DNA hybridization (Vandamme et al. 1996).

The latter DNA-DNA hybridization experiments are being replaced by whole genome sequence based parameters such as the degree of average nucleotide identity of shared genes (Richter & Rosselló-Móra 2009) or the genome-to-genome distance parameter (Meier-Kolthoff et al. 2013). Species delineated this way comprise phenotypically coherent strains that share a varying percentage of core genes (up to 70%) and that have a unique set of accessory genes that commonly represents up to 30% of their gene content (Lapierre & Gogarten 2009). Species defined this manner may thus group strains with an impressive functional diversity and 16S rRNA amplicon based diversity assessments cannot take this into account. Analysis of 16S rRNA amplicon datasets should therefore avoid further underestimation of species diversity by applying rigorous analysis methods.

After high-throughput sequencing and preprocessing the obtained 16S rRNA amplicon products, sequencing reads are clustered together based on sequence identity (mostly 97% as cut-off), a process often referred to as OTU clustering. Subsequently, those results are used as a proxy for a taxonomic species.

This fixed cut-off of 97% % is selected for practical reasons, as it offers a compromise between potential inflation of the number of OTUs due to sequencing errors and a cut-off used for taxonomic classification. However, most current high-throughput sequencing technologies only allow producing short reads covering a span of one or two variable regions only. Therefore, applying such a fixed cut-off implicitly assumes that all variable regions in all taxa evolve at the same rate, an assumption which is

certainly not met – as shown in previous experiments (Schmalenberger et al. 2001; Yu & Morrison 2004; Clarridge 2004; Schmidt et al. 2014; Schloss & Westcott 2011; Yarza et al. 2014) as well as the current work.

The first aim of this work was to confirm that the amount of sequence variation – and as such the evolutionary rate – was differing between different taxonomic families depending on the selected variable region, as already observed previously (Schmalenberger et al. 2001; Yu & Morrison 2004; Clarridge 2004; Schmidt et al. 2014; Schloss & Westcott 2011; Yarza et al. 2014).

The family level was used as it has been characterized as being more stable than other lower taxonomic levels (Yarza et al. 2014; Konstantinidis & Tiedje 2005), yet providing a good taxonomic resolution. Our results showed that the level of conservation of each variable region could differ dramatically between the various families, even when belonging to the same higher taxonomic class, as shown in **FIGURE 5:1**.

In addition, the level of conservation is not always consistent over all the various regions within one family, which is in line with previous findings (Chakravorty et al. 2007; Liu et al. 2008; Jeraldo et al. 2011; Kumar et al. 2011; Soergel et al. 2012). These observations have important consequences, as measurements of diversity for the same environment might lead to significantly different results when based on amplicon sequencing data derived for two amplicons covering different 16S rRNA variable regions, thereby hampering a solid comparison between studies focussing on different amplicons.

Clustering the 16S rRNA gene sequences as available in the LTP database, our results show a clear over-merging of different taxonomic species into one OTU (up to 44%), which in practice would lead to an underestimation of the bacterial diversity. Such over-merging is patently obvious when dealing with short sub-regions of these 16S rRNA genes (representing different amplicons), but, interestingly, even when using the full length sequences an over-merging of 35% of the species was observed as already shown in previous studies (Fox et al. 1992; Ash et al. 1991; Rosselló-Mora & Amann 2001; Staley 2006; Janda & Abbott 2007; Stackebrandt & Ebers 2006; Achtman & Wagner 2008; Kim et al. 2014). This implies that the database contains bacteria which are assigned to different species despite the fact that they show a sequence identity exceeding the specified cut-off (Ash et al. 1991; Rosselló-Mora & Amann 2001; Stackebrandt & Ebers 2006; Kim et al. 2014). This is not surprising as the classification of bacterial taxonomy is not solely based on the 16S rRNA gene sequence identity, but also requires a minimum ANI and DDH value.

Taking together two facts mentioned above, being that 1) current high-throughput sequencing technologies only allow to sequence a sub-region of the full length 16S rRNA gene, and 2) different variable regions are showing a different degree of conservation between different taxonomic families, urged us to develop a dynamic OTU clustering approach, an ad-hoc heuristic which is first grouping all sequencing reads based on their taxonomic family, and next uses a family-dependent cut-off for clustering the short sequencing reads into OTUs rather than using a generally applied fixed cut-off. Moreover, this dynamic clustering cut-off score is also depending on the exact sub-region of the 16S rRNA gene which is amplified.

Our proposed family-dependent cut-off approach was able to cluster OTUs with an increased efficiency compared to the default 97%, leading to a reduction of over-merging by 46%. This level of improvement was independent of the coverage of the different families (i.e. the number of species within a family). As such, the same level of efficiency can be extrapolated when dealing with novel species of existing families or with families with a high or low level of species sequenced.

When applying this analysis workflow on mock samples harbouring closely related species, the approach proposed within this work allowed to estimate the richness more accurately than the default pipeline using 97% as general cut-off. Upon applying both approaches on a dataset generated in the context of the Human Microbiome Project, a significantly closer correspondence was reported for the lookup table methodology versus the traditional approach between the number of OTUs obtained when various regions of the 16S rRNA gene were used for the same biological sample.

5.6. Conclusion

Conclusively, these results points out the different evolutionary rates within the different variable regions of the 16S rRNA gene will have a significant impact on the OTU clustering results derived from 16S rRNA amplicon sequencing data. As such caution should be taken when comparing diversity indices derived for the same environment however using different amplicons. To circumvent this problem, we developed a methodology that calculates a dynamic and evolutionary compliant OTU clustering cut-off per taxonomic family for each constructed amplicon. DynamiC (Dynamic Cut-off) software to create such a lookup table and apply it for any amplicon sequencing dataset using a preferred taxonomic reference database is freely available via https://github.com/M-Mysara/DynamiC/. In principle it can be used for any amplicon sequencing dataset, such as 16S rRNA, 18S rRNA, 23S rRNA or 28S rRNA. DynamiC translates amplicon sequencing data into high-quality OTUs tendering robust diversity estimates.

Chapter 6

From Reads to Operational Taxonomic Units: an Ensemble Processing Pipeline for MiSeq Amplicon Sequencing Data

Mohamed Mysara[1,2,3,4], Mercy Njima[1], Natalie Leys[1], Jeroen Raes2,[3,4] and Pieter Monsieurs[1]

[1] *Unit of Microbiology, Belgian Nuclear Research Centre (SCK-CEN), Mol, Belgium.*
[2] *Department of Bio-engineering Sciences, Vrije Universiteit Brussel, Brussels, Belgium.*
[3] *VIB Center for the Biology of Disease, VIB, Leuven, Belgium.*
[4] *Department of Microbiology and Immunology, REGA institute, KU Leuven, Belgium.*

This chapter is based on the following submitted publication:

Mysara M, Njima M, Leys N, Raes J, Monsieurs P. (2016). From Reads to Operational Taxonomic Units: Comparative Analysis of Amplicon Sequences Processing pipelines, GigaScience (submitted)

6.1. Abstract

The development of high-throughput sequencing technologies has provided microbial ecologists with an efficient approach to assess bacterial diversity at an unseen depth, particularly with the recent advances in the Illumina MiSeq sequencing platform. However, analysing such high-throughput data is posing important computational challenges, requiring specialized bioinformatics solutions at different stages during the processing pipeline, such as assembly of paired-end reads, chimera removal, correction of sequencing errors and clustering of those sequences into Operational Taxonomic Units (OTUs). Individual algorithms grappling with each of those challenges have been combined into various bioinformatics pipelines, such as mothur, QIIME, LotuS and USEARCH. Using a set of well-described bacterial mock communities, state-of-the-art pipelines for Illumina MiSeq amplicon sequencing data are benchmarked at the level of the amount of sequences retained, computational cost, error rate and quality of the OTUs. In addition, a new pipeline called OCToPUS is introduced, which is making an optimal combination of different algorithms. Noticeable variability is observed between the different pipelines in respect to the monitored performance parameters, where in general the amount of retained reads is found to be inversely proportional to the quality of the reads. By contrast, OCToPUS achieves the lowest error rate, minimum number of spurious OTUs, and the closest correspondence to the existing community, while retaining the uppermost number of reads when compared to other pipelines. The newly introduced pipeline translates Illumina MiSeq amplicon sequencing data into high-quality and reliable OTUs, with improved performance and accuracy compared to the currently existing pipelines.

6.2. Introduction

The application of new high-throughput sequencing technologies to assess microbial diversity is a fast-evolving discipline. The high through-put capacity of those technologies and the absence of the need to culture and isolate microbial species provides researchers in the field with a very powerful technology. The sequencing of the 16S rRNA gene as phylogenetic marker gene is very often used approach to assess the microbial diversity. The short length of the reads currently produced by most sequencing technologies is an important limitation, as those reads only cover one or a few variable regions within the 16S rRNA gene. However, this drawback is largely compensated by the reduction in economic cost and increase in throughput compared to traditional approaches.

The Roche 454 Pyrosequencing technology was the first high-throughput sequencing technology to be used in microbial ecology studies (Edwards et al. 2006; Sogin et al. 2006), followed by other technologies such as IonTorrent (Jünemann et al. 2012), and Illumina (Gloor et al. 2010) and PacBio (Fichot & Norman 2013). The introduction of the Illumina MiSeq platform, offering paired-end reads nowadays up to 2 x 300 bp at a reasonably high throughput combined with the announcement of Roche to shut down its 454 sequencing services by 2016, led to a shift towards the former technology. Therefore, results presented within this work are focused on sequencing data obtained from the Illumina MiSeq platform.

The ultimate goal of these amplicon sequencing approaches is to obtain a holistic view on the microbial composition within a sample, mostly obtained via binning the sequencing reads based on their sequence identity to each other, resulting in clusters of reads, commonly referred to as Operational Taxonomic Units (OTUs).

Eventually, in the ideal scenario each OTU should represent an actual bacterial species. Nonetheless, many researchers have reported an inflation of the number of OTUs when sequencing mock communities. Such an approach of using a well-defined mixture of microbial cells allows gaining insight into the numerous sources of errors potentially hampering the correct interpretation of amplicon sequencing data (Kozich et al. 2013; Mysara et al. 2016; Edgar & Flyvbjerg 2015).

A first source of errors originates from chimera formation within the PCR amplification step, thereby creating a chimeric sequence which consists of two or more fragments from distinct species (Ashelford et al. 2005; Odelberg et al. 1995; Judo et al. 1998; Smyth et al. 2010) As those chimeras will propagate in the same way as any other DNA sequence, they can take up to 30% of all unique sequencing reads. Falling short in removal of these artificial sequences will have an impact on the diversity estimates, since chimeras that go undetected will be interpreted as novel species (Wang & Wang 1996; Wang & Wang 1997).

Secondly, the high-throughput character of the new sequencing platforms comes at the cost of a decreased accuracy, in such a way posing important challenges at the level of data analysis. Illumina sequencing platforms suffer mainly from substitutions-type miscalls which frequently accompanies GC-rich regions (Allhoff et al. 2013; Tsai et al. 2014; Schirmer et al. 2015) or which are caused by improper phasing/prephasing (Kircher et al. 2009), or which resulted from the high correlation of emission spectra between A and C as well as G and T (Kircher et al. 2009; Bentley et al. 2008; Rougemont et al. 2008).

Additionally, to obtain reads with an acceptably low error rate, both forward and reverse reads needed to be at least partially overlapping, thus allowing the combination

of the prediction in both reads to generate a consensus amplicon (Kozich et al. 2013). Yet, as this overlapping region spans those parts of the reads with the lowest quality scores, such practice can still introduce errors, especially when conflicts between both reads occur.

Numerous bioinformatics algorithms have been developed for the different steps within the workflow of amplicon sequencing data produced by the Illumina MiSeq platform, , such as: a) paired-end assembly, by merging both forward and reverse reads into one consensus sequence, b) quality filtering, via filtering reads with low sequencing quality, c) denoising i.e. correction of sequencing errors, d) the removal of chimeric reads and e) clustering via binning the sequencing reads into OTUs based on their sequence identity to each other. An overview of previously developed algorithms is given in **TABLE 6:1**. Integration of those single-step tools into pipelines covering the whole processing stage, resulted in different workflows including MG-RAST (Meyer et al. 2008), mothur (Schloss et al. 2009), QIIME (Caporaso et al. 2010), USEARCH (Edgar 2010), LotuS (Hildebrand et al. 2014), and BioMaS (Fosso et al. 2015).

Various efforts have been made to compare the different individual tools developed for each preprocessing step, e.g. there exist benchmark studies comparing the paired-end assemblers (Edgar & Flyvbjerg 2015; Mysara et al. 2016), denoising tools (Kozich et al. 2013; Schloss et al. 2011; Mysara et al. 2016; Edgar & Flyvbjerg 2015), chimera detection tools (Mysara, Saeys, et al. 2015; Quince et al. 2011) and clustering algorithms (Westcott & Schloss 2015; Schloss & Westcott 2011; Edgar 2013; Sun et al. 2012). However, limited literature is available comparing pipelines, such as USEARCH, LotuS, mothur, and QIIME. Such a benchmarking analysis could be provide crucial information to microbial ecologists in term of accuracy, computational time, and retained sample size, as such offering guidance towards the selection of the

appropriate pipeline. First initiatives to perform such benchmark have already led to interesting results (Plummer & Twin (2015), Hildebrand et al. (2014), Fosso et al. (2015), D'Argenio et al. (2014)). However, each of those comparative studies used either biological samples or simulated datasets, thus making it difficult to assess the quality in terms of error rate and OTU accuracy.

Table 6:1. Overview of the algorithms available for different steps within amplicon sequencing data analysis.

Step	Tools	Reference
paired-end assembler	FLASH	(Magoč & Salzberg 2011)
	PANDAseq	(Masella et al. 2012)
	COPE	(Liu et al. 2012)
	PEAR	(Zhang et al. 2014).
Quality filtering	trim.seqs(mothur)	(Kozich et al. 2013)
	split_libraries (QIIME)	(Bokulich et al. 2013)
	fastq_filter (USEARCH)	(Edgar & Flyvbjerg 2015)
Denoising	Pre-cluster	(Kozich et al. 2013)
	UNOISE	(Edgar & Flyvbjerg 2015)
	IPED	(Mysara et al. 2016)
Chimera Detection	Pintail	(Ashelford et al. 2005)
	Bellerophon	(Huber et al. 2004)
	ChimeraSlayer	(Haas et al. 2011)
	DECIPHER	(Wright et al. 2012)
	Perseus	(Quince et al. 2011)
	UPARSE	(Edgar 2013)
	UCHIME	(Edgar et al. 2011)
	CATCh	(Mysara, Saeys, et al. 2015)
Clustering	Dotur	(Schloss & Handelsman 2005)
	ESPRIT	(Sun et al. 2009)
	ESPRIT-Tree	(Cai & Sun 2011)
	CD-HIT	(Fu et al. 2012)
	Uclust	(Edgar 2010)
	GramCluster	(Russell et al. 2010)
	DNAClust	(Ghodsi et al. 2011)
	CROP	(Hao et al. 2011)
	Swarm	(Mahé et al. 2014)
	UPARSE.	(Edgar 2013)

In this work, a comprehensive comparison was made between mothur, QIIME, LotuS and USEARCH pipelines, in respect to reads throughput, error rate and OTU accuracy. We also propose within this work a novel pipeline that combines the advantages of different existing individual tools, which is entitled OCToPUS (Optimized CATCh, mothur, IPED, UPARSE and SPAdes). In contrast to previous comparative analyses

described above, we used mock community datasets, as such providing a benchmark that can be used to calculate the error rate and correspondence of the resulting OTUs with the actual microbial composition. Important to notice is that this work has no intention of comparing the underlying individual algorithms built-in within each pipeline. It rather treats the entire pipeline as a black box and assesses the accuracy using a unified evaluation process apart from the implemented individual algorithms.

6.3. Materials and methods

6.3.1. Data Description

The benchmark analysis within this work is founded on three publicly available Illumina MiSeq sequencing datasets obtained for three different mock communities. The first mock community – called MOCK1 – contains of 21 species, and the corresponding amplicon sequencing dataset covers the V3-V4 and V4 regions of the 16S rRNA gene, each amplicon sequenced in triplicate (run IDs 130403, 130417 and 130422). The second mock community (MOCK2) consists of 20 different organisms covering the V4 and V4-V5 regions, each of them sequenced in duplicate (named v4.I.1 and v4.I.05 (for V4), v4.v5.I.1 and v4.v5.I.11 (for V4-V5). The third mock community (MOCK3) consists of 12 species, is sequenced in triplicate (named M1, M2, M3) and covers the V3-V4 region. MOCK1 is available via (http://www.mothur.org/MiSeqDevelopmentData.html) under accession 130403, 130417 and 130422, MOCK2 is available via European Bioinformatics Institute Nucleotide Archive SRA under project ID PRJEB4688 and MOCK3 is available via National Center for Biotechnology Information SRA under project ID: SRP066114. The detailed composition, library preparation and sequencing on the Illumina MiSeq platform are described in detail in the respective publications for MOCK1 (Kozich et al. 2013), MOCK2 (Nelson et al. 2014), MOCK3 (Mysara et al. 2016).

6.3.2. Standardization of the Pipelines

The samples were analysed using four pipelines: QIIME (Version 1.8.0), mothur (Version 1.33.3), LotuS (Version 1.506), USEARCH (Version v8.1.1861_i86linux32) and a new pipeline OCToPUS introduced within this work. In general, the standard commands were used for each pipeline, i.e. using the default parameters. However, in order to allow for a fair comparison on the number of spurious OTUs, OTUs were not rejected based on their relative abundance or their taxonomic classifications in any of the pipelines. This necessitated the deactivation of default singleton removal option in UPARSE and skipping the default *remove.lineage* command in mothur or putting the keepUnclassified parameter in LotuS. For the same reason the reference based mode of the chimera detection for all pipelines was not included. A detailed description of the commands used within each pipeline is described below, and a schematic overview of the different steps is summarized in **FIGURE 6:1**.

Mothur

In general, the Standard Operation Procedure of mothur for analysing 16S rRNA amplicon sequencing data (http://www.mothur.org/wiki/MiSeq_SOP, d.d. 2015-11-23) is used as guideline. In a first step, the forward and reverse reads are merged using the *make.contigs* command. Based on the quality scores, a heuristic has been implemented to resolve conflicts between both reads, thereby replacing problematic conflicts with "N". Reads exhibiting any ambiguous positions or containing a more than 8-base homopolymer are subsequently removed using the *screen.seqs* command. Next, reads are aligned to the SILVA reference database (Pruesse et al. 2007) using the *align.seqs* command. Those reads that fail to align to the correct location within the 16S rRNA gene (Schloss 2010; Schloss 2009; Schloss 2013) are culled using the *screen.seqs*

command. Aligned reads are simplified (via removing non-informative columns (using the *filter.seqs* command), dereplicated (via the *unique.seqs* command), and denoised with mothur implementation of the Single Linkage Preclustering algorithm (Huse et al. 2010) via, the *pre.cluster* command. The resulting reads are screened for presence of chimeras using UCHIME (Edgar et al. 2011) via the *chimera.uchime* command. Finally, sequences are clustered into OTUs using the *cluster.split* command.

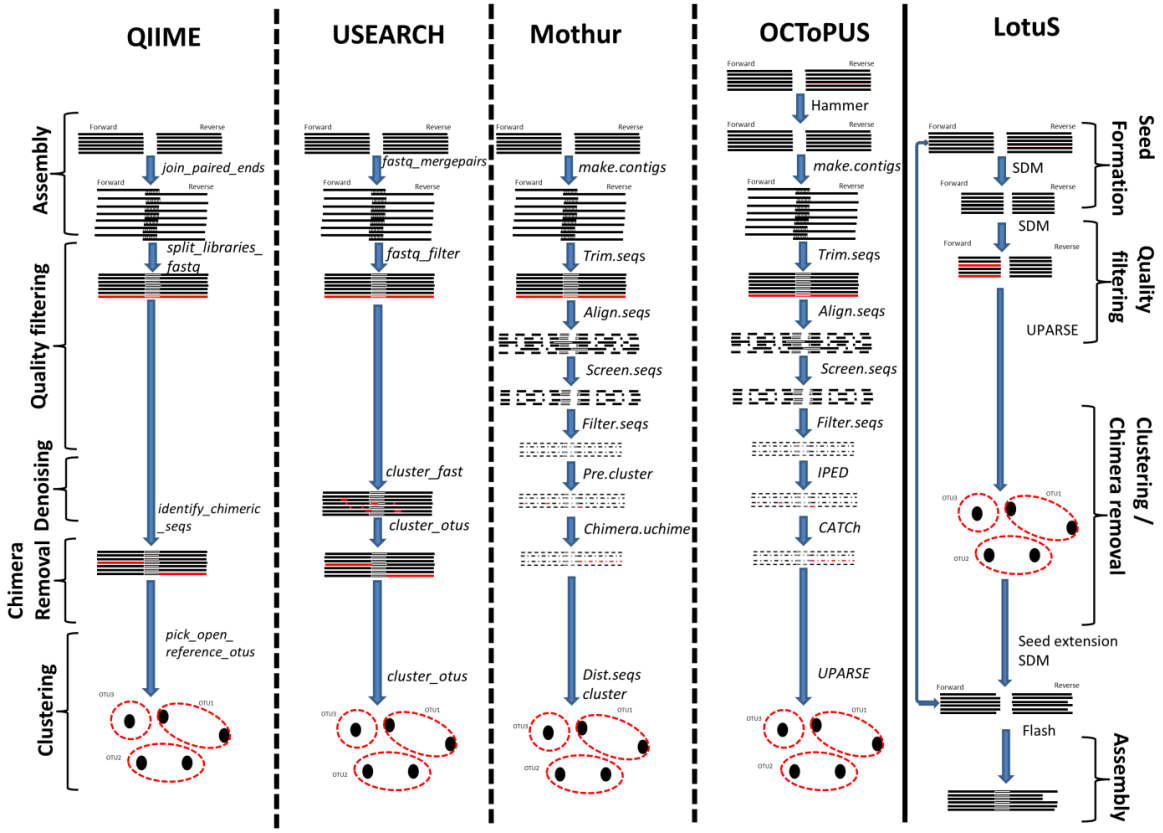


Figure 6:1. Overview of the different steps within each pipeline.

USEARCH

Following the recommendations by Edgar and Flyvbjerg (2015) and the online published workflow (http://drive5.com/usearch/manual/uparse_pipeline.html), both forward and reverse reads are merged by aligning them using the *fastq_mergepairs* command. The *fastq_filter* command is used to assess the expected number of errors, as described

in (Edgar & Flyvbjerg 2015), and filter the reads accordingly. Dereplication is performed via the *derep_fulllength* command, followed by denoising via *cluster_fast* which is the implementation of the UNOISE algorithm (Edgar & Flyvbjerg 2015). Via the *sortbysize* command reads are arranged in descending order of abundance, followed by the *cluster_otus* command that combines both the OTU clustering and chimera (*de novo*) removal step. Reads are mapped to the final OTUs list using *usearch_global* command to assign abundances to each OTU and formulate the OTU-table.

QIIME

Following the recommendations on QIIME website (http://qiime.org/), first both forward and reverse reads are merged via the *join_paired_ends.py* command, an implementation of the fastq-join approach (Erik Aronesty, 2011). Next a quality filtering step based on the Phred scores is applied, as described in Bokulich et al (Bokulich et al. 2013) via *split_libraries_fastq.py*. Chimeras are identified using *identify_chimeric_seqs.py* command (using the usearch61 option that runs the UCHIME algorithm), and subsequently removed via *filter_fasta.py*. OTU clustering is performed using the *pick_open_reference_otus.py* command utilizing the default UCLUST algorithm and Greengenes as reference database.

LotuS

LotuS requires specifying all parameters in a single command, which is different from the step-wise approach of previous pipelines. First, LotuS reads the mapping file specifying the input fastq files, which are subsequently demultiplexed and quality filtered using the simple demultiplexer (sdm) algorithm (Hildebrand et al. 2014). Reads are trimmed into "seeds" with a length of 170 bases, which are clustered and checked for chimera using the UPARSE algorithm to formulate the OTU table. Next, the seed

sequences of the shortlisted OTUs are extended and assembled via the sdm and Flash (Magoč & Salzberg 2011) algorithms, respectively, of which the output are used as the representative sequences of the OTUs.

OCToPUS

Within this work a new pipeline was developed which utilizes the benefits of various tools and state-of-the-art algorithms, described as an Optimized CATCh, mothur, IPED, UPARSE and SPAdes, abbreviated as OCToPUS. First, both forward and reverse reads are quality checked via looking at k-mer frequency to identify potential false k-mers using the Hammer algorithm (Medvedev et al. 2011) implemented in the SPAdes tool (Bankevich et al. 2012). Next, reads are assembled via the mothur *make.contigs* command, followed by screening, aligning, filtering and dereplication, similar to what was described in the mothur approach. Next, reads are denoised using the IPED algorithm, which applies an artificial intelligent classifier to identify and correct positions likely to be erroneous (Mysara et al. 2016). Chimera detection is performed via the CATCh algorithm, a second layer classifier that ensembles the scores of various chimera detection tools into a more accurate classification (Mysara, Saeys, et al. 2015). Subsequently, we apply the UPARSE clustering approach as implemented in USEARCH, using the *cluster_otus* and *usearch_global* commands to assign an abundance level to each OTU.

6.3.3. Evaluation Criteria

Comparison of the different pipelines was performed using four different parameters: 1) number of reads rejected, 2) the error rate, 3) the number of OTUs and their composition and 4) computation time. The number of reads retained within the different pipelines was calculated via the mothur *summary.seqs* command at different

stages within the workflow, i.e. after paired-end assembly, after quality filtering and after chimera removal. Due to different order of the processing steps in LotuS as illustrated in **FIGURE 6:1**, only the final number of reads can be reported.

Secondly, as the microbial composition of the mock sample is known – and as such the reference sequence of the corresponding 16S rRNA genes – actual error rates were calculated via the mothur *seq.error* command. Error rates were reported twice: once after the chimeric reads were accurately removed in order to have an idea on the sequencing error rate excluding the chimeric reads, and a second time after applying a regular chimera removal tool as implemented within each pipeline, thereby giving a more realistic estimation of the total error rate that will be retained within the sequencing data. As LotuS – unlike the other pipelines – does only perform the paired-end assembly step after creating the OTUs, it is not possible to calculate the error rate of assembled reads prior to clustering.

Operational taxonomic units (OTUs) were assessed in a quantitative as well as a qualitative way. For the quantitative approach, we calculated the number of OTUs produced via each pipeline per sample. Those numbers were plotted using rarefaction curves where the number of OTUs are shown in the vertical axis and read counts in the horizontal axis, reflecting the influence of sequencing depth on the number of OTUs. Additionally, we performed a qualitative analysis, following a similar approach as described in Edgar (2013), where the OTUs were classified into four different categories: 1) original (more than 97% sequence identity to a species within the mock community), 2) chimeric (similar to two or more species within the mock community), 3) contaminant (non-intended read with high sequence identity match to a species not in the targeted community), and 4) others (not fulfilling any of previous criteria).

Lastly, the computation time was calculated for the different steps within each pipeline: paired-end assembly, quality filtering (with denoising when integrated in the pipeline), chimera removal and OTU clustering using eight Intel Xeon E5-2640 2.50 GHz CPUs. The six samples of MOCK 1 were used for this analysis (with a coverage ranging from 20,000 to 700,000 reads).

6.4. Results and discussion

In this work, a benchmark analysis was conducted between our newly introduced OCToPUS pipeline and the existing state-of-the-art pipelines QIIME, USEARCH, LotuS, and mothur. A set of performance parameters were defined to assess each pipeline, i.e. the number of reads retained, the error rates, the computational time and the quality of the OTU clustering results. For this purpose, 13 mock samples with a known composition and originating from three different studies (six sequencing runs) were processed by all four pipelines, allowing us to calculate of the four performance parameters for each pipeline. Although each pipeline was initially exposed to the same number of reads, the number of reads retained by each of the workflows was dramatically differing between each of them. The percentages of rejected reads were on average 23%, 24%, 26%, 26%, and 47% for LotuS, OCToPUS, QIIME, mothur and USEARCH, respectively (see Additional Information 6:1). Important to notice is that the number of reads lost within a certain step is differing dramatically between different pipelines (see **FIGURE 6:2**), e.g. most of the reads are thrown away by QIIME during the assembly phase, while most of the reads are rejected by USEARCH in the quality filtering step.

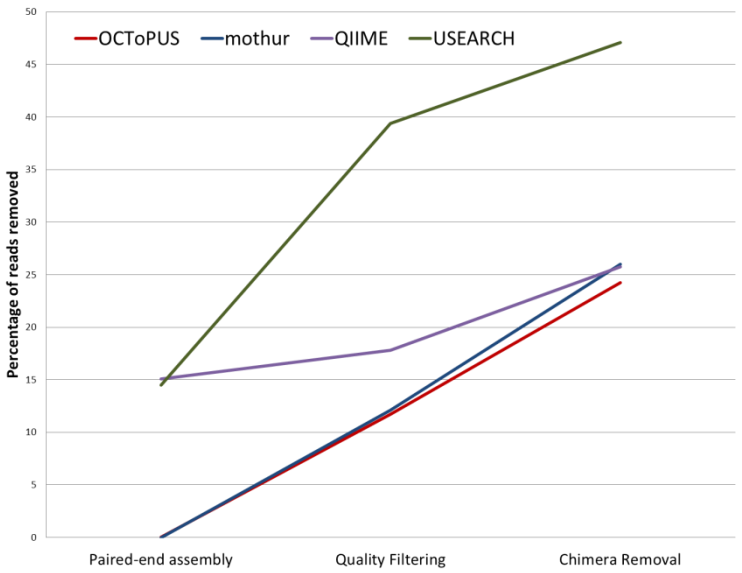


Figure 6:2. Average number of reads removed within the various pipelines, due to improper assembly, quality filtering or chimera removal. Due to different order of the processing steps in LotuS, this pipeline could not be included in the figure (on average LotuS removes 23% of the reads).

As the main reason for rejecting those reads was to get rid of poor quality or chimeric sequences, it was utterly important to assess their influence on the error rate obtained with each approach. In a first scenario chimeras were identified by using the known reference sequences for each community, and subsequently the error rate was calculated. It is important to notice that such an analysis can only be performed for mock communities, and is performed within this context purely as benchmark analysis. Within this context, OCToPUS obtained an error rate of 0.08% on average, while USEARCH, mothur and QIIME reduced the overall error rate to 0.14%, 0.15%, and 0.47% respectively averaged over all mock communities (see **TABLE 6:2**). With the exception of OCToPUS, there was a strong correlation between the amount of rejected reads and the extent to which the error rate has been reduced. Additionally, we assessed the error rate within the second scenario, where the removal of chimeras occurs using a traditional chimera detection algorithm, as such reflecting a real-life scenario. OCToPUS was able to reduce the error rate to 0.19% while USEARCH,

mothur and QIIME achieved 0.23%, 0.24%, and 0.59% respectively averaged over all mock communities (see **TABLE 6:2**). Due to the presence of some undetected chimeras, an inflation of the error rate was reported for the second scenario compared to the first one. Nonetheless, in both scenarios the OCToPUS pipeline was deemed successful in acquiring the highest quality in respect to the error rate of the sequencing reads, without affecting the number of reads retained. As discussed in the methods LotuS could not be included in this analysis.

Table 6:2. The error rates for the different samples after applying various pipelines, either with complete removal of chimeric reads (via the *seq.error* command), or after applying the chimera removal algorithm embedded within the workflow in question.

Sample ID	Chimera Absent				Chimera removal algorithms			
	QIIME	Mothur	USEARCH	OCToPUS	QIIME	Mothur	USEARCH	OCToPUS
130403(**V3-V4)**	0.0022	0.0006	0.0003	0.0003	0.0023	0.0008	0.0005	0.0004
130417(**V3-V4)**	0.0018	0.0005	0.0003	0.0003	0.0019	0.0007	0.0005	0.0004
130422(**V3-V4)**	0.0023	0.0012	0.0008	0.0009	0.0023	0.0011	0.0010	0.0009
130403(**V4)**	0.00055	0.00013	0.00010	0.00005	0.00208	0.00167	0.00161	0.00126
130417(**V4)**	0.00049	0.00010	0.00008	0.00003	0.00187	0.00150	0.00147	0.00114
130422(**V4)**	0.00048	0.00010	0.00008	0.00003	0.00182	0.00144	0.00141	0.00109
v4.I.1	0.00079	0.00007	0.00002	0.00002	0.00087	0.00013	0.00006	0.00003
v4.I.05	0.00087	0.00010	0.00002	0.00002	0.00099	0.00020	0.00008	0.00003
v4.v5.1	0.0257	0.0084	0.0075	0.0041	0.0241	0.0069	0.0049	0.0047
v4.v5.I.11	0.0218	0.0060	0.0072	0.0031	0.0218	0.0044	0.0047	0.0032
M1(**V3-V4)**	0.0014	0.0006	0.0006	0.0005	0.0052	0.0039	0.0042	0.0038
M2(**V3-V4)**	0.0014	0.0007	0.0006	0.0005	0.0058	0.0045	0.0047	0.0043
M3(**V3-V4)**	0.0011	0.0006	0.0005	0.0005	0.0052	0.0041	0.0041	0.0039
Average	**0.0047**	**0.0015**	**0.0014**	**0.0008**	**0.0059**	**0.0024**	**0.0023**	**0.0019**

The negative effect of sequencing errors and PCR artefacts are expected to influence the amount of spurious OTUs, thus a successful removal of these errors should ideally be reflected in a decrease of the number of OTUs. Although the number of OTUs are affected by the number of reads and the level of complexity within the mock samples, (Kozich et al. 2013), it has commonly been used by others as a metric for sequencing quality (Schloss et al. 2011; Kozich et al. 2013; Huse et al. 2010; Reeder & Knight 2010; Mysara, Leys, et al. 2015; Edgar & Flyvbjerg 2015; Quince et al. 2011; Kunin et al. 2010).

Thus, we calculated the average number of spurious OTUs –exceeding the expected number of OTUs –for all samples. OCToPUS produced on average 65 OTUs, while USEARCH, LotuS, mothur and QIIME produced 95, 208, 236, and 295 OTUs respectively (see Additional Information 6:2). Using the rarefaction curves we could demonstrate that the OCToPUS pipeline was able to achieve the least amount of spurious OTUs with increasing sequencing depth, followed by USEARCH, LotuS, mothur and QIIME (see **FIGURE 6:3**). Nonetheless, it is important to stress that the number of reads removed by USEARCH –the second best performance – is drastically lower compared with the other pipelines, as illustrated in Additional Information 6:1.

Achieving the least number of spurious OTUs, does not automatically imply that it will return OTU clustering results that reflect accurately the microbial composition within the mock community. Therefore, we performed an additional analysis to qualitatively assess the composition of the OTUs produced via each pipeline. Based on the classification used in Edgar et al. (Edgar 2013), the percentage of original species, escaped chimeras, existing contaminants and other unidentifiable sequences were calculated (see methods). Based on **FIGURE 6:4**, USEARCH, OCToPUS and QIIME report the most accurate correspondence to the original species, and USEARCH and OCToPUS report the least amount of chimera. The remainder of the OTUs represented contaminating reads or unidentified sequences (possibly formed via a combination of contaminants and PCR or sequencing errors). For the MOCK1 (V3-V4) and MOCK2 (V4-V5) samples USEARCH obtained a better prediction of the microbial community than OCToPUS. However, it is important to notice that USEARCH throws away on average 94% and 59% of the sequencing reads in MOCK1(V3-V4) and MOCK2 (V4-V5) samples respectively during processing – as such limiting the analysis to a small fraction of reads – while OCToPUS rejects on average 13 and 46% of the reads

respectively. It is important to point out that there is strong correlation between the depth of the coverage and the amount of artefacts (particularly chimeras) with the samples, for instance the number of reads in 130403(V3-V4) is at least an order of magnitude lesser than 130403(V4), the detailed information regarding the number of reads can be seen in TABLE 3:1.

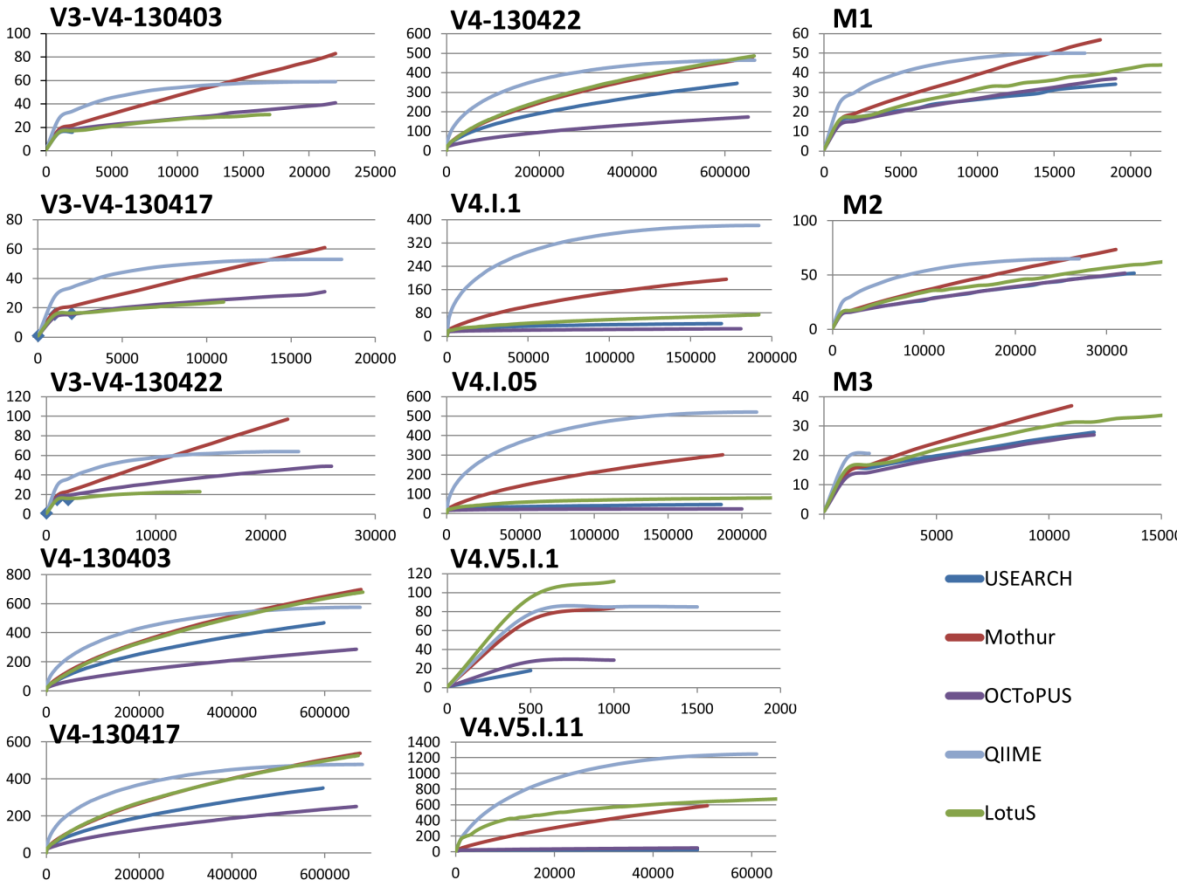


Figure 6:3. Rarefaction curves of the different samples. In the X-axis the sequencing depth is given, in the Y-axis the amount of OTUs returned by each pipeline.

Similarly, LotuS throws away only 53% of the MOCK1 sequencing data (V3-V4), yet obtaining a slightly better prediction compared to OCToPUS. Finally, we evaluated the number of species that were split over more than one OTU, and the species that were absent in the OTU production. All approaches were able to identify all species within the MOCK1 and MOCK2 communities. However, only OCToPUS and USEARCH

reported an average of 1 OTU per species, while LotuS reported 1.4 OTUs per species, mothur 1.7 OTUs per species and QIIME 5.2 OTUs per species, indicating a more pronounced over-splitting effect, it was also reported with MOCK3 samples (see Additional Information 6:3).

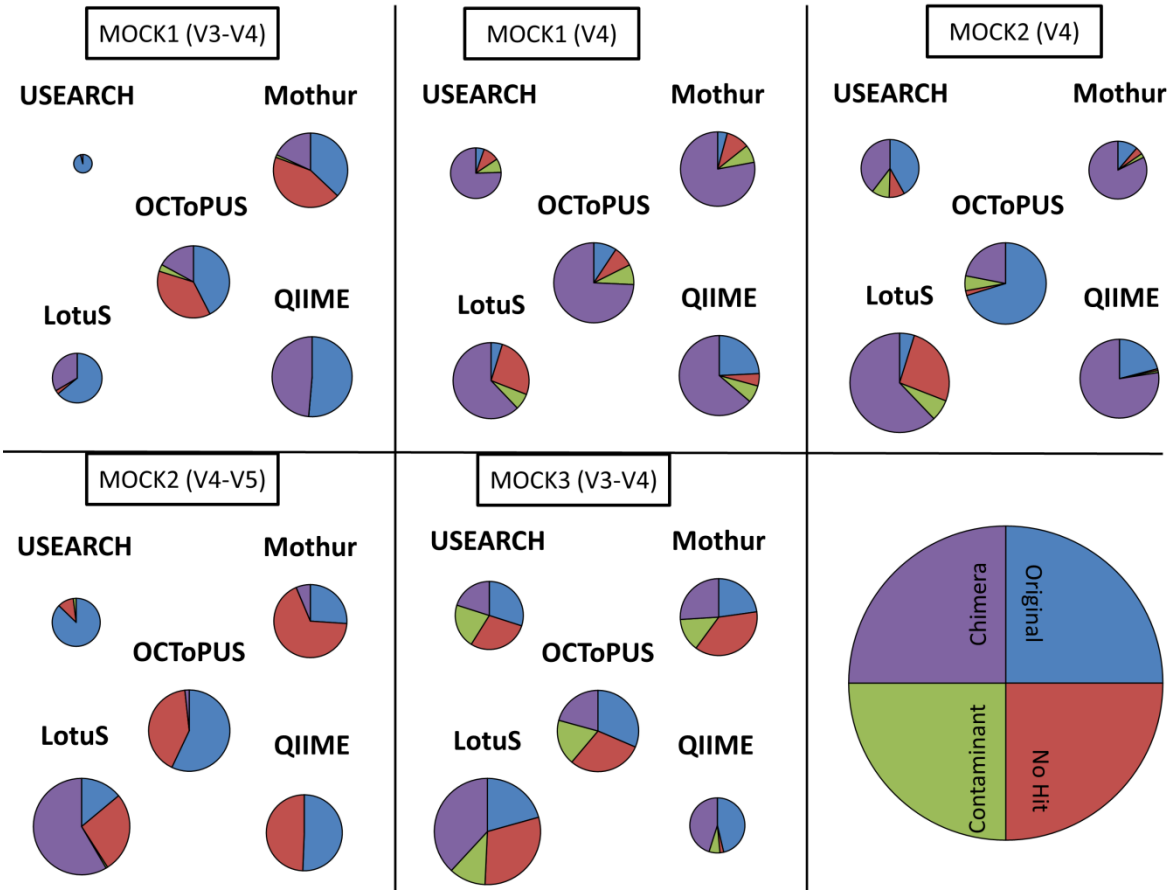


Figure 6:4. Composition of the OTUs produced via the various approaches, classified into different categories: original (blue), chimeric (violet),contaminant (green) and no hit (red). The size of the circles is representative for the number of reads retained after running each pipeline (exact percentages can be found in Additional Information 6:1).

The computational cost for USEARCH and LotuS was dramatically lower compared to the other pipelines, as it only required a few seconds to process the six samples of MOCK1. Mothur, OCToPUS, and QIIME required 2.1, 2.7, and 3 minutes respectively, (see Additional Information 6:). The computational time for mothur is evenly distributed across the different steps. For OCToPUS the most time-consuming step is

the paired-end assembly (including the preceding pre-assembly error correction) and chimera removal (requiring the execution of three chimera detection algorithms). Concerning QIIME, most of the computational time was dedicated to the OTU clustering step. As discussed earlier, the added computational burden for OCToPUS was overshadowed by the quality of the processed data.

6.5. Conclusion

Conclusively, our proposed pipeline OCToPUS combines the advantages of mothur, CATCh, IPED, UPARSE and SPAdes and was on average able to achieve the lowest error rate, the minimum number of spurious OTUs and the closest correspondence to the existing community without compromising the number of reads retained. With the exception of USEARCH, the required computation time was in line with the other pipelines. All included algorithms are freely available – with exception of the USEARCH licence that can be obtained from its author upon registration. Finally, our newly proposed OCToPUS pipeline is able to translate amplicon sequencing data into high-quality OTUs. OCToPUS is available via this link (https://github.com/M-Mysara/OCToPUS), under GNU licence.

Chapter 7

General Discussion and Perspectives

7.1. General discussion

High-throughput sequencing of the 16S rRNA gene has been widely adopted by microbiologists to assess microbial diversity, mostly due to the fact that it allows overcoming the bias of the bacteria culturing. Multiple studies have used 16S rRNA amplicon sequencing methods, an approach easily producing millions of short-read fragments covering specific regions of the 16S rRNA gene. Analysing those sequences to obtain an accurate estimation of microbial diversity has become an important challenge pursued for many microbial ecologists.

On two occasions I have been asked, "Pray, Mr. Babbage, if you put into the machine wrong figures, will the right answers come out?" I am not able rightly to apprehend the kind of confusion of ideas that could provoke such a question. (Babbage 1994)

The high-throughput characteristic of amplicon sequencing comes at the cost of a decreased accuracy. As such several factors have been associated with the artificial inflation of operational taxonomic units (OTUs), as discussed thoroughly throughout this PhD thesis. The principle of *garbage in, garbage out* governs the data processing, thus one should balance between the sequencing throughput (in terms of reads length and count) and the sequencing quality (in respect to the error rate and presence of chimeric sequences). Obtaining long reads with high quality and at the same time minimizing the number of rejected reads is the main aim of the *pre-processing* procedure. Additionally, these measures would drastically influence the quality of operational taxonomic units (OTUs) produced by the clustering step , within the *processing* step (see introduction section 1.4).

If I have seen further it is by standing on the shoulders of giants

Sir Isaac Newton (Fawcett et al. 2015)

This PhD work contributes to the emerging field of 16S rRNA amplicon sequencing, focusing on the production of high quality reads and OTUs that provide an accurate estimation of microbial diversity. It builds upon the known experimental and technological sources causing the occurrence of spurious OTUs and the existing algorithms developed to address this issue. The algorithms presented within this thesis significantly contribute to a more accurate prediction of the exact microbial composition within real-life samples, yet we still have a long way to go.

Regarding the chimera detection problem, it is predominantly challenging as chimeric reads, unlike sequencing errors, exist in high abundances and can be mistaken as novel organisms (Huber et al. 2004). Previously this problem was not given the attention needed, yet in the last few years various tools have been developed tackling this problem, ending with the CATCh tool, developed in the context of this PhD thesis and introduced in Chapter 4. Through combining the strengths of previous chimera detection tools, CATCh was able to produce accurate and robust results particularly for challenging chimeras such as those with low parents divergence, short chimeric range, or/and a varying number of parents. However, there is still room for improvement, as the existing tools often fail to detect all chimeric reads and occasionally flag non-chimeric reads as chimeric ones which has resulted in a drop of the sensitivity and specificity values respectively, as reported in Table 4:3.

This observation was also confirmed when benchmarking the OCToPUS methodology in comparison with other state-of-the-art pipelines, where the number of OTUs for all implementations was a few folds higher than expected (see Figure 6:3) and the accompanying composition of those OTUs demonstrated that the chimera problem persisted in spite of the applied *pre-processing* steps (Figure 6:4). It is important to report that throughout the whole comparative analysis, the removal of singletons was omitted in order to allow an accurate measurement of various artefacts. Yet, in real-life scenarios singletons are often removed (which means the true number of OTUs is as such normally higher than reported in most studies). Additionally, in respect to amplicons other than the 16S rRNA gene, we noticed a remarkable drop in the sensitivity of the chimera detection algorithms – including CATCh, which is probably related to the hard-coded parameters within the chimera detection tools which are fine-tuned for this specific marker gene. These results suggest that further improvement is required to resolve these problems, which were and still are an obstacle for the proper assessment of microbial diversity.

Another common source of artefacts are sequencing errors, where each sequencing platform reveals a different level of susceptibility to the various types of errors. A very preliminary approach adopted by the scientific community was to apply a strict quality filtering which removes any reads exhibiting a low quality or trims the read to the position from where the quality dropped. Yet, this resulted in a major reduction in the number of reads and/or the read length, emphasising the necessity for a more sophisticated approach to maintain the high-throughput results via the usage of less stringent quality filtering criteria. This can be achieved via the usage of denoising tools, which segregate sequencing errors from actual biological variation. In this PhD

thesis, we demonstrated the use of artificial intelligence classifiers trained on the error profile of different sequencing platforms to allocate and correct the erroneous bases.

Two denoising algorithms are introduced within Chapter 2 and Chapter 3 namely NoDe and IPED specifically developed for 454 Pyrosequencing platforms (the most widely used platform in the past decade) and Illumina MiSeq platforms (the currently dominant sequencing technology) respectively. Although both algorithms outperformed the available alternatives in reducing the error rate and the number of spurious OTUs, further improvement is still required. It was not possible to neither eliminate all of the sequencing errors (see Table 2:1 and Table 3:2) nor the spurious OTUs (see Table 2:3 and Additional Information 3:8). It is important to emphasise that applying a more advanced OTU-filtering approach, similar to what has been proposed by Eren et al. (2013) could result in a major reduction in the number of spurious OTUs. This can be achieved via applying a multi-step OTU filtering approach, a) minimal number of reads classified into one OTU across all samples, b) a minimum percentage abundance of an OTUs within one sample, c) a minimum number of samples where an OTU is detected, d) the minimum count of the most abundant unique sequence in an OTU. These default values differ in respect to the data size and the number of samples. Nonetheless, up to now both chimeric and erroneous reads constitute a persistent obstacle for accurate assessment of microbial diversity analysis using 16S rRNA gene amplicon sequencing.

Apart from that, we tried to bring the composition of the OTUs closer to bacterial taxonomy via the proposed approach within Chapter 5, where it was pointed out that the various variable regions within the 16S rRNA gene have evolved with different rates in different taxonomic families, which significantly affected the OTU clustering results upon applying a static clustering cut-off. To circumvent this problem, a novel methodology was introduced that calculates a dynamic and evolutionary compliant

OTU clustering cut-off per taxonomic family for each constructed amplicon, leading to improved OTU clustering results. Combining this methodology, together with proper chimera detection and denoising algorithms would increase the accuracy of microbial diversity estimation. Yet, applying the proposed approach could also result in an over-splitting of reads originated from the same species into various OTUs. It goes without saying that the main reason integrating a clustering step in most pipelines is to circumvent the potential negative impact of sequencing errors that exist within the reads. Yet, when the applied clustering cut-off as proposed in our methodology is excessively strict, certain reads (originated from the same species) with unresolved sequencing errors would be split into more than one OTU. Additionally, as 16S rRNA gene could exist in various non-identical copies within the genome, the same problem might occur whenever the difference between the various 16S rRNA paralogs exceeds (probably in combination with sequencing errors) the applied clustering cut-off.

These tools were applied on real biological data from a wide range of environments such as mice gut, deep geological layers, Antarctic, drinking water, nuclear reactor cooling water, fermentation bioreactors and plant microbiomes, as a part of the collaboration established during this PhD period. This collaborative work was crystalized into various research outputs summarized in APPENDIX B, thereby highlighting the contribution made within the context of this PhD. The complete list of publications is described in APPENDIX C.

Although the tools presented in this thesis were found efficient in resolving OTUs anomalies, we believe that they will not reach their maximum usability unless they are picked by integrated pipelines (such as mothur, QIIME or USEARCH). Most researchers within this field favour the use of one-stop software to handle their sequencing data rather than manually assembling the different algorithms. This was the

initial motivation to wrap the tools presented in this work into one pipeline, namely OCToPUS, introduced in Chapter 6. It was further extended to incorporate others tools from the scientific community, as such achieving the lowest error rate, the minimum number of spurious OTUs and the closest correspondence to the existing community without compromising the number of reads retained.

Another aspect that would impact the usability of software implementations is the computational cost. For instance, the CATCh algorithm requires the other chimera detection tools to be run first, after which all outputs are combined into one prediction using an artificial intelligence classifier. Similarly, the application of IPED algorithm requires additional time, due to the use of a sophisticated machine learning classifier. Thus we attempted to reduce the computational cost via incorporating them into OCToPUS pipeline in a time efficient manner (via a smooth integration of the various tools). The usability of the NoDe algorithm on the other hand is compromised due to the phasing out of the Roche 454 pyrosequencing.

It is important to emphasise that all conclusions were mainly derived from analyses performed on mock communities and only a few real biological data. Certainly this makes it a lot easier to compare different algorithms as within a mock community, unlike simulated data or a real biological data, the exact composition and abundances of the species are known. Consequently, it is possible to assess the error rate, chimerism and the accuracy of the produced OTUs (in terms of richness and evenness). However, the mock communities available only consist of a limited number of species (mostly around 20), thus it was not possible to mimic the real situation where high number of species exist exhibiting higher level of diversity. Additionally, as some of the mock samples were prepared after the DNA was successfully extracted and the 16S rRNA genes amplified and quantified for each species (considering the number of

paralogs), they did not account for the biases introduced within the DNA extraction or PCR amplification steps. To emphasize this point, we conducted an experiment where mock samples were prepared in different manners: a) DNA of the different species was added before PCR amplification, b) DNA of the different species was added after PCR amplification, or c) the species were added to the matrix (a sterile mock-up of the environmental sample) followed by DNA extraction and PCR amplification. Major variations were reported between the samples for instance, when adding mock community to the environmental matrix (clay), some species could not be detected after analysis due to difficulties when extracting DNA from such a matrix. This suggests that the bias introduced by PCR amplification and DNA extraction have major impact on the reproducibility and reliability of this analysis.

Performing a fair comparison between the different single-step tools implemented in different pipelines is not an easy task, as single-step algorithms, such as chimera detection and denoising algorithms, are normally applied within a context of a general pipeline involving various steps (paired-end assembly, quality filtering, clustering, etc). For this reason, we considered the widely used mothur pipeline as the default (www.mothur.org/wiki/454_SOP and www.mothur.org/wiki/MiSeq_SOP). As not every tool was included in mothur, other software such as QIIME and USEARCH were used which resulted in a small variations in the input data. This was observed in the denoising algorithms for the 454 Pyrosequencing platform, where mothur-based tools (NoDe, Pre-cluster, and AmpliconNoise) were compared to QIIME-based tools (Acacia and Denoiser). Similarly for Illumina MiSeq denoising algorithms, USEARCH-based tool (UNOISE) was compared to other mothur-based ones (IPED and Pre-cluster). Although these differences cannot be avoided, it might affect the accuracy of the comparison.

Additionally in order to conduct a more accurate comparative analysis and more closely mimic real-life scenarios, it is strongly advised to use a more diverse mock community. In fact the scientific community is missing such sequencing data derived from a mock community consisting of hundred(s) of species at different taxonomic distances. As artefacts within data analysis could result in over-merging (merging reads originated from two species into one OTU) and over-splitting (splitting of reads originated from one species into different OTUs), applying such mock community would certainly help assessing the occurrences of both phenomena. It might also be interesting to test other phylogenetic marker genes used for amplicon sequencing in the comparative analyses performed. In principle all of the tools presented in this work can work with any amplicon sequencing data, yet 16S rRNA amplicon sequencing data was mainly used to conduct the comparative analysis. As such, further investigation is required to assess the robustness of the conclusions presented here throughout the various amplicon sequencing data.

The development of various methodologies for enhancing the accuracy of 16S rRNA amplicon data analysis constitutes the main contribution of this thesis, pushing the analysis results towards a higher level of quality. As a result, the analyses performed within the context of this PhD allow providing guidelines for an optimal balance between the length and quality of the reads versus the total number of reads. From my point of view, I consider that it is crucially important to end up with reads having an error rate lower than 0.1% before the OTU clustering step, in order to obtain high quality OTUs. Allowing for a higher error rate was shown to result in an inflation of the number of OTUs. On the contrary increasing the stringency of the quality filtering steps in order to reduce this error rate further would jeopardize the read length or/and the number of reads retained. For this purpose, including a mock sample within every

sequencing run, preferably in three technical replicates, is of utter importance to assess the error rate for either the raw data, as a quality control step, or the pre-processed data to fine-tune the quality filtering approaches applied.

It has always been frustrating for microbiologists utilizing this type of analysis not to be able attach a species name to the OTUs detected within their samples. Thus secondly, I generally believe that the read length is more important than the number of reads, as it allows for a deeper taxonomic classification. Needless to say that this is highly dependent on the biological question at hand, where for instance monitoring the microbial shifts, regardless of which exact species is responsible for it, might already be sufficient. To properly cluster the sequencing reads into OTUs they should have similar lengths, thus a balance between the number of reads removed and the extent of length trimming should be achieved for platforms producing sequencing fragments with variable read lengths, e.g. 454 Pyrosequencing, Ion torrent and PacBio. In order to achieve such a balance we should take into consideration the influence of applying different length trimming stringency on the error rate versus the depth of the taxonomic classification.

The analysis presented in Chapter 5 can provide a good guidance for the level of discriminatory power gained with the inclusion of the various regions within the 16S rRNA gene. This is particularly helpful in case that improvements in sequencing technology lead to longer sequencing fragments, as such allowing the sequencing of one or more additional hypervariable regions. Additionally, it can be used to identify the regions that should be amplified in order to guarantee an accurate approximation of the species level by the resulting OTUs. Indeed, in case a large study needs to be performed, a small pilot study can give a rough idea on the taxonomic families dominating the environment of interest. Based on those preliminary results, the optimal

combination of variable regions can be identified for the bulk of the samples, as such leading to an optimal taxonomic resolution of the OTU clustering results.

Concerning the selection of the appropriate sequencing platforms, they differ in respect to the read quality and read length, and number of reads (see section 1.2 in the introduction). 454 Pyrosequencing, in my perspective, is the best performing platform for 16S rRNA amplicon sequencing, as it produces long reads (~700 bp), and the sequencing fragments show a low error rate (<0.1%) whilst having acceptable throughput after pre-processing. However, due to the announced shutdown of 454 Pyrosequencing mid-2016 and the advancements in Illumina sequencing platforms, Illumina MiSeq technology took the lead for 16S rRNA amplicon sequencing approaches, currently providing 300 bp paired-end reads (V3 chemistry) with an error rate lower than 0.1%, and with a much higher throughput than 454 Pyrosequencing.

The total amplicon length can be enhanced via partially overlapping forward and reverse reads, yet this can come at a cost of increasing the error rate to percentages higher than 0.1%, and consequently an inflation of the number of OTUs. An intermediate compromise is allowing around 150 bp overlap between both reads, allowing the amplification of both the V3 and V4 region of the 16S rRNA gene, with an error rate of around 0.1% after pre-processing. Ion torrent on the other hand, although providing very high throughput, requires extra attention to handle the relatively high error rate (>0.1%) and the read length (200-400 bp). Nonetheless, better alternatives seem to be available within the coming few years. With the modified PacBio platforms and the promising NanoPore platforms, it would be possible to have read lengths covering the whole 16S rRNA gene, which might outperform the Illumina MiSeq platform given that the throughput and quality will increase and costs will lower.

Several factors influence the resemblance between the relative abundances of the OTUs and the actual community composition. First, the 16S rRNA gene may occur in multiple copies (paralogs) within the same genome (Hashimoto et al. 2003). Their count -ranging from 1 to 15 copies per genome- compromises the reliability of the calculated relative OTU abundances. Additionally, these paralogs have a significant degree of sequence divergence, which might result in an overestimation of the number of species as reported in *Clostridium paradoxum* (Rainey et al. 1996) and *Paenibacillus polymyxa* (Nübel et al. 1996). To circumvent this obstacle, OTU abundances normalization should be performed taking into account the number of paralogs (and their sequence variation), using databases such as https://rrndb.umms.med.umich.edu/.

Another factor is the DNA extraction efficiency, where various studies demonstrated significant differences in the DNA yield and bacterial DNA composition via the application of various extraction kits on the same sample (Kennedy et al. 2014; Fouhy et al. 2016). Additionally, when applying different primers to target various regions of the 16S rRNA gene, major discrepancy between the microbial composition is reported, due to primer bias and difference in primer efficiency (Schloss et al. 2011; Kozich et al. 2013; Schloss et al. 2016; Kennedy et al. 2014; Fouhy et al. 2016). Certainly, the work presented in Chapter 5 introduces an approach to adjust the clustering cut-off in respect to the region within the 16S rRNA gene, yet, this cannot compensate for the differences in DNA yield caused by primer bias towards certain taxonomic clades.

Additionally, different library preparation methods have been applied to attach the barcode-sequence or multiplexing identifiers (MIDs) to the amplicon sequences, allowing the identification of reads from different samples, such as: fusion primer, adaptor ligase, and universal tailed-tag amplicon design. It was reported that the library preparation methods, the number of PCR cycles, and sequencing platform are the most

significant sources of bias and can have remarkable impact on the results (Ahn et al. 2012; Salipante et al. 2014; D'Amore et al. 2016; Schirmer et al. 2015). Consequently, these problems jeopardize the reliability of 16S rRNA metagenomics approach and suggest that the sequencing platform and the experimental design (DNA extraction, library preparation, primer selection and PCR-cycle) should be carefully considered, studied and unified among researchers addressing the same microbiome, to minimize those discrepancies and facilitate comparison between different experiments. Additionally, the use of a mock sample and small pilot studies will help assessing the different artefacts and experimental biases and guide towards the selection of the optimal experimental setting.

7.2. Perspectives

Within the context of this PhD, various obstacles in 16S rRNA amplicon sequencing data analysis were tackled, such as chimeric artefacts, sequencing errors and incompliance between the operational taxonomic units and the exact biological species present in the sample. As a result, various tools and algorithms were presented: CATCh, NoDe, IPED, DynamiC and OCToPUS focus on fine-tuning the analysis pipeline of 16S rRNA amplicon sequencing data. Although different improvements have been proposed throughout this PhD thesis to obtain a more accurate prediction of microbial diversity analysis, further optimization can still be achieved via investigating other perspectives. Various research topics are proposed below that could complement the methodologies and results presented in this PhD work and may further improve the diversity analysis results.

Secondary structure based alignment and OTU accuracy

A point of dispute in the OTU clustering step is the necessity for deriving the distances between the reads from a multiple sequence alignment or from pairwise alignments between the reads. Analysis pipelines such as QIIME (Caporaso et al. 2010) and USEARCH (Edgar 2010) propose calculating the distances between the reads via pairwise alignments, yet pipelines such as mothur (Schloss et al. 2009) promote the use of guided multiple sequence alignment partially due to the incorporation of the secondary structure information within the reference database.

The concept of incorporating the secondary structure information appears logically acceptable, as taking into account the conserved and variable regions within the 16S rRNA gene structure could guide the algorithm to a more accurate alignment, thus more accurate distance calculations. Nonetheless, Wang el at (2012) do not favour the added computational burden accompanying such analysis, which was later refuted by Schloss's work demonstrating the need for secondary structure information (Schloss 2013). Addressing this point could be of utter importance especially with the longer read lengths achieved by third-generation sequencing technologies like PacBio and NanoPore. Although not included within this thesis work, two master students (Mona Shahin and Akshaya Ramakrishnan) supervised by Mohamed Mysara Ahmed were studying such matter. A comparative study across various alignment and distance calculation algorithms was conducted, where primary results suggest that multiple sequence alignment driven distances, particularly those incorporating the secondary structure, are more accurate than pairwise alignment approaches. Additionally, disregarding the computation burden, the *de novo* multiple sequence aligners are superior to reference based aligners (such as the one used in mothur). Nonetheless, a more extensive analysis is required to help optimize the analysis guidelines.

Alternatives to distance based OTU clustering

Probably the main drawback of OTU clustering approach is that the identity based cut-off used to approximate existing taxa is arbitrary. Although OTUs were expected to comply with the concept of bacterial speciation (Gevers et al. 2005), the commonly applied 97% cut-off is expected to underestimate the total existing diversity (Koeppel & Wu 2013), and it is incapable of capturing the different lineages of evolution, as it is based on the naïve assumption that 16S rRNA genes evolve at the same rate regardless of their taxonomic classification. This particular problem is thoroughly discussed in Chapter 5, and a novel approach is proposed providing closer correspondence to the species boundaries.

Apart from the OTUs-based approach, others propose using alternative measures to resolve this matter. As a result, several evolutionary-compliant diversity units incorporating phylogenetic information have been proposed, such as Yule-coalescent (Pons et al. 2006; Powell et al. 2011; Barraclough et al. 2009; Powell 2012), Ecotype Simulation (Koeppel et al. 2008) and AdaptML (Hunt et al. 2008). Nonetheless, conflicts between microbial diversity and ecotypes has been reported, suggesting that extra improvement is required for these approaches (Doolittle 2012; Achtman & Wagner 2008; Schmidt et al. 2014). A more recent approach for diversity estimation has been proposed by Murat Eren et al. (2013) termed oligotyping. Instead of calculating the distances between all positions of the reads and applying a specific clustering cut-off, oligotyping uses only the most discriminating positions revealed by entropy analysis. Initially it was applied as a second layer refinement of the OTU clusters, resulting in refined taxonomic units (referred to as oligotypes). Yet, other approaches utilizing this concept have emerged that do not require previous clustering nor user supervision (as oligotyping) such as Minimum Entropy Decomposition (Eren et al. 2015) and DADA2 (Callahan et al. 2016). There is a need to compare these

emerging methodologies using a complex (challenging) mock community to assess their advantages and disadvantages compared to the traditional OTU approach.

Application of other marker genes

In general, the use of high throughput sequencing opens the possibility for any gene to be used for microbial diversity estimation. Although 16S rRNA gene was the most commonly used marker gene for Bacteria and Archaea, over the years researchers have attempted to identify and apply other marker genes to compensate for the limitations of the 16S rRNA gene, addressed in the Chapter 1.

For instance, the recombinase A gene involved in recombinational DNA repair and the RNA polymerase beta subunit gene involved in the initiation and elongation of transcription have been proposed for bacterial and archaeal diversity analysis (Wu et al. 2013). Similarly for fungal diversity analysis, large subunit (28S), SSU regions (18S rRNA) and most commonly the internal transcribed spacer (ITS) have been utilized as marker genes (Hodkinson & Grice 2015). The methods presented within this thesis focus on the bacterial 16S rRNA gene, yet it can be optimized for other phylogenetic markers. With the promise of increased read length, offered by the third generations sequencing platforms, extra analyses should be performed to evaluate the accuracy of the alternative marker genes in diversity estimation in comparison to the traditional small ribosomal subunit.

Shotgun sequencing methods

16S rRNA amplicon sequencing provides a reliable method for revealing the identity and the amount of species within the community. Albeit the recent efforts for predicting the functional profile of a microbial community relying on 16S rRNA data and reference genomes (Langille et al. 2013), it is unlikely to comprehend the level of information inferred from shotgun

sequencing methods. Shotgun metagenomic sequencing allows retrieving the taxonomic composition and the functional profile of the whole microbial community. Moreover, as it involves little or no PCR amplification, more precise calculation of the relative abundances can be inferred. Additionally, metatranscriptomics provides the full range of actively transcribed genes within the microbial community. With the decrease in the sequencing cost and accuracy together with incremental increase in the reference genomes offered via whole genome sequencing these methods are becoming more commonly applied. This emerging field is the future of microbial analysis, yet it poses huge challenges in term of data analysis which necessitates the development and the optimization of various bioinformatics solutions.

BIBLIOGRAPHY

Achtman, M. & Wagner, M., 2008. Microbial diversity and the genetic nature of microbial species. *Nature reviews. Microbiology*, 6(6), pp.431–40.

Ahn, J.-H. et al., 2012. Effects of PCR cycle number and DNA polymerase type on the 16S rRNA gene pyrosequencing analysis of bacterial communities. *Journal of microbiology* (Seoul, Korea), 50(6), pp.1071–4..

Allhoff, M. et al., 2013. Discovering motifs that induce sequencing errors. *BMC bioinformatics*, 14 Suppl 5, p.S1.

Alonso-Sáez, L. & Gasol, J.M., 2007. Seasonal variations in the contributions of different bacterial groups to the uptake of low-molecular-weight compounds in northwestern Mediterranean coastal waters. *Applied and environmental microbiology*, 73(11), pp.3528–35.

Altschul, S.F. et al., 1990. Basic local alignment search tool. *Journal of molecular biology*, 215(3), pp.403–10.

Amann, R.I., Ludwig, W. & Schleifer, K.H., 1995. Phylogenetic identification and in situ detection of individual microbial cells without cultivation. *Microbiological reviews*, 59(1), pp.143–69.

Ash, C. et al., 1991. Comparative analysis of Bacillus anthracis, Bacillus cereus, and related species on the basis of reverse transcriptase sequencing of 16S rRNA. *International journal of systematic bacteriology*, 41(3), pp.343–6.

Ashelford, K.E. et al., 2005. At least 1 in 20 16S rRNA sequence records currently held in public repositories is estimated to contain substantial anomalies. *Applied and environmental microbiology*, 71(12), pp.7724–36.

Au, K.F. et al., 2012. Improving PacBio long read accuracy by short read alignment. *PloS one*, 7(10), p.e46679.

Auch, A.F. et al., 2010. Digital DNA-DNA hybridization for microbial species delineation by means of genome-to-genome sequence comparison. *Standards in genomic sciences*, 2(1), pp.117–34..

Babbage, C., 1994. *Passages from the Life of a Philosopher* M. Campbell-Kelly, ed., New Brunswick, NJ, USA: Rutgers University Press.

Balzer, S., Malde, K. & Jonassen, I., 2011. Systematic exploration of error sources in pyrosequencing flowgram data. *Bioinformatics (Oxford, England)*, 27(13), pp.i304–9.

Bankevich, A. et al., 2012. SPAdes: a new genome assembly algorithm and its applications to single-cell sequencing. *Journal of computational biology : a journal of computational molecular cell biology*, 19(5), pp.455–77.

Barns, S.M. et al., 2007. Acidobacteria phylum sequences in uranium-contaminated subsurface sediments greatly expand the known diversity within the phylum. *Applied and environmental microbiology*, 73(9), pp.3113–6.

Barraclough, T.G. et al., 2009. Inferring evolutionarily significant units of bacterial diversity from broad environmental surveys of single-locus data. *Biology Letters*, 5(3), pp.425–8.

Bartlett, J.M.S. & Stirling, D., 2003. A short history of the polymerase chain reaction. *Methods in molecular biology (Clifton, N.J.)*, 226, pp.3–6.

Benítez-Páez, A., Portune, K.J. & Sanz, Y., 2016. Species-level resolution of 16S rRNA gene amplicons

sequenced through the MinION™ portable nanopore sequencer. *GigaScience*, 5(1), p.4.

Bentley, D.R. et al., 2008. Accurate whole human genome sequencing using reversible terminator chemistry. *Nature*, 456(7218), pp.53–9.

Bokulich, N.A. et al., 2013. Quality-filtering vastly improves diversity estimates from Illumina amplicon sequencing. *Nature methods*, 10(1), pp.57–9.

Boone, D.R., Castenholz, R.W. & Garrity, G.M. eds., 2001. *Bergey's Manual® of Systematic Bacteriology*, New York, NY: Springer New York.

Borg, I. & Groenen, P., 2005. *Modern Multidimensional Scaling: theory and applications* 2nd ed.,

Böttger, E.C., 1989. Rapid determination of bacterial ribosomal RNA sequences by direct sequencing of enzymatically amplified DNA. *FEMS microbiology letters*, 53(1-2), pp.171–6.

Bottos, E.M. et al., 2008. Prokaryotic diversity of arctic ice shelf microbial mats. *Environmental microbiology*, 10(4), pp.950–66.

Bragg, L. et al., 2012. Fast, accurate error-correction of amplicon pyrosequences using Acacia. *Nature methods*, 9(5), pp.425–6.

Branin, J.J. & Case, M., 1998. Reforming Scholarly Publishing in the Sciences: A Librarian Perspective.

Brenner, D.J., Staley, J.T. & Krieg, N.R., 2001. Classification of Procaryotic Organisms and the Concept of Bacterial Speciation. In D. R. Boone, R. W. Castenholz, & G. M. Garrity, eds. *Bergey's Manual of Systematic Bacteriology Volume Two: The Proteobacteria, Part A Introductory Essays*. Springer New York, pp. 27–38.

Brodie, E.L. et al., 2007. Urban aerosols harbor diverse and dynamic bacterial populations. *Proceedings of the National Academy of Sciences of the United States of America*, 104(1), pp.299–304.

Cai, Y. & Sun, Y., 2011. ESPRIT-Tree: hierarchical clustering analysis of millions of 16S rRNA pyrosequences in quasilinear computational time. *Nucleic acids research*, 39(14), p.e95.

Callahan, B.J. et al., 2016. dada2: high-resolution sample inference from illumina amplicon data. *Nature Methods*.

Campbell-Platt, G., 1994. Fermented foods — a world perspective. *Food Research International*, 27(3), pp.253–257.

Caporaso, J.G. et al., 2010. QIIME allows analysis of high-throughput community sequencing data. *Nature methods*, 7(5), pp.335–6.

Chakravorty, S. et al., 2007. A detailed analysis of 16S ribosomal RNA gene segments for the diagnosis of pathogenic bacteria. *Journal of microbiological methods*, 69(2), pp.330–9.

Chen, L. et al., 2014. Rapid Sanger sequencing of the 16S rRNA gene for identification of some common pathogens. *PloS one*, 9(2), p.e88886.

Chen, W. et al., 2013. A comparison of methods for clustering 16S rRNA sequences into OTUs. *PloS one*, 8(8), p.e70837.

Clarridge, J.E., 2004. Impact of 16S rRNA gene sequence analysis for identification of bacteria on clinical microbiology and infectious diseases. *Clinical microbiology reviews*, 17(4), pp.840–62, table of contents.

Cline, J., Braman, J.C. & Hogrefe, H.H., 1996. PCR fidelity of pfu DNA polymerase and other thermostable DNA polymerases. *Nucleic acids research*, 24(18), pp.3546–51.

Cohan, F.M. & Perry, E.B., 2007. A systematics for discovering the fundamental units of bacterial diversity. *Current biology : CB*, 17(10), pp.R373–86.

Cole, J.R. et al., 2014. Ribosomal Database Project: data and tools for high throughput rRNA analysis. *Nucleic acids research*, 42(Database issue), pp.D633–42.

Cole, J.R. et al., 2007. The ribosomal database project (RDP-II): introducing myRDP space and quality controlled public data. *Nucleic acids research*, 35(Database issue), pp.D169–72.

Connor, N. et al., 2010. Ecology of speciation in the genus Bacillus. *Applied and environmental microbiology*, 76(5), pp.1349–58.

Cook, V.J. et al., 2003. Conventional methods versus 16S ribosomal DNA sequencing for identification of nontuberculous mycobacteria: cost analysis. *Journal of clinical microbiology*, 41(3), pp.1010–5.

Curtis, T.P., Sloan, W.T. & Scannell, J.W., 2002. Estimating prokaryotic diversity and its limits. *Proceedings of the National Academy of Sciences of the United States of America*, 99(16), pp.10494–9.

D'Amore, R. et al., 2016. A comprehensive benchmarking study of protocols and sequencing platforms for 16S rRNA community profiling. *BMC Genomics*, 17(1), p.55.

D'Argenio, V. et al., 2014. Comparative metagenomic analysis of human gut microbiome composition using two different bioinformatic pipelines. *BioMed research international*, 2014, p.325340.

Delhalle, L. et al., 2016. Exploring the Bacterial Diversity of Belgian Steak Tartare Using Metagenetics and Quantitative Real-Time PCR Analysis. *Journal of food protection*, 79(2), pp.220–9.

Demain, A.L., 2000. Microbial biotechnology. *Trends in biotechnology*, 18(1), pp.26–31.

DeSantis, T.Z., Hugenholtz, P., Larsen, N., et al., 2006. Greengenes, a chimera-checked 16S rRNA gene database and workbench compatible with ARB. *Applied and environmental microbiology*, 72(7), pp.5069–72.

DeSantis, T.Z., Hugenholtz, P., Keller, K., et al., 2006. NAST: a multiple sequence alignment server for comparative analysis of 16S rRNA genes. *Nucleic acids research*, 34(Web Server issue), pp.W394–9.

Doolittle, W.F., 2012. Population genomics: how bacterial species form and why they don't exist. *Current biology : CB*, 22(11), pp.R451–3.

Doolittle, W.F. & Zhaxybayeva, O., 2009. On the origin of prokaryotic species. *Genome research*, 19(5), pp.744–56.

Dostie, J. et al., 2006. Chromosome Conformation Capture Carbon Copy (5C): a massively parallel solution for mapping interactions between genomic elements. *Genome research*, 16(10), pp.1299–309.

Droege, M. & Hill, B., 2008. The Genome Sequencer FLX System--longer reads, more applications, straight forward bioinformatics and more complete data sets. *Journal of biotechnology*, 136(1-2), pp.3–10.

Edgar, R.C., 2010. Search and clustering orders of magnitude faster than BLAST. *Bioinformatics (Oxford, England)*, 26(19), pp.2460–1.

Edgar, R.C. et al., 2011. UCHIME improves sensitivity and speed of chimera detection. *Bioinformatics (Oxford, England)*, 27(16), pp.2194–200.

Edgar, R.C., 2013. UPARSE: highly accurate OTU sequences from microbial amplicon reads. *Nature methods*, 10(10), pp.996–8.

Edgar, R.C. & Flyvbjerg, H., 2015. Error filtering, pair assembly and error correction for next-generation sequencing reads. *Bioinformatics (Oxford, England)*, 31(21), pp.3476–82.

Edwards, R.A. et al., 2006. Using pyrosequencing to shed light on deep mine microbial ecology. *BMC genomics*, 7, p.57.

Eichorst, S.A., Breznak, J.A. & Schmidt, T.M., 2007. Isolation and characterization of soil bacteria that define Terriglobus gen. nov., in the phylum Acidobacteria. *Applied and environmental microbiology*, 73(8), pp.2708–17.

Eren, A.M. et al., 2013. A filtering method to generate high quality short reads using illumina paired-end technology. *PloS one*, 8(6), p.e66643.

Eren, A.M. et al., 2015. Minimum entropy decomposition: unsupervised oligotyping for sensitive partitioning of high-throughput marker gene sequences. *Isme J*, 9(4), pp.968–979.

Eren, A.M. et al., 2013. Oligotyping: differentiating between closely related microbial taxa using 16S rRNA gene data R. Freckleton, ed. *Methods in Ecology and Evolution*, 4(12), pp.1111–1119.

Ewing, B. et al., 1998. Base-calling of automated sequencer traces using phred. I. Accuracy assessment. *Genome research*, 8(3), pp.175–85.

Excoffier, L., Smouse, P.E. & Quattro, J.M., 1992. Analysis of molecular variance inferred from metric distances among DNA haplotypes: application to human mitochondrial DNA restriction data. *Genetics*, 131(2), pp.479–91.

Fawcett, T.N., Holloway, A. & Rhynas, S., 2015. If I have seen further it is by standing on the shoulders of giants: Finding a voice, a positive future for nursing. *Journal of Advanced Nursing*, 71(6), pp.1195–1197.

Feng, Q., 2015. *Metagenomics Study on the Polymorphism of Gut Microbiota and Their Function on Human Health*. University of Copenhagen.

Fichot, E.B. & Norman, R.S., 2013. Microbial phylogenetic profiling with the Pacific Biosciences sequencing platform. *Microbiome*, 1(1), p.10.

Fierer, N. et al., 2007. Metagenomic and small-subunit rRNA analyses reveal the genetic diversity of bacteria, archaea, fungi, and viruses in soil. *Applied and environmental microbiology*, 73(21), pp.7059–66.

Fierer, N. & Jackson, R.B., 2006. The diversity and biogeography of soil bacterial communities. *Proceedings of the National Academy of Sciences of the United States of America*, 103(3), pp.626–31.

Fitch, W.M. & Margoliash, E., 1967. Construction of phylogenetic trees. *Science (New York, N.Y.)*, 155(3760), pp.279–84.

Fonseca, V.G. et al., 2012. Sample richness and genetic diversity as drivers of chimera formation in nSSU metagenetic analyses. *Nucleic Acids Res*, 40(9), p.e66.

Fosso, B. et al., 2015. BioMaS: a modular pipeline for Bioinformatic analysis of Metagenomic AmpliconS. *BMC bioinformatics*, 16, p.203.

Fouhy, F. et al., 2016. 16S rRNA gene sequencing of mock microbial populations- impact of DNA extraction method, primer choice and sequencing platform. *BMC Microbiology*, 16(1), p.123.

Fox, G.E., Wisotzkey, J.D. & Jurtshuk, P., 1992. How close is close: 16S rRNA sequence identity may not be sufficient to guarantee species identity. *International journal of systematic bacteriology*, 42(1), pp.166–70.

Frias-Lopez, J. et al., 2008. Microbial community gene expression in ocean surface waters. *Proceedings of the National Academy of Sciences of the United States of America*, 105(10), pp.3805–10.

Fu, L. et al., 2012. CD-HIT: accelerated for clustering the next-generation sequencing data.

Bioinformatics (Oxford, England), 28(23), pp.3150–2.

Gaspar, J.M. & Thomas, W.K., 2013. Assessing the consequences of denoising marker-based metagenomic data. *PloS one*, 8(3), p.e60458.

Gerlach, W., 2012. *Taxonomic classification of metagenomic sequences*. Bielefeld University.

Gevers, D. et al., 2005. Opinion: Re-evaluating prokaryotic species. *Nature reviews. Microbiology*, 3(9), pp.733–9.

Ghodsi, M., Liu, B. & Pop, M., 2011. DNACLUST: accurate and efficient clustering of phylogenetic marker genes. *BMC bioinformatics*, 12, p.271.

Ghyselinck, J. et al., 2013. The effect of primer choice and short read sequences on the outcome of 16S rRNA gene based diversity studies. *PloS one*, 8(8), p.e71360.

Gilles, A. et al., 2011. Accuracy and quality assessment of 454 GS-FLX Titanium pyrosequencing. *BMC genomics*, 12(1), p.245.

Giovannoni, S.J. & Stingl, U., 2005. Molecular diversity and ecology of microbial plankton. *Nature*, 437(7057), pp.343–8.

Gloor, G.B. et al., 2010. Microbiome profiling by illumina sequencing of combinatorial sequence-tagged PCR products. *PloS one*, 5(10), p.e15406.

González-Aguilar, G.A. et al., 2009. Preserving quality of fresh-cut products using safe technologies. *Journal für Verbraucherschutz und Lebensmittelsicherheit*, 5(1), pp.65–72.

Greisen, K. et al., 1994. PCR primers and probes for the 16S rRNA gene of most species of pathogenic bacteria, including bacteria found in cerebrospinal fluid. *Journal of clinical microbiology*, 32(2), pp.335–51.

Gutell, R.R. et al., 1985. Comparative anatomy of 16-S-like ribosomal RNA. *Progress in nucleic acid research and molecular biology*, 32, pp.155–216.

Haas, B.J. et al., 2011. Chimeric 16S rRNA sequence formation and detection in Sanger and 454-pyrosequenced PCR amplicons. *Genome research*, 21(3), pp.494–504.

Hall, M. et al., 2009. The WEKA Data Mining Software : An Update. *SIGKDD Explorations*, 11(1), pp.10–18.

Hao, X., Jiang, R. & Chen, T., 2011. Clustering 16S rRNA for OTU prediction: a method of unsupervised Bayesian clustering. *Bioinformatics (Oxford, England)*, 27(5), pp.611–8.

Hashimoto, J.G., Stevenson, B.S. & Schmidt, T.M., 2003. Rates and consequences of recombination between rRNA operons. *Journal of bacteriology*, 185(3), pp.966–72.

Hildebrand, F. et al., 2014. LotuS: an efficient and user-friendly OTU processing pipeline. *Microbiome*, 2(1), p.30.

Hodkinson, B.P. & Grice, E.A., 2015. Next-Generation Sequencing: A Review of Technologies and Tools for Wound Microbiome Research. *Advances in wound care*, 4(1), pp.50–58.

Huber, J.A. et al., 2007. Microbial population structures in the deep marine biosphere. *Science (New York, N.Y.)*, 318(5847), pp.97–100.

Huber, T., Faulkner, G. & Hugenholtz, P., 2004. Bellerophon: a program to detect chimeric sequences in multiple sequence alignments. *Bioinformatics (Oxford, England)*, 20(14), pp.2317–9.

Hugenholtz, P., 2002. Exploring prokaryotic diversity in the genomic era. *Genome biology*, 3(2), p.REVIEWS0003.

Hunt, D.E. et al., 2008. Resource Partitioning and Sympatric Differentiation Among Closely Related Bacterioplankton. *Science*, 320(5879), pp.1081–1085.

Huse, S.M. et al., 2007. Accuracy and quality of massively parallel DNA pyrosequencing. *Genome biology*, 8(7), p.R143.

Huse, S.M. et al., 2010. Ironing out the wrinkles in the rare biosphere through improved OTU clustering. *Environmental Microbiology*, 12(7), pp.1889–1898.

Hutchinson, G.E., 1968. When are species necessary? In. In R. C. Lewontin, ed. *Population Biology and Evolution*. Syracuse, NY: Syracuse University Press, pp. 177–186.

Janda, J.M. & Abbott, S.L., 2007. 16S rRNA gene sequencing for bacterial identification in the diagnostic laboratory: pluses, perils, and pitfalls. *Journal of clinical microbiology*, 45(9), pp.2761–4.

Jeraldo, P. et al., 2014. IM-TORNADO: A Tool for Comparison of 16S Reads from Paired-End Libraries. *PLoS ONE*, 9(12), p.e114804.

Jeraldo, P., Chia, N. & Goldenfeld, N., 2011. On the suitability of short reads of 16S rRNA for phylogeny-based analyses in environmental surveys. *Environmental microbiology*, 13(11), pp.3000–9.

Johnson, R.M. & Cameron, R.E., 1973. The Physiology and Distribution of Bacteria in Hot and Cold Deserts. *Journal of the Arizona Academy of Science*.

Judo, M.S., Wedel, a B. & Wilson, C., 1998. Stimulation and suppression of PCR-mediated recombination. *Nucleic acids research*, 26(7), pp.1819–25.

Jünemann, S. et al., 2012. Bacterial community shift in treated periodontitis patients revealed by ion torrent 16S rRNA gene amplicon sequencing. *PloS one*, 7(8), p.e41606.

Kennedy, N.A. et al., 2014. The Impact of Different DNA Extraction Kits and Laboratories upon the Assessment of Human Gut Microbiota Composition by 16S rRNA Gene Sequencing Y. Sanz, ed. *PLoS ONE*, 9(2), p.e88982..

Kilianski, A. et al., 2015. Bacterial and viral identification and differentiation by amplicon sequencing on the MinION nanopore sequencer. *GigaScience*, 4, p.12.

Kim, M. et al., 2014. Towards a taxonomic coherence between average nucleotide identity and 16S rRNA gene sequence similarity for species demarcation of prokaryotes. *International journal of systematic and evolutionary microbiology*, 64(Pt 2), pp.346–51.

Kircher, M., Stenzel, U. & Kelso, J., 2009. Improved base calling for the Illumina Genome Analyzer using machine learning strategies. *Genome biology*, 10(8), p.R83.

Klindworth, A. et al., 2013. Evaluation of general 16S ribosomal RNA gene PCR primers for classical and next-generation sequencing-based diversity studies. *Nucleic acids research*, 41(1), p.e1.

Koeppel, A. et al., 2008. Identifying the fundamental units of bacterial diversity: a paradigm shift to incorporate ecology into bacterial systematics. *Proceedings of the National Academy of Sciences of the United States of America*, 105(7), pp.2504–9.

Koeppel, A.F. & Wu, M., 2013. Surprisingly extensive mixed phylogenetic and ecological signals among bacterial Operational Taxonomic Units. *Nucleic Acids Research*, 41(10), pp.5175–5188.

Konstantinidis, K.T., Ramette, A. & Tiedje, J.M., 2006. The bacterial species definition in the genomic era. *Philosophical transactions of the Royal Society of London. Series B, Biological sciences*, 361(1475), pp.1929–40.

Konstantinidis, K.T. & Tiedje, J.M., 2005. Towards a genome-based taxonomy for prokaryotes. *Journal*

of bacteriology, 187(18), pp.6258–64.

Kozich, J.J. et al., 2013. Development of a dual-index sequencing strategy and curation pipeline for analyzing amplicon sequence data on the MiSeq Illumina sequencing platform. *Applied and environmental microbiology*, 79(17), pp.5112–20.

Kuczynski, J. et al., 2011. Using QIIME to analyze 16S rRNA gene sequences from microbial communities. *Current protocols in bioinformatics / editoral board, Andreas D. Baxevanis ... [et al.]*, Chapter 10, p.Unit 10.7.

Kumar, P.S. et al., 2011. Target region selection is a critical determinant of community fingerprints generated by 16S pyrosequencing. *PloS one*, 6(6), p.e20956.

Kunin, V. et al., 2010. Wrinkles in the rare biosphere: pyrosequencing errors can lead to artificial inflation of diversity estimates. *Environmental microbiology*, 12(1), pp.118–23.

Lahr, D.J.G. & Katz, L. a, 2009. Reducing the impact of PCR-mediated recombination in molecular evolution and environmental studies using a new-generation high-fidelity DNA polymerase. *BioTechniques*, 47(4), pp.857–66.

Langille, M.G. et al., 2013. Predictive functional profiling of microbial communities using 16S rRNA marker gene sequences. *Nat Biotechnol*, 31(9), pp.814–821.

Lapierre, P. & Gogarten, J.P., 2009. Estimating the size of the bacterial pan-genome. *Trends in genetics : TIG*, 25(3), pp.107–10.

Leblanc, J.G. et al., 2011. B-Group vitamin production by lactic acid bacteria - current knowledge and potential applications. *Journal of Applied Microbiology*, 111(6), pp.1297–1309.

Levene, M.J. et al., 2003. Zero-mode waveguides for single-molecule analysis at high concentrations. *Science (New York, N.Y.)*, 299(5607), pp.682–6.

Liu, B. et al., 2012. COPE: an accurate k-mer-based pair-end reads connection tool to facilitate genome assembly. *Bioinformatics (Oxford, England)*, 28(22), pp.2870–4.

Liu, Z. et al., 2008. Accurate taxonomy assignments from 16S rRNA sequences produced by highly parallel pyrosequencers. *Nucleic acids research*, 36(18), p.e120.

Logares, R. et al., 2012. Environmental microbiology through the lens of high-throughput DNA sequencing: Synopsis of current platforms and bioinformatics approaches. *Journal of microbiological methods*, 91(1), pp.106–13.

Loman, N.J. et al., 2012. Performance comparison of benchtop high-throughput sequencing platforms. *Nature biotechnology*, 30(5), pp.434–9.

Lozupone, C. & Knight, R., 2005. UniFrac: a new phylogenetic method for comparing microbial communities. *Applied and environmental microbiology*, 71(12), pp.8228–35.

Ludwig, W. et al., 1998. Bacterial phylogeny based on comparative sequence analysis. *Electrophoresis*, 19(4), pp.554–68.

Ludwig, W. & Klenk, H.-P., 2001. Overview: A phylogenetic backbone and taxonomic framework for procaryotic systematics. In Boone D. R., R. W. Castenholz, & G. M. Garrity, eds. *Bergey's Manual of Systematic Bacteriology*. Springer New York, pp. 49–65.

Ludwig, W. & Schleifer, K.H., 1994. Bacterial phylogeny based on 16S and 23S rRNA sequence analysis. *FEMS microbiology reviews*, 15(2-3), pp.155–73.

Magoč, T. & Salzberg, S.L., 2011. FLASH: fast length adjustment of short reads to improve genome assemblies. *Bioinformatics (Oxford, England)*, 27(21), pp.2957–63.

Magurran, A.E., 2004. *Wiley: Measuring Biological Diversity - Anne E. Magurran*, Wiley-Blackwell.

Mahé, F. et al., 2014. Swarm: robust and fast clustering method for amplicon-based studies. *PeerJ*, 2, p.e593.

Margulies, M. et al., 2005. Genome sequencing in microfabricated high-density picolitre reactors. *Nature*, 437(7057), pp.376–80.

Marshall, C.W. et al., 2012. Electrosynthesis of commodity chemicals by an autotrophic microbial community. *Appl Environ Microbiol*, 78(23), pp.8412–8420.

Masella, A.P. et al., 2012. PANDAseq: paired-end assembler for illumina sequences. *BMC bioinformatics*, 13, p.31.

Matias Rodrigues, J.F. & von Mering, C., 2014. HPC-CLUST: distributed hierarchical clustering for large sets of nucleotide sequences. *Bioinformatics (Oxford, England)*, 30(2), pp.287–8.

Matthews, B.W., 1975. Comparison of the predicted and observed secondary structure of T4 phage lysozyme. *Biochimica et biophysica acta*, 405(2), pp.442–51.

Maxam, A.M. & Gilbert, W., 1977. A new method for sequencing DNA. *Proceedings of the National Academy of Sciences of the United States of America*, 74(2), pp.560–4.

Mayr, E., 1982. *The growth of biological thought: Diversity, evolution, and inheritance.*, Harvard University Press.

McDonald, D. et al., 2012. An improved Greengenes taxonomy with explicit ranks for ecological and evolutionary analyses of bacteria and archaea. *The ISME journal*, 6(3), pp.610–8.

Medini, D. et al., 2008. Microbiology in the post-genomic era. *Nature reviews. Microbiology*, 6(6), pp.419–30.

Medvedev, P. et al., 2011. Error correction of high-throughput sequencing datasets with non-uniform coverage. *Bioinformatics (Oxford, England)*, 27(13), pp.i137–41.

Meier-Kolthoff, J.P. et al., 2013. Genome sequence-based species delimitation with confidence intervals and improved distance functions. *BMC bioinformatics*, 14, p.60.

Meyer, F. et al., 2008. The metagenomics RAST server - a public resource for the automatic phylogenetic and functional analysis of metagenomes. *BMC bioinformatics*, 9, p.386.

Milani, C. et al., 2013. Assessing the fecal microbiota: an optimized ion torrent 16S rRNA gene-based analysis protocol. *PloS one*, 8(7), p.e68739.

Morgan, M.J. et al., 2013. Improved inference of taxonomic richness from environmental DNA. *PLoS One*, 8(8), p.e71974.

Mosher, J.J. et al., 2013. Efficacy of a 3rd generation high-throughput sequencing platform for analyses of 16S rRNA genes from environmental samples. *Journal of microbiological methods*, 95(2), pp.175–81.

Mosher, J.J. et al., 2014. Improved performance of the PacBio SMRT technology for 16S rDNA sequencing. *Journal of microbiological methods*, 104, pp.59–60.

Mysara, M., Saeys, Y., et al., 2015. CATCh, an ensemble classifier for chimera detection in 16S rRNA sequencing studies. *Applied and environmental microbiology*, 81(5), pp.1573–84.

Mysara, M. et al., 2016. IPED: a highly efficient denoising tool for Illumina MiSeq Paired-end 16S rRNA gene amplicon sequencing data. *BMC Bioinformatics*, 17(1), p.192.

Mysara, M., Leys, N., et al., 2015. NoDe: a fast error-correction algorithm for pyrosequencing amplicon reads. *BMC Bioinformatics*, 16(1), p.88.

Nakamura, K. et al., 2011. Sequence-specific error profile of Illumina sequencers. *Nucleic acids research*, 39(13), p.e90.

Nawrocki, E., 2009. *Structural RNA Homology Search and Alignment Using Covariance Models*. Washington University in Saint Louis.

Nawrocki, E.P. & Eddy, S.R., 2013. Infernal 1.1: 100-fold faster RNA homology searches. *Bioinformatics (Oxford, England)*, 29(22), pp.2933–5.

Nelson, M.C. et al., 2014. Analysis, optimization and verification of Illumina-generated 16S rRNA gene amplicon surveys. *PloS one*, 9(4), p.e94249.

Newman, D.K. & Banfield, J.F., 2002. Geomicrobiology: how molecular-scale interactions underpin biogeochemical systems. *Science (New York, N.Y.)*, 296(5570), pp.1071–7.

Noller, H.F., 1984. Structure of ribosomal RNA. *Annual review of biochemistry*, 53, pp.119–62.

Nübel, U. et al., 1996. Sequence heterogeneities of genes encoding 16S rRNAs in Paenibacillus polymyxa detected by temperature gradient gel electrophoresis. *Journal of bacteriology*, 178(19), pp.5636–43.

Nunoura, T. et al., 2015. Hadal biosphere: insight into the microbial ecosystem in the deepest ocean on Earth. *Proceedings of the National Academy of Sciences of the United States of America*, 112(11), pp.E1230–6.

Odelberg, S.J. et al., 1995. Template-switching during DNA synthesis by Thermus aquaticus DNA polymerase I. *Nucleic acids research*, 23(11), pp.2049–57.

Olsen, G.J. & Woese, C.R., 1993. Ribosomal RNA: a key to phylogeny. *FASEB journal : official publication of the Federation of American Societies for Experimental Biology*, 7(1), pp.113–23.

Olsen, G.J., Woese, C.R. & Overbeek, R., 1994. The winds of (evolutionary) change: breathing new life into microbiology. *Journal of bacteriology*, 176(1), pp.1–6.

Pace, N.R., 1997. A molecular view of microbial diversity and the biosphere. *Science (New York, N.Y.)*, 276(5313), pp.734–40.

Patel, J.B., 2001. 16S rRNA gene sequencing for bacterial pathogen identification in the clinical laboratory. *Molecular diagnosis : a journal devoted to the understanding of human disease through the clinical application of molecular biology*, 6(4), pp.313–21.

Pedrós-Alió, C., 2006. Marine microbial diversity: can it be determined? *Trends in microbiology*, 14(6), pp.257–63.

Van de Peer, Y., Chapelle, S. & De Wachter, R., 1996. A quantitative map of nucleotide substitution rates in bacterial rRNA. *Nucleic acids research*, 24(17), pp.3381–91.

Pei, A.Y. et al., 2010. Diversity of 16S rRNA genes within individual prokaryotic genomes. *Applied and environmental microbiology*, 76(12), pp.3886–97.

Platt, J.C., 1998. Fast training of support vector machines using sequential minimal optimization. In B. Schölkopf, C. Burges, & A. Smola, eds. *Advances in Kernel Methods: Support Vector Learning*. Cambridge: The MIT Press, pp. 185–208.

Plummer, E. et al., 2015. A Comparison of Three Bioinformatics Pipelines for the Analysis of Preterm Gut Microbiota using 16S rRNA Gene Sequencing Data. *Journal of Proteomics & Bioinformatics*, 8(12).

Pons, J. et al., 2006. Sequence-Based Species Delimitation for the DNA Taxonomy of Undescribed Insects. *Systematic Biology*, 55(4), pp.595–609.

Porazinska, D.L. et al., 2012. The nature and frequency of chimeras in eukaryotic metagenetic samples. *Journal of nematology*, 44(1), pp.18–25.

Powell, J.R., 2012. Accounting for uncertainty in species delineation during the analysis of environmental DNA sequence data. *Methods in Ecology and Evolution*, 3(1), pp.1–11.

Powell, J.R. et al., 2011. Evolutionary criteria outperform operational approaches in producing ecologically relevant fungal species inventories. *Molecular Ecology*, 20(3), pp.655–666.

Pruesse, E. et al., 2007. SILVA: a comprehensive online resource for quality checked and aligned ribosomal RNA sequence data compatible with ARB. *Nucleic acids research*, 35(21), pp.7188–96.

Quast, C. et al., 2013. The SILVA ribosomal RNA gene database project: improved data processing and web-based tools. *Nucleic acids research*, 41(Database issue), pp.D590–6.

Quince, C. et al., 2009. Accurate determination of microbial diversity from 454 pyrosequencing data. *Nature methods*, 6(9), pp.639–41.

Quince, C. et al., 2011. Removing noise from pyrosequenced amplicons. *BMC bioinformatics*, 12, p.38.

Rådström, P. et al., 1994. Detection of bacterial DNA in cerebrospinal fluid by an assay for simultaneous detection of Neisseria meningitidis, Haemophilus influenzae, and streptococci using a seminested PCR strategy. *Journal of clinical microbiology*, 32(11), pp.2738–44.

Rainey, F.A. et al., 1996. Clostridium paradoxum DSM 7308T contains multiple 16S rRNA genes with heterogeneous intervening sequences. *Microbiology (Reading, England)*, 142 (Pt 8, pp.2087–95.

Reeder, J. & Knight, R., 2010. Rapidly denoising pyrosequencing amplicon reads by exploiting rank-abundance distributions. *Nature methods*, 7(9), pp.668–9.

Richter, M. & Rosselló-Móra, R., 2009. Shifting the genomic gold standard for the prokaryotic species definition. *Proceedings of the National Academy of Sciences of the United States of America*, 106(45), pp.19126–31.

Roesch, L.F.W. et al., 2007. Pyrosequencing enumerates and contrasts soil microbial diversity. *The ISME journal*, 1(4), pp.283–90.

Ronaghi, M., 1998. DNA SEQUENCING:A Sequencing Method Based on Real-Time Pyrophosphate. *Science*, 281(5375), pp.363–365.

Rosselló-Móra, R., 2012. Towards a taxonomy of Bacteria and Archaea based on interactive and cumulative data repositories. *Environmental microbiology*, 14(2), pp.318–34.

Rosselló-Mora, R. & Amann, R., 2001. The species concept for prokaryotes. *FEMS microbiology reviews*, 25(1), pp.39–67.

Rougemont, J. et al., 2008. Probabilistic base calling of Solexa sequencing data. *BMC bioinformatics*, 9, p.431.

Rusk, N., 2009. Cheap third-generation sequencing. *Nature Methods*, 6(4), pp.244–244.

Russell, D.J. et al., 2010. A grammar-based distance metric enables fast and accurate clustering of large sets of 16S sequences. *BMC bioinformatics*, 11, p.601.

Salipante, S.J. et al., 2014. Performance comparison of Illumina and ion torrent next-generation sequencing platforms for 16S rRNA-based bacterial community profiling. *Applied and environmental microbiology*, 80(24), pp.7583–91.

Sanger, F., Nicklen, S. & Coulson, A.R., 1977. DNA sequencing with chain-terminating inhibitors. *Proceedings of the National Academy of Sciences of the United States of America*, 74(12), pp.5463–7.

Schirmer, M. et al., 2015. Insight into biases and sequencing errors for amplicon sequencing with the Illumina MiSeq platform. *Nucleic Acids Research*.

Schloss, P.D., 2009. A high-throughput DNA sequence aligner for microbial ecology studies. *PloS one*, 4(12), p.e8230.

Schloss, P.D. et al., 2009. Introducing mothur: open-source, platform-independent, community-supported software for describing and comparing microbial communities. *Applied and environmental microbiology*, 75(23), pp.7537–41.

Schloss, P.D., 2013. Secondary structure improves OTU assignments of 16S rRNA gene sequences. *The ISME journal*, 7(3), pp.457–60.

Schloss, P.D. et al., 2016. Sequencing 16S rRNA gene fragments using the PacBio SMRT DNA sequencing system. *PeerJ*, 4, p.e1869.

Schloss, P.D. et al., 2015. Sequencing 16S rRNA gene fragments using the PacBio SMRT DNA sequencing system. *PeerJ*.

Schloss, P.D., 2010. The effects of alignment quality, distance calculation method, sequence filtering, and region on the analysis of 16S rRNA gene-based studies. *PLoS computational biology*, 6(7), p.e1000844.

Schloss, P.D., Gevers, D. & Westcott, S.L., 2011. Reducing the effects of PCR amplification and sequencing artifacts on 16S rRNA-based studies. *PloS one*, 6(12), p.e27310.

Schloss, P.D. & Handelsman, J., 2005. Introducing DOTUR, a computer program for defining operational taxonomic units and estimating species richness. *Applied and environmental microbiology*, 71(3), pp.1501–6.

Schloss, P.D. & Handelsman, J., 2006. Toward a census of bacteria in soil. *PLoS computational biology*, 2(7), p.e92.

Schloss, P.D. & Westcott, S.L., 2011. Assessing and improving methods used in operational taxonomic unit-based approaches for 16S rRNA gene sequence analysis. *Applied and environmental microbiology*, 77(10), pp.3219–26.

Schmalenberger, A., Schwieger, F. & Tebbe, C.C., 2001. Effect of primers hybridizing to different evolutionarily conserved regions of the small-subunit rRNA gene in PCR-based microbial community analyses and genetic profiling. *Applied and environmental microbiology*, 67(8), pp.3557–63.

Schmidt, T.S.B., Matias Rodrigues, J.F. & von Mering, C., 2014. Ecological Consistency of SSU rRNA-Based Operational Taxonomic Units at a Global Scale. *PLoS Computational Biology*, 10(4), p.e1003594.

Schmidt, T.S.B., Matias Rodrigues, J.F. & von Mering, C., 2015. Limits to robustness and reproducibility in the demarcation of operational taxonomic units. *Environmental microbiology*, 17(5), pp.1689–706.

Sender, R., Fuchs, S. & Milo, R., 2016. *Revised estimates for the number of human and bacteria cells in the body*, Cold Spring Harbor Labs Journals.

Singer, E. et al., 2016. High-resolution phylogenetic microbial community profiling. *The ISME Journal*.

Smyth, R.P. et al., 2010. Reducing chimera formation during PCR amplification to ensure accurate genotyping. *Gene*, 469(1-2), pp.45–51.

Soergel, D.A.W. et al., 2012. Selection of primers for optimal taxonomic classification of environmental 16S rRNA gene sequences. *The ISME journal*, 6(7), pp.1440–4.

Sogin, M.L. et al., 2006. Microbial diversity in the deep sea and the underexplored "rare biosphere". *Proceedings of the National Academy of Sciences of the United States of America*, 103(32), pp.12115–20.

Stackebrandt, E. et al., 2002. Report of the ad hoc committee for the re-evaluation of the species definition in bacteriology. *International journal of systematic and evolutionary microbiology*, 52(Pt 3), pp.1043–7.

Stackebrandt, E. & Ebers, J., 2006. Taxonomic parameters revisited: tarnished gold standards. *Microbiology today*, 33(4), p.152.

Stackebrandt, E. & Goebel, B.M., 1994. Taxonomic Note: A Place for DNA-DNA Reassociation and 16S rRNA Sequence Analysis in the Present Species Definition in Bacteriology. *International Journal of Systematic Bacteriology*, 44(4), pp.846–849.

Staley, J.T., 2006. The bacterial species dilemma and the genomic-phylogenetic species concept.

Philosophical transactions of the Royal Society of London. Series B, Biological sciences, 361(1475), pp.1899–909.

Staley, J.T., 2009. Universal species concept: pipe dream or a step toward unifying biology? *Journal of industrial microbiology & biotechnology*, 36(11), pp.1331–6.

Stevens, H. & Ulloa, O., 2008. Bacterial diversity in the oxygen minimum zone of the eastern tropical South Pacific. *Environmental microbiology*, 10(5), pp.1244–59.

Sun, Y. et al., 2012. A large-scale benchmark study of existing algorithms for taxonomy-independent microbial community analysis. *Briefings in bioinformatics*, 13(1), pp.107–21.

Sun, Y. et al., 2009. ESPRIT: estimating species richness using large collections of 16S rRNA pyrosequences. *Nucleic acids research*, 37(10), p.e76.

Thompson, J.D., Higgins, D.G. & Gibson, T.J., 1994. CLUSTAL W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice. *Nucleic acids research*, 22(22), pp.4673–80.

Thompson, J.R., Marcelino, L.A. & Polz, M.F., 2002. Heteroduplexes in mixed-template amplifications : formation , consequence and elimination by " reconditioning PCR ." , 30(9), pp.2083–2088.

Torsvik, V. & Øvreås, L., 2002. Microbial diversity and function in soil: from genes to ecosystems. *Current opinion in microbiology*, 5(3), pp.240–5.

Tsai, I.J. et al., 2014. Summarizing specific profiles in Illumina sequencing from whole-genome amplified DNA. *DNA research : an international journal for rapid publication of reports on genes and genomes*, 21(3), pp.243–54.

Turcatti, G. et al., 2008. A new class of cleavable fluorescent nucleotides: synthesis and optimization as reversible terminators for DNA sequencing by synthesis. *Nucleic acids research*, 36(4), p.e25.

Turnbaugh, P.J. et al., 2010. Organismal, genetic, and transcriptional variation in the deeply sequenced gut microbiomes of identical twins. *Proceedings of the National Academy of Sciences of the United States of America*, 107(16), pp.7503–8.

Üstün, B., Melssen, W.J. & Buydens, L.M.C., 2006. Facilitating the application of Support Vector Regression by using a universal Pearson VII function based kernel. *Chemometrics and Intelligent Laboratory Systems*, 81(1), pp.29–40.

Vandamme, P. et al., 1996. Polyphasic taxonomy, a consensus approach to bacterial systematics. *Microbiological reviews*, 60(2), pp.407–38.

Venter, J.C. et al., 2004. Environmental genome shotgun sequencing of the Sargasso Sea. *Science (New York, N.Y.)*, 304(5667), pp.66–74.

Vinje, H. et al., 2014. A systematic search for discriminating sites in the 16S ribosomal RNA gene. *Microbial informatics and experimentation*, 4(1), p.2.

Wang, G.C. & Wang, Y., 1997. Frequency of formation of chimeric molecules as a consequence of PCR coamplification of 16S rRNA genes from mixed bacterial genomes. *Appl Environ Microbiol*, 63(12), pp.4645–4650.

Wang, G.C. & Wang, Y., 1996. The frequency of chimeric molecules as a consequence of PCR co-amplification of 16S rRNA genes from different bacterial species. *Microbiology (Reading, England)*, 142 (Pt 5, pp.1107–14.

Wang, Q. et al., 2007. Naive Bayesian classifier for rapid assignment of rRNA sequences into the new bacterial taxonomy. *Applied and environmental microbiology*, 73(16), pp.5261–7.

Wang, X. et al., 2012. Secondary structure information does not improve OTU assignment for partial 16s rRNA sequences. *The ISME journal*, 6(7), pp.1277–80.

Ward, B.B., 2002. How many species of prokaryotes are there? *Proceedings of the National Academy of Sciences of the United States of America*, 99(16), pp.10234–6.

Watson, J.D. & Crick, F.H., 1953. Molecular structure of nucleic acids; a structure for deoxyribose nucleic acid. *Nature*, 171(4356), pp.737–8.

Wayne, L.G., 1988. International Committee on Systematic Bacteriology: announcement of the report of the ad hoc Committee on Reconciliation of Approaches to Bacterial Systematics. *Zentralblatt für Bakteriologie, Mikrobiologie, und Hygiene. Series A, Medical microbiology, infectious diseases, virology, parasitology*, 268(4), pp.433–4.

Westcott, S.L. & Schloss, P.D., 2015. *De novo* clustering methods outperform reference-based methods for assigning 16S rRNA gene sequences to operational taxonomic units. *PeerJ*, 3, p.e1487.

Whiteman, L., 2008. Microbes to People: Without Us, You're Nothing. *U. S. News & World Report*.

Whitman, W.B., Coleman, D.C. & Wiebe, W.J., 1998. Prokaryotes: the unseen majority. *Proceedings of the National Academy of Sciences of the United States of America*, 95(12), pp.6578–83.

Whittaker, R.H., 1972. Evolution and Measurement of Species Diversity. *Taxon*, 21(2/3), pp.213–251.

Woese, C.R., 1987. Bacterial evolution. *Microbiological reviews*, 51(2), pp.221–71.

Wright, E.S., Yilmaz, L.S. & Noguera, D.R., 2012. DECIPHER, a search-based approach to chimera identification for 16S rRNA sequences. *Applied and environmental microbiology*, 78(3), pp.717–25.

Wu, D., Jospin, G. & Eisen, J.A., 2013. Systematic Identification of Gene Families for Use as "Markers" for Phylogenetic and Phylogeny-Driven Ecological Studies of Bacteria and Archaea and Their Major Subgroups. *PLoS ONE*, 8(10), p.e77033.

Wu, R., 1970. Nucleotide sequence analysis of DNA. I. Partial sequence of the cohesive ends of bacteriophage lambda and 186 DNA. *Journal of molecular biology*, 51(3), pp.501–21.

Yarza, P. et al., 2008. The All-Species Living Tree project: a 16S rRNA-based phylogenetic tree of all sequenced type strains. *Systematic and applied microbiology*, 31(4), pp.241–50.

Yarza, P. et al., 2014. Uniting the classification of cultured and uncultured bacteria and archaea using 16S rRNA gene sequences. *Nature Reviews. Microbiology*, 12(9), pp.635–645.

Yergeau, E. et al., 2012. Next-generation sequencing of microbial communities in the Athabasca River and its tributaries in relation to oil sands mining activities. *Applied and environmental microbiology*, 78(21), pp.7626–37.

Yilmaz, P. et al., 2014. The SILVA and "All-species Living Tree Project (LTP)" taxonomic frameworks. *Nucleic acids research*, 42(Database issue), pp.D643–8.

Youssef, N. et al., 2009. Comparison of species richness estimates obtained using nearly complete fragments and simulated pyrosequencing-generated fragments in 16S rRNA gene-based environmental surveys. *Applied and environmental microbiology*, 75(16), pp.5227–36.

Yu, Z. & Morrison, M., 2004. Comparisons of Different Hypervariable Regions of rrs Genes for Use in Fingerprinting of Microbial Communities by PCR-Denaturing Gradient Gel Electrophoresis. *Applied and Environmental Microbiology*, 70(8), pp.4800–4806.

Zhang, J. et al., 2014. PEAR: a fast and accurate Illumina Paired-End reAd mergeR. *Bioinformatics (Oxford, England)*, 30(5), pp.614–20.

Zhang, Y.-J. et al., 2015. Impacts of gut bacteria on human health and diseases. *International journal of molecular sciences*, 16(4), pp.7493–519.

Zuckerkandl, E. & Pauling, L., 1965. Molecules as documents of evolutionary history. *Journal of theoretical biology*, 8(2), pp.357–66.

APPENDIX A: ADDITIONAL INFORMATION

Additional Information 1:1: *Mock Communities*

Seven different mock communities have been used throughout this thesis to test the various algorithms for 16S rRNA data processing (e.g. chimera detection, denoising, clustering). The composition of the first mock consist of: *Acinetobacter baumannii* ATCC 17978,*Actinomyces odontolyticus* ATCC 17982, *Bacillus cereus* ATCC 10987, *Bacteroides vulgatus*ATCC 8482, *Clostridium beijerinckii* ATCC 51743, *Deinococcus radiodurans* ATCC 13939,*Enterococcus faecalis* ATCC 47077, *Escherichia coli* ATCC 70096, *Helicobacter pylori*ATCC 700392, *Lactobacillus gasseri* ATCC 33323, *Listeria monocytogenes* ATCC BAA-679, *Neisseria meningitidis* ATCC BAA-335, *Porphyromonas gingivalis* ATCC 33277,*Propionibacterium acnes* DSM 16379, *Pseudomonas aeruginosa* ATCC 47085, *Rhodobacter sphaeroides* ATCC 17023, *Staphylococcus aureus* ATCC BAA-1718, *Staphylococcus epidermidis* ATCC 12228, *Streptococcus agalactiae* ATCC BAA-611, *Streptococcus mutans*ATCC 700610, and *Streptococcus pneumoniae* ATCC BAA-334.

The second mock consists of *Acinetobacter baumannii str. 5377*, *Actinomyces odontolyticus str.* 1A.21, *Bacillus cereus* str. NRS 248, *Bacteroides vulgatus* str. NCTC 11154, *Clostridium beijerinckii* str. NCIMB 8052, *Deinococcus radiodurans* str. R1 (smooth), *Enterococcus faecalis str.* OG1RF, *Escherichia coli* str. K12 substr. MG1655, *Helicobacter pylori* str. 26695, *Lactobacillus gasseri* str. 63 AM, *Listeria monocytogenes str. EGDe*, *Neisseria meningitidis* str. MC58, *Propionibacterium acnes* str. KPA171202, *Pseudomonas aeruginosa* str. PAO1-LAC, *Rhodobacter sphaeroides str.* ATH 2.4.1, *Staphylococcus aureus* TCH1516, *Staphylococcus epidermidis* FDA str. PCI 1200, *Streptococcus agalactiae* str. 2603 V/R, *Streptococcus mutans* str. UA159, and *Streptococcus pneumoniae str.* TIGR4.

The third mock consists of *Lactobacillus casei* LMG 6904, *Cupriavidus pinatubonensis* LMG 1197, *Cupriavidus metallidurans* LMG 1195, *Pseudomonas putida* LMG 24210, *Lysinibacillus spaericus* LMG 22257, *Alcanivorax borkumensis* SK2, *Alcanivorax dieselolei* 293, *Roseburia hominis* A2-138, *Geobacter sulfurreducens* DSM 12127, *Acetobacterium woodii* DSM 1030, *Porphyromonas gingivalis* ATCC 33277, and *Fusobacterium nucleatum* ATCC 10953.

The forth mock consists of *Bifidobacterium adolescentis , Bifidobacterium angulatum , Bifidobacterium bifidum , Bifidobacterium breve, Bifidobacterium gallicum , Bifidobacterium catenulatum , Bifidobacterium dentium , Bifidobacterium pseudocatenulatum, Collinsella intestinalis, Collinsella stercoris , Alistipes putredinis , Bacteroides pectinophilus , Bacteroides coprocola , Bacteroides coprophilus , Bacteroides dorei , Bacteroides eggerthii , Bacteroides finegoldii, Bacteroides intestinalis , Bacteroides plebeius , Bacteroides caccae, Bacteroides cellulosilyticus , Bacteroides ovatus , Bacteroides stercoris , Parabacteroides johnsonii , Prevotella copri , Anaerococcus hydrogenalis , Anaerofustis stercorihominis , Anaerotruncus colihominis , Butyrivibrio crossotus , Catenibacterium mitsuokai , Clostridium sporogenes, Clostridium asparagiforme , Clostridium bartlettii , Clostridium celatum , Clostridium hathewayi , Clostridium hiranonis , Clostridium hylemonae, Clostridium methylpentosum , Clostridium nexile , Clostridium orbiscindens , Clostridium ramosum, Clostridium saccharolyticum, Coprococcus comes, Dorea formicigenerans , Eubacterium biforme,*

Eubacterium ramulus , Holdemania filiformis , Mitsuokella multacida , Roseburia faecis , Roseburia intestinalis , Roseburia inulinivorans , Ruminococcus hansenii , Ruminococcus torques , Ruminococcus gnavus, Ruminococcus lactaris , Streptococcus infantarius , Subdoligranulum variabile , Providencia rustigianii , Cedecea davisae, Citrobacter sp. , Desulfitobacterium hafniense , Edwardsiella tarda , Enterobacter cancerogenus, Proteus penneri , Providencia alcalifaciens, Providencia rettgeri, and *Providencia stuartii.*

The fifth mock consists of *Acinetobacter baumanii* (NC_009085), *Actinomyces odontolyticus* (DS264586), *Bacillus cereus* (AE017194), *Bacteroides vulgatus* (NC_009614), *Clostridium beijerinckii* (NC_009617), *Deinococcus radiodurans* (NC_001263), *Enterococcus faecalis (NC_004668), Escherichia coli* (NC_000913), *Helicobacter pylori* (NC_000915), *Lactobacillus gasseri* (NC_008530), *Listeria monocytogenes (NC_003210), Neisseria meningitidis* (NC_003112), *Propionibacterium acnes* (NC_006085), *Pseudomonas aeruginosa* (NC_002516), *Rhodobacter sphaeroides* (NC_007493, NC_007494), *Staphylococcus aureus* (NC_007793), *Staphylococcus epidermidis* (NC_004461), *Streptococcus agalactiae* (NC_004116), *Streptococcus mutans* (NC_004350), *Streptococcus pneumoniae* (NC_003028), and *Methanobrevibacter smithii* (NC_009515).

The sixth mock consists of *Acidobacterium capsulatum* ATCC 51196, *Akkermansia muciniphila* ATCC BAA-835, *Anaerocellum thermophilum* Z-1320, DSM 6725, *Bacteroides thetaiotaomicron* VPI-5482, *Bacteroides vulgatus* ATCC 8482, *Bordetella bronchiseptica* RB50, *Burkholderia xenovorans* LB400, *Caldicellulosiruptor saccharolyticus* DSM 8903, *Chlorobaculum tepidum TLS, Chlorobium limicola* DSM 245, *Chlorobium phaeobacteroides* DSM 266, *Chlorobium phaeovibrioides* DSM 265, *Chloroflexus aurantiacus* J-10-fl, *Clostridium thermocellum* ATCC 27405, *Deinococcus radiodurans R1, Desulfovibrio desulfuricans desulfuricans* ATCC 27774, *Desulfovibrio piger* ATCC 29098, *Dictyoglomus turgidum* DSM 6724, *Erwinia chrysanthemi, Enterococcus faecalis* V583v, *Fusobacterium nucleatum nucleatum* ATCC 25586, *Gemmatimonas aurantiaca* T-27T, *Herpetosiphon aurantiacus* ATCC 23779, *Hydrogenobaculum* sp. Y04AAS1, *Leptothrix cholodnii SP-6, Nitrosomonas europaea* ATCC 19718, *Nostoc* sp. PCC 7120, *Pelodictyon phaeoclathratiforme* BU-1, *Persephonella marina* EX-H1, *Porphyromonas gingivalis* ATCC 33277, *Rhodopirellula baltica* SH 1, *Rhodospirillum rubrum* ATCC 11170, *Ruegeria pomeroyi* DSS-3, *Salinispora arenicola* CNS-205, *Salinispora tropica* CNB-440, *Shewanella baltica* OS185, *Shewanella baltica* OS223, *Sulfitobacter* sp. EE-36, *Sulfitobacter sp.* NAS-14.1, *Sulfurihydrogenibium sp. YO3AOP1, Sulfurihydrogenibium yellowstonense* SS-5, *Thermoanaerobacter pseudethanolicus* ATCC 33223, *Thermotoga neapolitana* DSM 4359, *Thermotoga petrophila* RKU-1, *Thermotoga sp. RQ2, Thermus thermophilus HB8, Treponema denticola* ATCC 35405, *Treponema vincentii I, Zymomonas mobilis mobilis* ZM4, *Archaeoglobus fulgidus* DSM 4304, *Ignicoccus hospitalis* KIN4/I, *Methanocaldococcus jannaschii* DSM 2661, *Methanococcus maripaludis* C5, *Methanococcus maripaludis S2, Nanoarchaeum equitans* Kin4-M, *Pyrobaculum aerophilum* IM2, *Pyrobaculum calidifontis* JCM 11548, *Pyrococcus horikoshii OT3,* and *Sulfolobus tokodaii 7*(S311).

The seventh mock community consists of *Acidovorax facilis* (DSM 649), *Pseudomonas xanthomarina* (KMM 1447), *Pseudomonas aeruginosa* (DSM 50071), *Paracoccus denitrificans* (NBRC 102528), *Rhodospirillum rubrum* (ATCC 11170), *Microbacterium phyllosphaerae* (P 369/06), *Arthrobacter oryzae (*KV-651), *Delftia tsuruhatensis* (NBRC 16741), *Nitrosomonas europaea* (ATCC 25978), *Cupriavidus metallidurans* (CH34), *Clostridium botulinum (*ELTDK 103), *Staphylococcus aureus* (NBRC 100910), *Bacillus cereus (ATCC 14579), Arthrospira platensis* (PCC 7345), *Enterococcus faecium* (NBRC 100486), *Yersinia enterocolitica* (Y11), and *Desulfovibrio oxamicus* (DSM 1925).

Mock Communities	1	2	3	4	5			6		7
number of species	21	20	12	67	20			59		17
number of run	3	2	1	1	3			2		1
Platform	Illumina (MiSeq)	Illumina (MiSeq)	Illumina (MiSeq)	roche 454 Pyrosequencing	roche 454 Pyrosequencing			Illumina (MiSeq)		roche 454 Pyrosequencing
Chemistry version	V2	V2	V3	GS-FLX	Titanium			V2		Titanium
16S rRNA region	V3-V4	V4	V1-V3	V2	V1-V3	V3-V5	V6-V9	V4	V1-V9	V1-V3
Length after assembly	422-428	251-253	422-428	-	-	-	-	253, 292	330	-
Length of overlapping region	70	250	141	-	-	-	-	250, 210	170	-
Number of samples	4	4	3	3	30	30	30	57	14	3
Sample IDs	130401(V34), 130403(V34), 130417(V34), 130422(V34)	130401(V4), 130403(V4), 130417(V4), 130422(V4)	130401(V45), 130403(V45), 130417(V45), 130422(V45)	V4.V5.1.1, V4.V5.1.11	V4.1.1, V4.1.05			IDs: 19-26, 35-53, 59-62, 64-69, 71, 74-83, 85-91, 93, 94, 96	IDs: 27-34, 97-102	50, 51
Source	MOCK1 is available via (http://www.mothur.org/MiSeqDevelopment Data.html).	EBI SRA project ID:PRJEB4688	NCBI SRA project ID:SRP066114	NCBI SRA project ID: SRP003773	10 samples (in triplicates)	10 samples (in triplicates)	10 samples (in triplicates)	NCBI SRA project ID: SRP002397	EBI SRA project ID:PRJEB6244	NCBI SRA project ID: PRJNA257992
Reference	Kozich et al (2013)	Nelson et al. 2014	Mysara et al 2016	Turnbaugh et al 2010	Schloss et al 2011			Schirmer et al. 2015		Mysara et al 2015
Number of samples used in the various chapters										
Chapter2	-	-	-	-	1 training (TrainingDB), 68 testing (MOCK1), 21 missing			-		3 testing (MOCK2)
Chapter3	4 training (TrainingDB), 9 testing (MOCK1)	4 testing (MOCK2)	3 testing (MOCK3)	-	-			69 testing (MOCK4)		-
Chapter4	12 testing (MOCK2)	-	-	3 training (TrainingDB)	90 testing (MOCK3)			-		-
Chapter5	-	-	-	-	-			34 testing (rejecting the other samples having less than 25,000 reads)		-
Chapter6	6 testing [V3-V4 and V4], (MOCK1)	4 testing (MOCK2)	3 testing (MOCK3)	-	-			-		-

Additional Information 2:1: *MOCK2 Concentration*

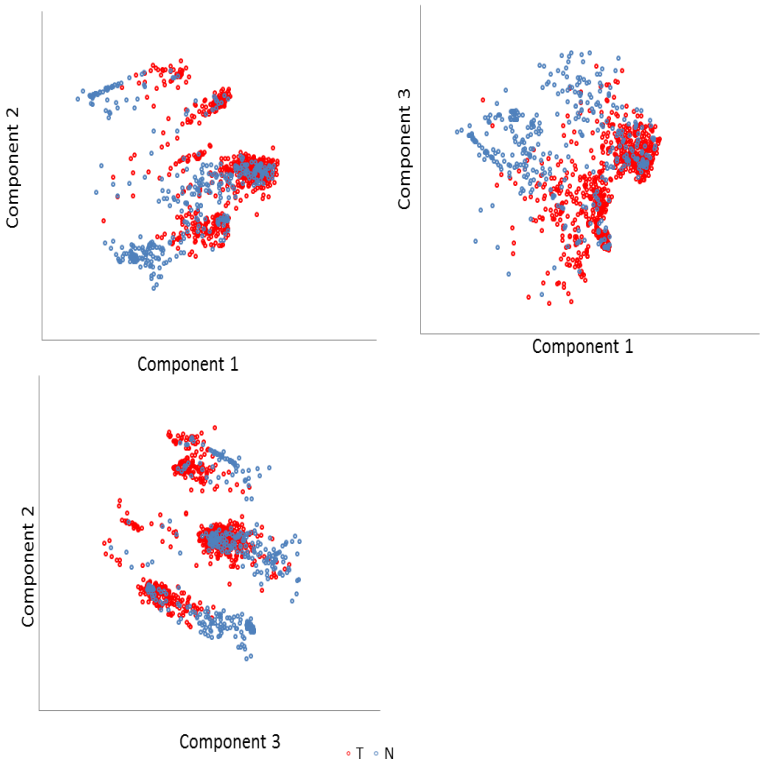
Table giving the targeted concentrations for the MOCK2 community, and the exact concentrations together with the number of paralogous genes in the 16S genes of the reference genomes), accessible via this link (https://static-content.springer.com/esm/art%3A10.1186%2Fs12859-015-0520-5/MediaObjects/12859_2015_520_MOESM1_ESM.pdf).

Additional Information 2:2: *Taxonomic Classification of MOCK2*

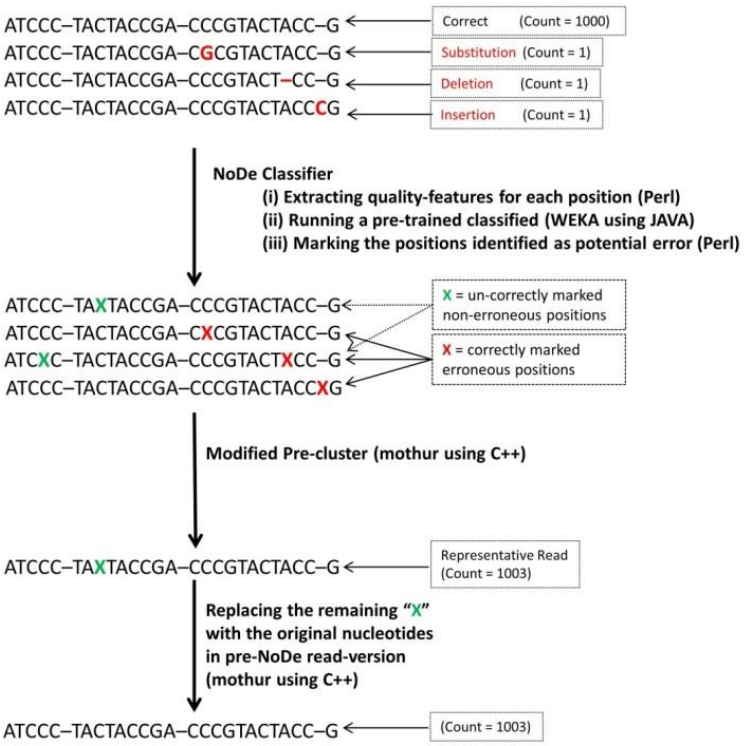
Excel sheet of the detail taxonomic classification of each of the three MOCK2 replicates after being treated with the different denoising approaches, together with the taxonomic classification of the reference sequences for the same region, accessible via this link (https://static-content.springer.com/esm/art%3A10.1186%2Fs12859-015-0520-5/MediaObjects/12859_2015_520_MOESM2_ESM.xlsx).

Additional Information 2:3: *Principle Component Analysis*

Illustration of principle component analysis showing biplots for component 1 versus component 2 (top left), component 1 versus component 3 (top right), component 2 versus component 3 (bottom) Bases on these plots it can be derived that for a significant portion of the data it will be hard to separate erroneous (N) and non-erroneous (T) instances.

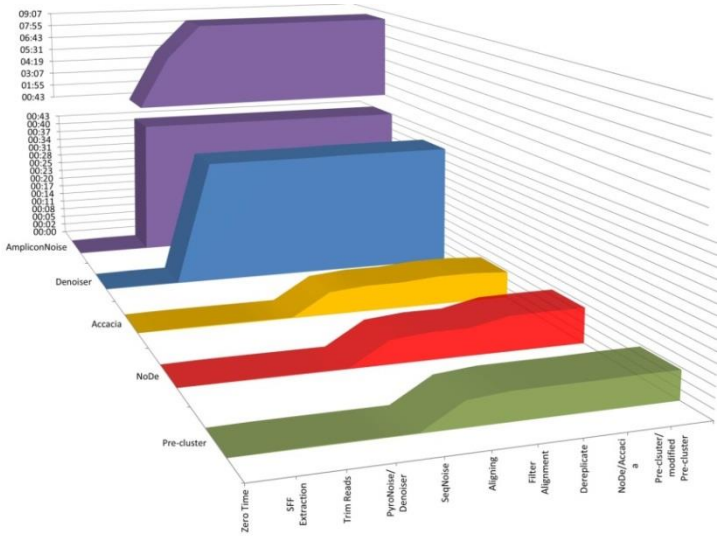


Additional Information 2:4: *Schematic Overview Showing the Different steps of the NoDe Algorithm.*



Additional Information 2:5: *Schematic Overview of the Computational Cost*

Schematic overview of the computational cost of the different denoising algorithms. To have an idea about the computational cost for each step, the complete pipeline was subdivided over different steps to illustrate its running time. Trimming, aligning and filtering of the alignments are recurring steps for all approaches, as shown in the manuscript. AmpliconNoise (that consist of PyroNoise and SeqNoise) and Denoiser use as input directly the flow file (before the trimming step), but NoDe (consisting of the classifier and the modified Pre-cluster algorithm) and Pre-cluster are applied on the multiple sequence alignments. From the graph, it can be observed that the computational burden added on Pre-cluster (by integrating the NoDe algorithm) was only minor, and it was largely compensated with a significant improvement in the error rate, that exceeded the second best performing (but computationally intensive) algorithm AmpliconNoise.



Additional Information 2:6: *Rate of the Different Error Types*

Illustration of the percentage of different error types (insertion, deletion and substitution) after being treated by different algorithms using the MOCK1 dataset

Basic	Insertion	Deletion	Substitution
Initial	0.0008	0.0008	0.0010
Denoiser	0.0007	0.0009	0.0009
Pre-cluster	0.0004	0.0005	0.0005
AmpliconNoise	0.0003	0.0005	0.0006
NoDe	0.0002	0.0003	0.0003

Additional Information 2:7: *Expected versus Actual Percentage of Species After Denoising Algorithms*

Denoising algorithms effect on OTU level. Three figures. A) the logarithmic percentage of each OTU against the expected percentage, B) the relative deviation from the expected value for each OTU. C) Information on extra analyses performed to assess the effect of denoising algorithms on the clustering, via this link (https://static-content.springer.com/esm/art%3A10.1186%2Fs12859-015-0520-5/MediaObjects/12859_2015_520_MOESM7_ESM.pdf).

Additional Information 3:1: *From Sequencing Data of a Mock community to Training Dataset*

As training data, 3,000 reads were randomly selected from sample 130401 of MOCK1 (1,000 reads from the V34, V4 and V45 regions respectively). Important to notice is that all three samples used during the training process are completely disregarded in the subsequent benchmarking analysis. Each nucleotide in those reads was evaluated as being erroneous (mismatch, insertion or deletion) or correct based on a aligning those reads against the reference genomes using a combination of Blast and ClustalW.

This lead to a training dataset consisting of 1,031,625 instances representing nucleotide positions. This training data was cleaned as follows: dereplication (668,962 remaining), removal of outliers based on interquartile ranges (remainder of 664,301), randomization, and simplification via selecting subset of the features [see below]. Next, the data are split into three folds preserving the same ratio between errors:non-errors instances throughout the three folds: (a) a learning data set for training the classifier, (b) a validation set for selecting the most optimal classifier and (c) a testing data set for final validation. Subset (a) was further modified by adjusting the ratio between erroneous and non-erroneous instances [see below], and for subset (b) we used an equal ratio between both classes, while subset (c) was kept unchanged.

Attribute Selection

Feature selection is performed using a combination of two algorithms i.e. a searching algorithm and an evaluation algorithm. Here we applied a search algorithm named "best first search" that allows backtracking along the searching process via moving through the search space making an addition or a deletion of a single feature subset at a time. In the occasions where the search path leads to a less promising subset of features, it would recall a more promising subset and continue searching from there. Subsets are evaluated via an evaluation algorithm named "CfsSubsetEval " that allows identification of those features having a high predictive power for the class (distinguishing erroneous positions from correct positions) via selecting subset of feature that have a high correlation with the class label, while having a low inter=correlation with other predictive features. Here we use three folds cross validation to select the minimum set of feature with a high capability to distinguish erroneous instances from a total of 17 features.

Forward attributes	**Percentage**	**Reverse attributes**	**Percentage**
Position number	100	Position number	33
Succeeding position homopolymer	0	Succeeding position homopolymer	0
Position homopolymer	33	Position homopolymer	0
Proceeding position homopolymer	0	Proceeding position homopolymer	0
Succeeding position Phred	0	Succeeding position Phred	0
Position Phred	100	Position Phred	100
Proceeding position Phred	0	Proceeding position Phred	0
Motif marker	0	Motif marker	0
Contig attribute		Percentage	
Relation between forward and reverse calling		100	

Selecting The Optimal Ratio between Error:Non-error Instances.

There is a skewed class distribution in the complete training data set as the ratio between error : non-error instances is 1 : 45. Machine learning classifiers have difficulties when dealing with imbalanced data, as they would obtain the maximum accuracy when they would skew their prediction towards the majority class (in this case the non-erroneous instances). Detailed description of this step is illustrated in the link (https://static-content.springer.com/esm/art%3A10.1186%2Fs12859-016-1061-2/MediaObjects/12859_2016_1061_MOESM1_ESM.docx), section 1.

Additional Information 3:2: *Classifier Building*

To test the performance of each of the considered classifiers using their default settings, we used the learning set (subset "a") for training the classifiers and validation set (subset "b") for testing the classifiers. Random Forest was found to have the highest performance

single tools	Specificity	Sensitivity	MCC	ROC
MLP	92	44	42	82
SMO	1	99.9	7	51
IBK	86	58	46	76
Logistic	96	18	22	72
RandomForest	91	59	52	85

Several parameter settings have been tested for the selected classifiers (implemented in WEKA). Here we show the performance of each classifier evaluated via the Matthews Correlation Coefficient (MCC) upon testing using the TrainingDB subset (b) after training using the TrainingDB subset (a). Extra information describing the various parameters and their performance, as well as the voting classifier is illustrated in the link (https://static-content.springer.com/esm/art%3A10.1186%2Fs12859-016-1061-2/MediaObjects/12859_2016_1061_MOESM1_ESM.docx), section 2.

Additional Information 3:3: *Pre-Processing Sequencing Data*

The sequencing data (MOCK1, MOCK2 and MOCK3) were trimmed, aligned, screened, filtered and dereplicated using the mothur software package (v.1.33.3), is illustrated via this link (https://static-content.springer.com/esm/art%3A10.1186%2Fs12859-016-1061-2/MediaObjects/12859_2016_1061_MOESM1_ESM.docx), section 3.

Additional Information 3:4: *Different Paired-End Assemblers*

A comparative analysis was performed to assess the differences between various paired-end assembly tools (and their quality trimming) used for amplicon sequencing implemented in mothur, QIIME and UPARSE, together with PEAR. is illustrated in the link (https://static-content.springer.com/esm/art%3A10.1186%2Fs12859-016-1061-2/MediaObjects/12859_2016_1061_MOESM1_ESM.docx), section 4.

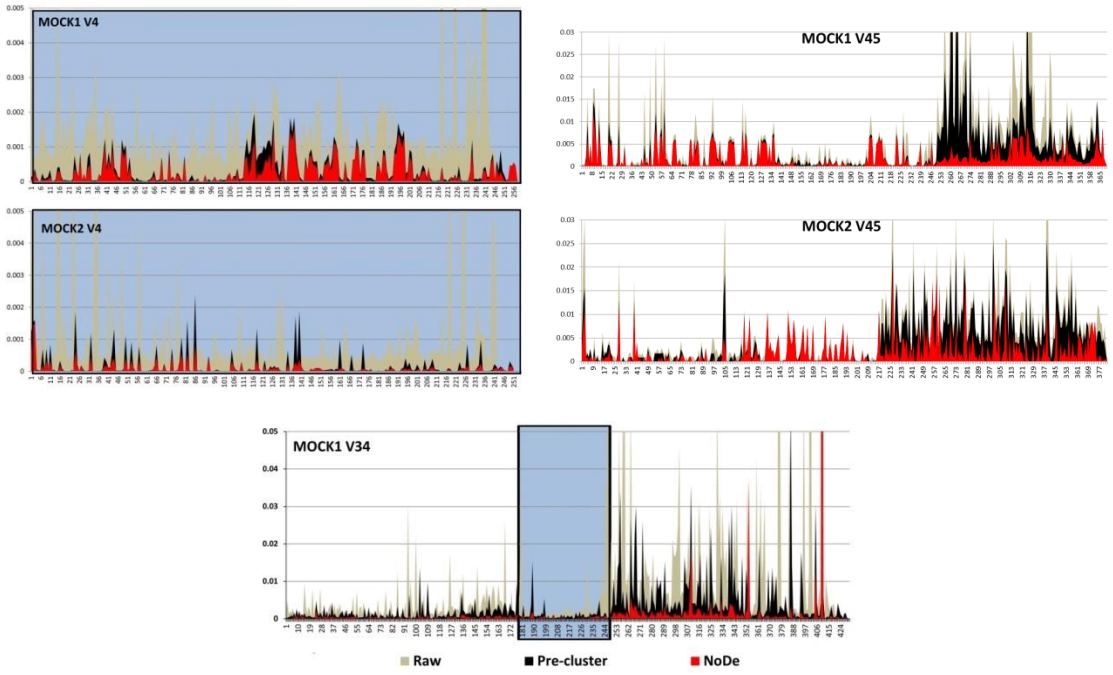
Additional Information 3:5: *MOCK4 Data Analysis*

We examined the effect of applying IPED on a fourth mock community - called MOCK4 - consisting of 73 samples recently published in Schirmer et al (2015). Their microbial composition ranges from single species to diverse mock communities (49 bacteria and 10 Archaea) with either even or uneven abundance distributions. Contigs constructed via merging both reads resulted in different lengths, ranging from 253 nucleotides (i.e. almost completely overlapping reads) to 450 nucleotides (partially overlapping reads). From the results we can observe the same trend in lowering the error rate when applying IPED. Indeed, when both reads are almost completely overlapping (contig length ranging between 253 and 292), IPED was able to reduce the error rate from 0.0041 to 0.0032, while the effect was more prominent when dealing with contigs have a smaller overlap between both reads (contig length ranging between 330 and 450), showing a decrease in the error rate from 0.0065 to 0.0033.

Meta ID	Region	ave. Length	Without IPED	IPED	**Meta ID**	Region	ave. Length	Without IPED	IPED
19	V4	292	0.0082	0.0078	**62**	V4	292	0.0057	0.0055
20	V4	292	0.0081	0.0076	**64**	V4	292	0.0058	0.0057
21	V4	292	0.0081	0.0078	**65**	V4	292	0.0056	0.0055
22	V4	292	0.0082	0.0075	**66**	V4	292	0.0059	0.0058
23	V4	292	0.0077	0.0050	**67**	V4	292	0.0057	0.0055
24	V4	292	0.0077	0.0054	**68**	V4	292	0.0048	0.0042
25	V4	292	0.0077	0.0046	**69**	V4	292	0.0047	0.0037
26	V4	292	0.0078	0.0056	**71**	V4	292	0.0046	0.0035
27	V3/V4	275	0.0053	0.0042	**74**	V4	292	0.0061	0.0058
28	V3/V4	275	0.0041	0.0040	**75**	V4	292	0.0058	0.0056
29	V3/V4	275	0.0024	0.0017	**76**	V4	292	0.0058	0.0000
30	V3/V4	275	0.0046	0.0047	**77**	V4	253	0.0013	0.0008
31	V3/V4	275	0.0047	0.0043	**78**	V4	253	0.0020	0.0017
32	V3/V4	275	0.0032	0.0030	**79**	V4	253	0.0019	0.0016
33	V3/V4	275	0.0024	0.0015	**80**	V4	253	0.0020	0.0017
34	V3/V4	275	0.0038	0.0036	**81**	V4	253	0.0020	0.0017
35	V4	292	0.0063	0.0046	**82**	V4	253	0.0020	0.0015
37	V4	292	0.0074	0.0056	**83**	V4	253	0.0025	0.0019
39	V4	253	0.0022	0.0018	**85**	V4	253	0.0020	0.0015
40	V4	253	0.0029	0.0025	**86**	V4	292	0.0071	0.0043
41	V4	253	0.0022	0.0019	**87**	V4	292	0.0070	0.0034
42	V4	253	0.0022	0.0017	**88**	V4	292	0.0081	0.0048
43	V4	253	0.0019	0.0014	**89**	V4	292	0.0063	0.0026
44	V4	253	0.0023	0.0015	**90**	V4	292	0.0064	0.0027
45	V4	253	0.0022	0.0014	**91**	V4	292	0.0063	0.0028
46	V4	253	0.0021	0.0014	**93**	V4	292	0.0054	0.0014
47	V4	253	0.0015	0.0009	**94**	V4	292	0.0055	0.0016
49	V4	292	0.0065	0.0034	**96**	V4	292	0.0057	0.0018
50	16S	330	0.0017	0.0015	**97**	V3/V4	450	0.0097	0.0056
51	16S	330	0.0056	0.0036	**98**	V3/V4	450	0.0094	0.0045
52	V4	292	0.0051	0.0049	**99**	V3/V4	450	0.0098	0.0049
53	V4	292	0.0057	0.0055	**100**	V3/V4	450	0.0089	0.0037
59	V4	292	0.0063	0.0062	**101**	V3/V4	450	0.0088	0.0039
60	V4	292	0.0061	0.0060	**102**	V3/V4	450	0.0091	0.0039
61	V4	292	0.0063	0.0061					

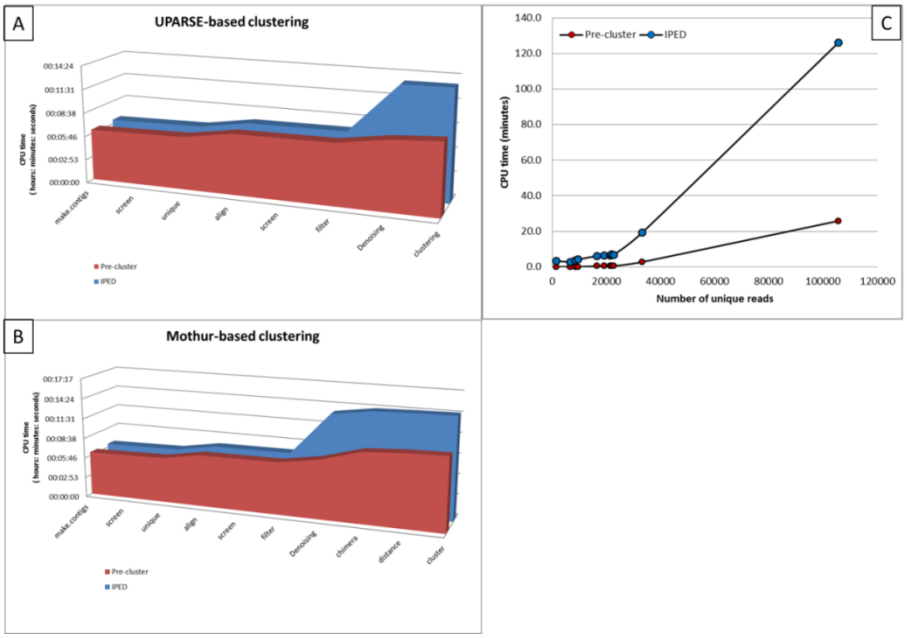
Additional Information 3:6: *Error Rate Over Different Read Positions*

We examined the effect of applying both denoising algorithms relative to the position in the read in comparison to the non-denoised error rate of the samples. As expected, for those samples where both reads were completely overlapping the non-denoised error rates for both mock communities were low. As such, the improvements due to denoising algorithms were found to be minimal. Although, IPED is capable reducing the error rate effectively and robustly throughout all of the read positions, it is more pronounced when both paired reads are not completely overlapping, with the most significant improvement for the second read.



Additional Information 3:7: *Computational Cost Analysis*

To investigate the extra computational cost related to IPED, the calculation time was registered for all three samples of the MOCK1 dataset covering the V4 region, each sample stratified to 6000 unique reads. When one processor (single Intel Xeon E5-2640 2.50 GHz CPU) was used for each sample (i.e. a total of three processors), IPED required 70 seconds for running all three samples, while Pre-cluster could end the analysis in 14 secs. Taking into account all the different steps required in going from raw Illumina MiSeq 16S rRNA sequencing data to operational taxonomic units (OTUs), this will lead to an increase of 40-50% of the total computational time when IPED is integrated, using either UPARSE or mothur OTU clustering approaches. To evaluate the scalability of both approaches against different datasets sizes, we plotted the computational cost of the individual samples treated with IPED and Pre-cluster. As the unique reads were used to represent each sample, they range from 1,589-131,910 (the total number of reads range from 2,084 to 696,310) IPED showed on average a 5-fold increase of computational costs compared to Pre-cluster.



Additional Information 3:8: *Denoising Effect on Operational Taxonomic Units (OTUs) Clustering*

The effect of denoising algorithms on the OTUs clustering in respect to the number of OTUs, error rates and clustering quality is detailed in section 8 of the document accessible via this link (https://static-content.springer.com/esm/art%3A10.1186%2Fs12859-016-1061-2/MediaObjects/12859_2016_1061_MOESM1_ESM.docx).

Additional Information 3:9: *Denoising Effect on a Real-Life Biological Samples*

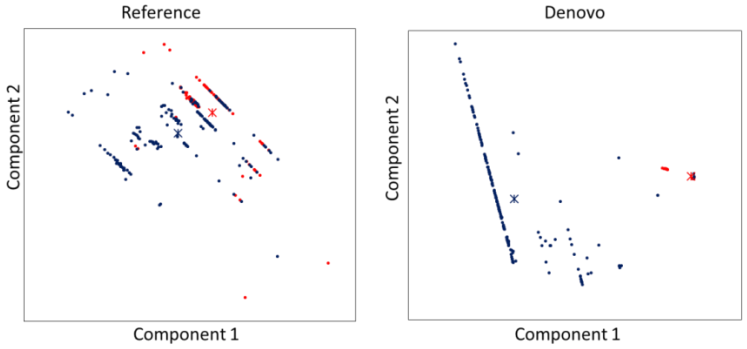
An illustration of the IPED denoising effect on a real-life biological samples is shown at this link (https://static-content.springer.com/esm/art%3A10.1186%2Fs12859-016-1061-2/MediaObjects/12859_2016_1061_MOESM1_ESM.docx), section 9.

Additional Information 3:10: *Effect of Presence of Chimera and Alignment Reference Database*

Is illustrated in section 10 of the document accessible via this link (https://static-content.springer.com/esm/art%3A10.1186%2Fs12859-016-1061-2/MediaObjects/12859_2016_1061_MOESM1_ESM.docx).

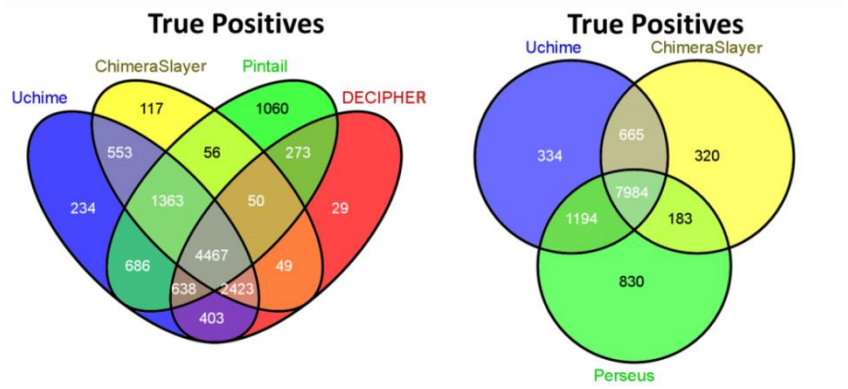
Additional Information 4:1: *Principle Component Analysis*

Plot illustrating the principle component analysis on the training data for both the reference and de novo application, where principal component 1 is in the X axis, and principal component 2 in the Y-axis. The red points represent the non-chimeric class and the blue points represent the chimeric class, respectively. The red asterisk represents the centroid of the non-chimeric class, and the blue asterisk represent the centroid of the chimeric class.



Additional Information 4:2: *Decisions Venn Diagram*

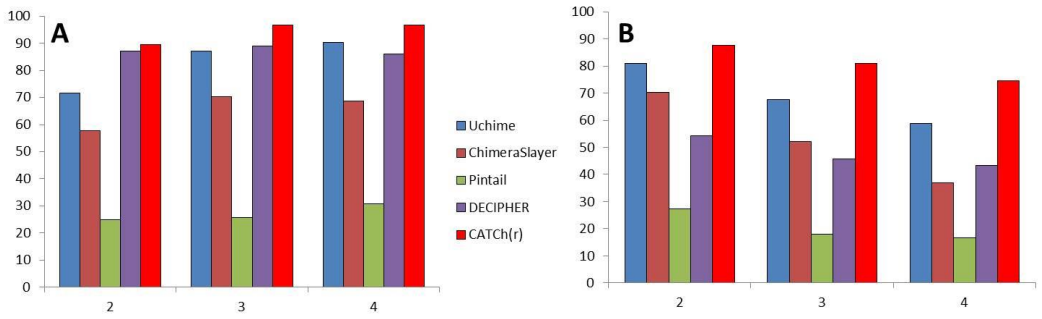
Venn diagrams showing the overlap between the chimeric sequences correctly predicted by individual chimera detection tools in the MOCK2-a dataset. Left: reference based tools; right: de novo tools



.

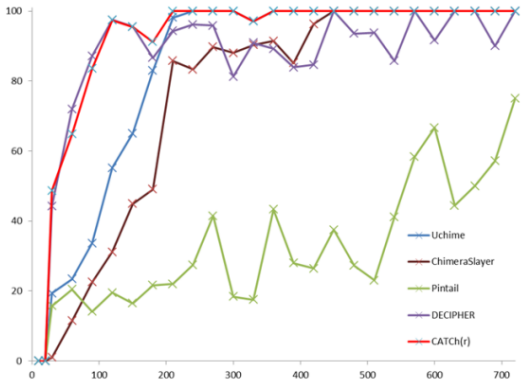
Additional Information 4:3: *Sensitivity and Number of Chimeric Parents*

Sensitivity (in percentage) of all five reference-based algorithms (UCHIME, ChimeraSlayer, Pintail, DECIPHER and CATCh) for detection of different types of chimeras, i.e. having two (bimera), three (trimera) and four (tetramera) parents, applied on (A) Simu3 and (B) Simu2.



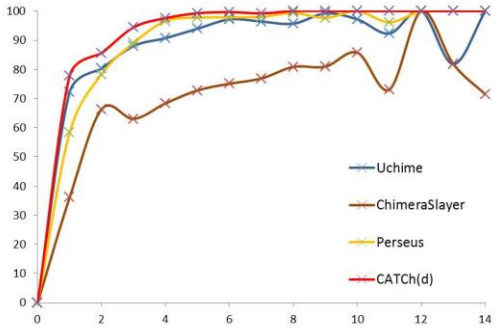
Additional Information 4:4: *Sensitivity and Chimeric Range*

Effect of the chimeric range (X-axis) on the sensitivity of different algorithms (Y-axis) (UCHIME, ChimeraSlayer, Pintail, DECIPHER and CATCh), using Simu3 subset (b).



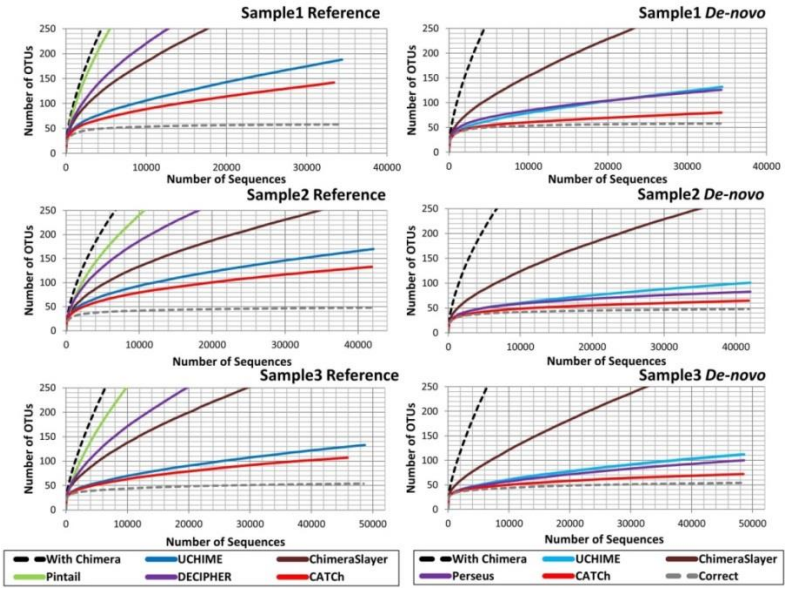
Additional Information 4:5: *Sensitivity and Divergence*

Illustration of the effect of the divergence of the chimeric sequences (X-axis) on sensitivity of the different de novo chimera detection tools (y-axis) using the MOCK1 dataset with a divergence ranging from 1 to 14%. As shown, UCHIME de novo, ChimeraSlayer de novo, Perseus and CATCh de novo tend to detect chimeras with higher divergence more accurately than chimeras with lower divergence. As shown in the figure, CATCh produced the best performance compared to the other tools.



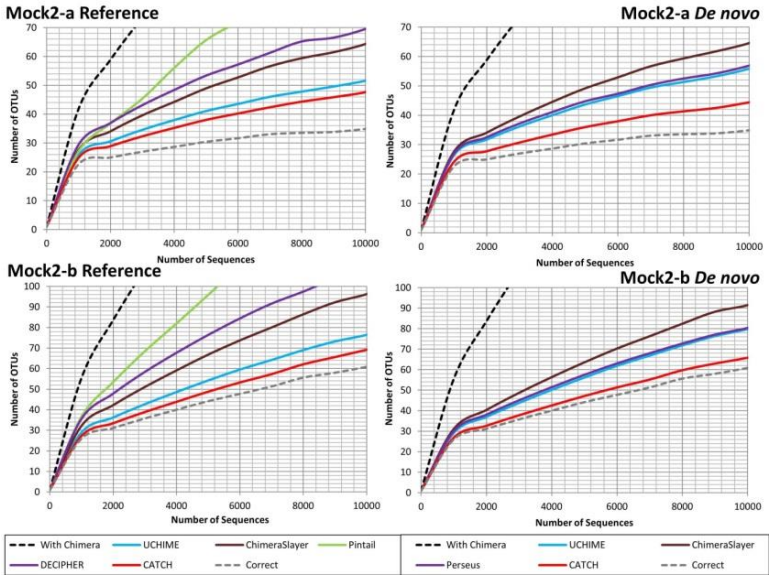
Additional Information 4:6: *Rarefaction Curves MOCK1*

Rarefaction curves for MOCK1 (sample 1,2 and 3) including the datasets i) without any chimera removal ('with chimera'), ii) removing all chimeric sequences ('correct'), and iii) with chimera removal using all tested chimera detection tools.



Additional Information 4:7: *Rarefaction Curves MOCK2*

Rarefaction curves (up to 10,000 reads) for MOCK2 (a and b) including the results i) without any chimera removal ('with chimera'), ii) removing all chimeric sequences ('correct'), and iii) with chimera removal using all tested chimera detection tools (reference-based approaches on the left side and de novo approaches on the right side), averaged over all 69 samples.



Additional Information 4:8: *Comparison between Chimera Detection Tools*

The comparison is illustrated in table S1 of this document (link: http://aem.asm.org/content/suppl/2015/02/10/AEM.02896-14.DCSupplemental/zam999116046so1.pdf)

Additional Information 4:9: *Performance of Different Classifiers*

Overview of the performance of the different regression kernels trained and tested in WEKA according to the procedure described in Material and Methods section. The upper table gives the sensitivity and specificity values obtained on the testing data for the reference based implementation, the lower table gives the overview for the de novo implementation. The selected kernel is highlighted.

Reference	Linear	Multi-Layer	SVM linear	SVM PUK	Decision Stump	Random Forest
Senstivity	0.54	0.80	0.80	0.85	0.59	0.83
Specificity	**0.90**	**0.90**	**0.90**	**0.90**	**0.75**	**0.85**
MCC	0.41	0.66	0.66	0.71	0.31	0.64
Accuracy	0.65	0.83	0.83	0.86	0.64	0.83
De novo	Linear	Multi-Layer	SVM linear	SVM PUK	Decision Stump	Random Forest
Senstivity	0.89	0.91	0.91	0.92	0.80	0.89
Specificity	**0.93**	**0.93**	**0.93**	**0.93**	**0.94**	**0.93**
MCC	0.78	0.80	0.81	0.83	0.68	0.78
Accuracy	0.90	0.91	0.92	0.93	0.84	0.90

Additional Information 4:10: *Principle Component Analysis*

Illustration of the principle component analysis results is shown in table S3 at this link (http://aem.asm.org/content/suppl/2015/02/10/AEM.02896-14.DCSupplemental/zam999116046so1.pdf).

Additional Information 4:11: *Performance of Simplified Model*

Illustration of several classifiers built using all tools except one, is shown in table S4 accessible via this link (http://aem.asm.org/content/suppl/2015/02/10/AEM.02896-14.DCSupplemental/zam999116046so1.pdf).

Additional Information 4:12: *Union and CATCh Performance*

Comparison between different individual tools, the union of all results, and CATCh for A) reference-based algorithms and B) de novo algorithms. The reference-based implementations were tested on 25% of the Titanium dataset (this dataset is not used for training the reference-based CATCh classifier). For the de novo implementations, testing was performed on the F01QS4Z01_rep2_v13 subset from MOCK2-b.

Reference	UCHIME	ChimeraSlayer	Pintail	DECIPHER	Union	CATCh
Sensitivity	0.41	0.24	0.00	0.80	0.87	0.85
Specificity	0.90	0.95	1.00	0.86	0.81	0.90
MCC	0.32	0.23	-	0.63	0.66	0.72
Accuracy	0.57	0.46	0.31	0.82	0.85	0.87
De novo	**UCHIME**	**ChimeraSlayer**	**Perseus**	**Union**	**CATCh**	
Sensitivity	0.79	0.71	0.79	0.93	0.92	
Specificity	0.93	0.91	0.94	0.87	0.93	
MCC	0.66	0.56	0.67	0.79	0.82	
Accuracy	0.83	0.77	0.83	0.91	0.92	

Additional Information 4:13: *Merging Reference and De novo*

Illustration of the performance of three trained classifiers: CATCh reference, CATCh de novo and CATCh merged, using the MOCK2-a and MOCK2-b datasets. The results show marginal improvement in sensitivity when using CATCh merged compared CATCH de novo however with the same cost in specificity in MOCK2-b and no improvement in MOCK2-a.

	MOCK2-**a**			MOCK2-**b**		
	CATCh	**CATCh**	**CATCh**	**CATCh**	**CATCh**	**CATCh**
Sensitivity	0.82	0.85	0.85	0.93	0.95	0.96
Specificity	0.92	0.91	0.91	0.92	0.91	0.90
MCC	0.70	0.72	0.71	0.82	0.84	0.87
Accuracy	0.86	0.87	0.87	0.92	0.94	0.95

Additional Information 4:14: *Performance with Errors*

Performance of different reference-based tools when dealing with deletions, insertions, indels and mismatches for Simu4 data, and indels and substitutions for Simu2 data.

		UCHIME	***ChimeraSlayer***	***Pintail***	***DECIPHER***	***CATCh***		***UCHIME***	***ChimeraSlayer***	***Pintail***	***DECIPHER***	***CATCh***
Simu4	***Del(1%***	0.94	0.8	0.12	0.33	0.95	***Del(5%)***	0.75	0.81	0.04	0.16	0.90
	Ind(1%	0.94	0.8	0.15	0.34	0.95	***Ind(5%)***	0.74	0.80	0.14	0.19	0.89
	Ins(1%	0.95	0.8	0.18	0.34	0.95	***Ins(5%)***	0.85	0.79	0.33	0.18	0.91
	Mis(1%	0.94	0.8	0.13	0.32	0.95	***Mis(5%)***	0.85	0.67	0.06	0.25	0.86
Simu2	***Ind(1%***	0.63	0.5	0.21	0.42	0.84	***Ind(5%)***	0.36	0.26	0.26	0.16	0.67
	Sub(1	0.66	0.3	0.20	0.48	0.82	***Sub(5%)***	0.47	0.10	0.20	0.30	0.68

Additional Information 4:15: *Performance with Different Data*

Sensitivity, specificity, Mathew correlation coefficient (MCC) and accuracy values of all tested chimera prediction tools. The last three rows give respectively the average sensitivity and specificity, & the average increase or decrease in sensitivity/specificity when compared to CATCh.

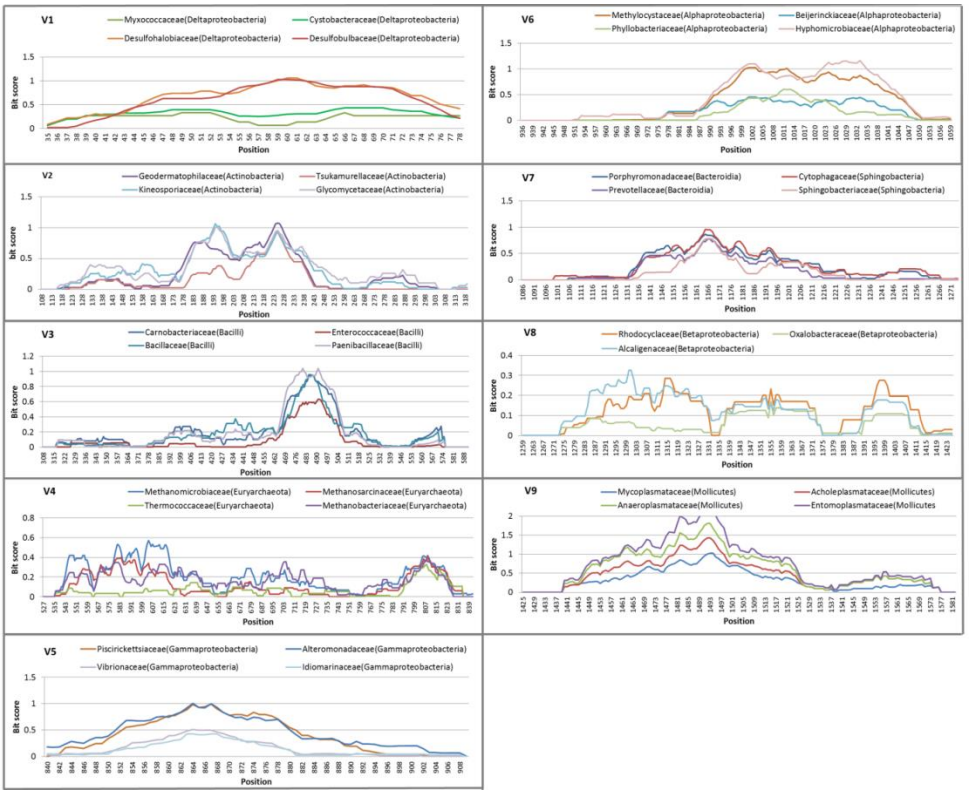
	Reference tools						***De novo* tools**				
Evaluation	**Datasets**	**UCHIME**	**ChimeraSlayer**	**Pintail**	**DECIPHER**	**CATCh**	**Datasets**	**UCHIME**	**ChimeraSlayer**	**Perseus**	**CATCh**
Sensitivity	MOCK2-A	0.79	0.66	0.58	0.61	0.82	MOCK2-A	0.74	0.66	0.76	0.85
Specificity		0.94	0.95	0.42	0.93	0.92		0.94	0.94	0.92	0.91
MCC		0.68	0.56	-0.01	0.49	**0.7**		0.63	0.55	0.62	**0.72**
Accuracy		0.84	0.76	0.52	0.72	**0.86**		0.81	0.76	0.81	**0.87**
Sensitvity	MOCK2-B	0.90	0.77	0.58	0.70	0.93	MOCK2-B	0.87	0.80	0.87	0.95
Specificity		0.94	0.96	0.46	0.93	0.92		0.94	0.94	0.93	0.91
MCC		0.79	0.66	0.04	0.56	**0.82**		0.76	0.67	0.75	**0.84**
Accuracy		0.91	0.82	0.54	0.76	**0.92**		0.89	0.83	0.89	**0.94**
Sensitivity	MOCK3	0.75	0.63	0.06	0.61	0.81	MOCK3	0.64	0.65	0.73	0.81
Specificity		0.95	0.99	1.00	0.98	0.94		0.98	0.97	0.98	0.96
MCC		0.65	0.63	NA	0.6	**0.69**		0.63	0.62	0.69	**0.74**
Accuracy		0.82	0.78	0.5	0.78	**0.84**		0.8	0.79	0.84	**0.87**
Sensitivity	Simu1	0.56	0.51	0.15	0.45	0.70	MOCK1	0.92	0.71	0.94	0.97
Specificity		1.00	1.00	0.88	1.00	0.96		1.00	1.00	1.00	1.00
MCC		0.64	0.6	0.04	0.55	**0.69**		0.7	0.42	0.76	0.87
Accuracy		0.8	0.77	0.54	0.75	**0.84**		0.93	0.74	0.94	0.98
Sensitivity	Simu2	0.69	0.53	0.21	0.48	0.81	MOCK4	0.37	0.27	0.27	0.47
Specificity		1.00	0.99	0.80	0.99	0.98		0.96	0.98	0.94	0.91
MCC		1	0.99	0.8	0.99	**0.98**		0.33	0.28	0.22	0.35
Accuracy		0.4	0.29	0.01	0.27	**0.51**		0.54	0.47	0.45	0.59
Sensitivity	Simu3	0.83	0.66	0.27	0.87	0.94	MOCK5	0.08	0.11	0.14	0.16
Specificity		0.99	1.00	0.86	0.98	0.97		1.00	0.95	0.99	0.98
MCC		0.86	0.76	0.15	0.88	**0.90**		0.24	0.08	0.27	0.26
Accuracy		0.95	0.91	0.71	0.95	**0.96**		0.85	0.81	0.85	0.85
Sensitivity	Simu4	0.95	0.90	0.17	0.27	0.95					
Specificity		1.00	0.99	0.86	1.00	0.97					
MCC		0.96	0.9	0.04	0.44	**0.91**					
Accuracy		0.98	0.96	0.62	0.75	**0.96**					
Sensitivity	**Average**	**0.78**	**0.67**	**0.29**	**0.57**	**0.85**	**Average**	**0.60**	**0.53**	**0.62**	**0.70**
Specificity		**0.97**	**0.98**	**0.75**	**0.97**	**0.96**		**0.97**	**0.96**	**0.96**	**0.95**
Difference		**7 / -2**	**18 / -2**	**56 / 21**	**28 / -1**	**-**		**10 / -2**	**17 / -1**	**8 / -1**	**-**
MCC		0.71	0.63	0.05	0.54	0.75		0.55	0.44	0.55	0.63
Accuracy		0.86	0.80	0.53	0.75	0.89		0.80	0.73	0.80	0.85

Additional Information 5:1: *16S rRNA Variability Across the Different Taxonomic Families.*

Illustration of the different levels of conservation within the 16S rRNA genes compared to the prokaryotic families, at this link (https://1drv.ms/b/s!AuqBgNz38MzDmAWdPajCAbG1QO57).

Additional Information 5:2: *Bit-scores of 16S rRNA gene's Variable Regions*

Three or four prokaryotic families (from the same class) are plotted together for each variable regions, where the X axis shows the position within the 16S rRNA gene, and the Y axis shows the reciprocal alignment bit-score, where 2 reflects the most variable and 0 reflects the most conserved position.



Additional Information 5:3: *Heatmap of Distances Between Various Regions of the 16S rRNA Gene*

Heatmap illustrating the distances between various regions for each family defined via applying a moving window of a length 300 or 500 bps, can be accessed via this link (https://1drv.ms/b/s!AuqBgNz38MzDmAaG54HcYdFJrQ1c).

Additional Information 5:4: *Bit-scores of 16S rRNA Gene's Variable Regions*

Illustration of the expected number of OTUs together with number of total, over-split, and over-merged OTUs obtained using both the 97% cut-off and our proposed lookup table for the 34 mock samples

Meta ID	Expected number of OTUs	Total Number of OTUs	# over-merged OTUs	# over-split OTUs	Total Number of OTUs	# over-merged OTUs	#over-split OTUs
		97% Cut-off			Lookup table		
35	51	46	7	2	52	3	4
36	47	44	5	2	48	3	4
37	50	45	7	2	49	3	2
38	45	43	5	3	47	3	5
42	46	43	5	2	45	4	3
43	46	43	5	2	45	3	2
44	46	43	5	2	45	3	2
45	46	42	5	1	44	3	1
46	46	42	5	1	43	4	1
47	44	38	6	0	40	4	0
48	40	37	5	2	40	3	3
49	41	39	5	3	46	3	8
59	48	48	5	5	52	4	8
60	49	49	5	5	51	3	5
61	48	48	5	5	49	3	4
66	43	41	5	3	46	3	6
74	44	42	5	3	45	3	4
75	46	44	5	3	47	3	4
76	45	43	5	3	47	3	5
77	43	37	6	0	38	5	0
80	46	42	5	1	43	4	1
81	45	41	5	1	43	3	1
85	46	42	5	1	43	4	1
86	47	43	6	2	46	3	2
87	47	43	6	2	46	3	2
88	49	44	7	2	48	3	2
89	46	39	7	0	42	5	1
90	46	39	7	0	41	5	0
91	46	39	7	0	41	5	0
98	43	38	5	0	39	4	0
99	43	37	6	0	39	5	1
100	43	38	5	0	39	4	0
101	43	37	6	0	38	5	0
102	43	37	6	0	38	5	0
Average	**46**	**42**	**6**	**2**	**44**	**4**	**2**

Additional Information 6:1: *Percentage of Reads Removed*

Table illustrating the percentage of reads removed by each pipeline throughout the various samples.

Percentage of rejected reads					
Sample ID		QIIME	MOTHUR	USEARCH	OCTOPUS
MOCK1	130403(**V34)**	19	16	94	17
	130417(**V34)**	18	17	95	17
	130422(**V34)**	15	17	95	5
	130403(**V4)**	2	4	15	5
	130417(**V4)**	3	3	15	5
	130422(**V4)**	3	3	8	5
MOCK2	v4.I.1	10	19	20	15
	v4.I.05	13	22	22	17
	v4.v5.1	27	48	73	47
	v4.v5.I.11	29	43	46	45
MOCK 3	M1(**V34)**	54	47	43	45
	M2(**V34)**	56	48	44	47
	M3(**V34)**	87	49	43	46
Average		**26**	**26**	**47**	**24**

Additional Information 6:2: *Number of OTUs*

Number of OTUs per sample after being processed via the various pipelines.

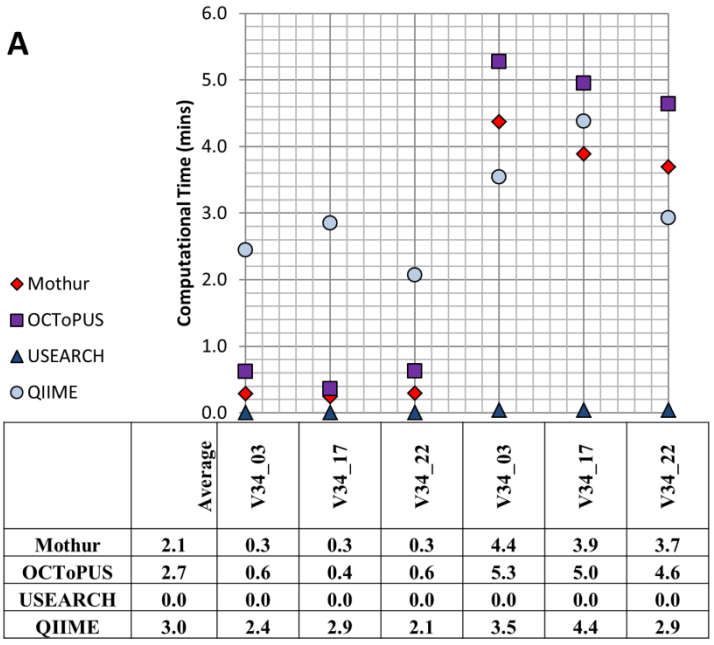
	Number of OTUs												
	MOCK1						MOCK2				MOCK3		
	V34-130403	V34-130417	V34-130422	V4-130403	V4-130417	V4-130422	V4.I.1	V4.I.05	V4.V5.1	V4.V5.I.11	**M1**	**M2**	**M3**
USEARCH	16	16	15	469	351	347	44	47	18	25	34	52	28
MOTHUR	83	61	97	697	538	478	196	302	84	590	58	74	37
OCTOPUS	41	31	49	287	251	174	26	24	29	48	36	52	27
QIIME	59	53	64	575	478	465	380	521	85	1248	50	65	21

Additional Information 6:3: *Number of OTUs per Species*

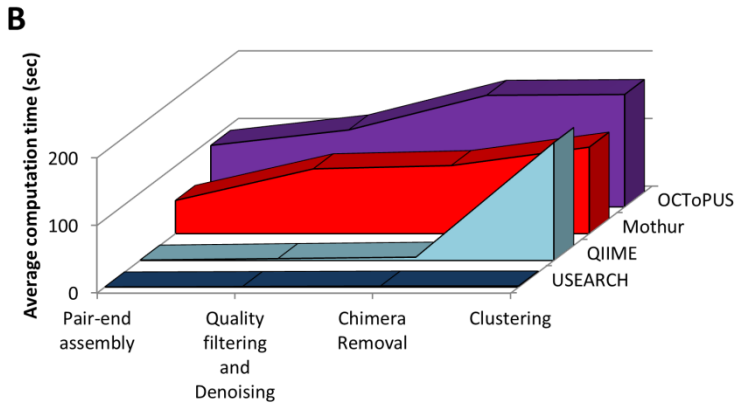
Table showing the number of OTUs per species within each sample, as well as the average number of OTUs per species to illustrate the over-splitting phenomenon among the various pipelines, accessible via this link (https://1drv.ms/b/s!Ark4l8D69K0UgikvMvlRRYx0euBL).

Additional Information 6:4: *Computational Cost*

Plot illustrating the computational time (in minutes) of MOCK1 samples for the three various pipelines (A), and the average computational time (in seconds) for the different steps within each pipeline (B).



	Average	V34_03	V34_17	V34_22	V34_03	V34_17	V34_22
Mothur	**2.1**	**0.3**	**0.3**	**0.3**	**4.4**	**3.9**	**3.7**
OCToPUS	**2.7**	**0.6**	**0.4**	**0.6**	**5.3**	**5.0**	**4.6**
USEARCH	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**
QIIME	**3.0**	**2.4**	**2.9**	**2.1**	**3.5**	**4.4**	**2.9**



APPENDIX B
CONTRIBUTIONS

BN Experiment: Intermediate Results of the Microbiological Analyses

Moors H, Cherkouk A., Mysara M, Bleyen N., Boven P, Selenska-Pobell S, Leys N, Valcke E

Mont Terri consortium (2013). 68 p (Technical Note)

Abstract: The Bitumen-Nitrate-Clay interaction (BN) experiment aims to clarify the biochemical, chemical and microbiological processes that could potentially occur by disposal of Bituminized Intermediate-level radioactive waste. The physico-chemical observations correlate perfectly with the observed shifts in bacterial populations. Under normal supposed unperturbated borehole solution conditions, the most abundant microbes detected by genetic analyses are oligotrophic heterotrophs. As soon as an electron acceptor is present, which renders more chemical redox energy than the naturally present sulphate an overwhelming shift appears to nitrate reducing bacteria. If in parallel a carbon source is offered the community composition does not alter that much, but the speed of nitrate reduction is increased tenfold. Beside denitrifying prokaryotes, every interval community harbours bacterial species which are capable of fixing N2 gas and reducing the N2 gas further into its most reduced form, ammonia. Molecular biology analysis also shows each interval harbours an autotrophic population, capable of converting inorganic carbon molecules into organic carbon molecules. The latter can be used as carbon source for organotrophs. As soon as the energetic carbon source is used up or the easily accessible electron acceptor is exhausted, the microbial communities of the intervals gradually shift towards a normal residing community.

Contribution: For this technical report, I helped in the data processing, statistical analysis and visualization of the results. The project includes data analysed by either Sanger or 454 Pyrosequencing platforms. A tailored bioinformatics analysis pipeline for Sanger dataset and modified mothur pipeline for 454 dataset including our in house developed tools (CATCh and NoDe) was applied in this study to obtain reproducible and high quality sequence data. The bioinformatics pipeline consists of six consecutive *pre-processing* steps. First, all primer– and vector sequences were removed with an in-house developed script. In a second step, a quality improvement step, the single nucleotide errors generated by the automated sequence analysis were remove, by using a sliding window approach (applied window/lens width = 100 bp) with the average Phred quality score having a minimum of 25 and 30 for Sanger and 454 sequencing data respectively. In the third step an alignment was performed against SILVA-based bacterial reference database. In a fourth step, the alignment result was screened for outliers. The fifth step is the denoising step using *pre.cluster* command for Sanger sequencing reads and NoDe algorithm for 454 sequencing reads. The sixth and last step of our data cleanup pipeline is the removal of almost all chimeric sequences using UCHIME for Sanger and CATCh for 454 sequencing data. By following this six-step sequence data analysis pipeline, a large number of false sequences, often wrongly identified as novel or rare species, could be removed. For optimisation of the pipeline described above a MOCK community of 18 species was prepared to act as a control, where its composition is well known. The overall data analyse error rate was calculated to be 0.7% however after applying our preprocessing approach the error rate was reduced to 0.17%. This is leading to a lower number of reads however with a high level of accuracy. Finally, I wrote a part of the report covering the methods I used and performed all revisions.

Snow surface microbiome on the High Antarctic Plateau (DOME C)

Michaud L, Lo Giudice A, Mysara M, Monsieurs P, Raffa C, Leys N, Amalfitano S, Van Houdt R.

PLoS One (2014) 9:e104505

Abstract: The cryosphere is an integral part of the global climate system and one of the major habitable ecosystems of Earth's biosphere. These permanently frozen environments harbour diverse, viable and metabolically active microbial populations that represent almost all the major phylogenetic groups. In this study, we investigated the microbial diversity in the surface snow surrounding the Concordia Research Station on the High Antarctic Plateau through a polyphasic approach, including direct prokaryotic quantification by flow cytometry and catalyzed reporter deposition fluorescence *in situ* hybridization (CARD-FISH), and phylogenetic identification by 16S RNA gene clone library sequencing and 454 16S amplicon pyrosequencing. Although the microbial abundance was low (<103 cells/ml of snowmelt), concordant results were obtained with the different techniques. The microbial community was mainly composed of members of the *Alpha-proteobacteria* class (e.g. *Kiloniellaceae* and*Rhodobacteraceae*), which is one of the most well-represented bacterial groups in marine habitats, *Bacteroidetes* (e.g. *Cryomorphaceae* and *Flavobacteriaceae*) and *Cyanobacteria*. Based on our results, polar microorganisms could not only be considered as deposited airborne particles, but as an active component of the snowpack ecology of the High Antarctic Plateau.

Contribution: For this work, I helped in the data processing, statistical analysis and visualization of the results. Reads produced by Sanger sequencing and 454 pyrosequencing were primer and barcode clipped. Reads were trimmed via removal of read with length lower than 200 nts, containing one or more ambiguous bases, or having a homopolymer longer than 8 (for pyrosequencing data). Next, a quality trimming approach was applied using a sliding window of 100 nt length with a cut-off on the Phred score of 30 and 20 for pyrosequencing and Sanger reads, respectively. For Sanger, we used the TraceTuner software in order to convert the quality files in Phred score. All reads were aligned to the Silva reference database, thereby removing reads starting extremely late, or ending extremely early in the alignment using the mothur filter and screen commands. Pyrosequencing reads were denoised using NoDe, while the Sanger sequencing data were denoised using the SLP algorithm as implemented in mothur. Chimera removal and operational taxonomic unit (OTU) clustering of the pyrosequencing reads was performed using UPARSE. For the Sanger sequencing data we used the UCHIME chimera detection tool together with nearest neighbour clustering algorithm implemented in mothur. Phylotyping was performed using the RDP classifier retrained with the greengenes dataset. For each phylotype the most representative read was selected by calculating the pairwise distance between all reads in a single Phylotype and selecting the read with the smallest maximum distance to all other reads (mothur *get.oturep* command). For the pyrosequencing reads, only the 15 most abundant phylotypes were selected in order to improve readability of the tree. The closest relatives of these phylotypes were obtained using the RDP classifier, with the RDP cultured and the greengenes dataset. The full set of sequences were aligned against the SILVA database and the resulting alignments were cleaned (using the *align.seqs* and *filter.seqs* commands). The distance was calculated and parsed for tree creation using mothur (*dist.seqs* and *clearcut* commands). Visualization of the phylogenetic tree was done using FigTree. Finally, I wrote the part of the manuscript covering my contribution and revised the manuscript.

Measuring the Biodiversity of Microbial Communities by Flow Cytometry.

Props R, Monsieurs P, Mysara M, Clement L, Boon N. (2016).

Methods in Ecology and Evolution.

Abstract: Measuring the microbial diversity in natural and engineered environments is important for ecosystem characterization, ecosystem monitoring and hypothesis testing. Although the conventional assessment through single marker gene surveys has resulted in major advances, the complete procedure remains slow (i.e., weeks to months), labour-intensive and susceptible to multiple sources of laboratory and data processing bias. Growing interest, in highly resolved, temporal surveys of microbial diversity, necessitates rapid, inexpensive and robust analytical platforms that require limited computational effort. Here, we demonstrate that sensitive single-cell measurements of phenotypic attributes, obtained via flow cytometry, can provide fast (i.e., within minutes) first-line assessments of microbial diversity dynamics, without demanding extensive sample preparation and downstream data processing. We developed a data processing pipeline that fits bivariate kernel density functions to phenotypic parameter combinations of an entire microbial community, and concatenates them to a single one-dimensional phenotypic fingerprint. By calculating established diversity metrics from such phenotypic fingerprints, we construct an alternative interpretation of the microbial diversity that incorporates distinct phenotypic traits underlying cell-to-cell heterogeneity (i.e., morphology and nucleic acid content). Based on a detailed longitudinal study of a highly dynamic microbial ecosystem, our approach delivered temporal alpha diversity profiles that strongly correlated with the reference diversity, as estimated by 16S rRNA amplicon sequencing. This strongly suggests that the distribution of a limited amount of phenotypic features

within a microbial community already provides sufficient resolving power for the measurement of diversity dynamics at the species level. We present a fast, robust analysis method for monitoring the microbial biodiversity of natural and engineered ecosystems that correlates well with the conventional marker gene surveys. Our work has both applied and fundamental implications that stretch from ecosystem monitoring and studies on microbial community dynamics, to supervised sampling strategies. Furthermore, our approach offers perspectives for the development of on-line and in situ monitoring systems for microbial ecosystems.

Contribution: Parallel to flow cytometric analysis, high-throughput amplicon sequencing of the V3-V4 hypervariable region was performed with the Illumina MiSeq platform. This generated 50,000 raw 300 bp paired-end reads per sample. Contigs were created by merging paired-end reads based on the Phred quality score heuristic as described in www.mothur.org/wiki/MiSeq_SOP using mothur. Contigs were aligned to the SILVA database, and filtered from those with (i) ambiguous bases, (ii) more than 8 homopolymers, and (iii) those not corresponding to the V3–V4 region. The aligned sequences were filtered and dereplicated while sequencing errors were removed using IPED algorithm. Chimera removal was performed with the CATCh tool set in *de novo* mode. Sequences were clustered into operational taxonomic units with UPARSE on default settings. Quality of the sequencing and post-processing pipeline was verified by incorporating triplicate mock samples (12 species) into the same sequencing run. Finally, all samples were subsampled down to the minimum sequencing depth provided to avoid sample size dependencies in the diversity assessment. Alpha diversities were directly calculated on the OTU abundance matrix while inference on beta diversity was based on weighted UniFrac distances, generated in QIIME. The phylogenetic tree required to calculate the UniFrac distance was built by the FastTree method using the OTU representative sequences. I wrote the part of the manuscript covering my contribution and revised the manuscript.

Microbial Biomarkers For Radiation-Induced Oral Mucositis

De Ryck T, Kerckhof F.M, Mysara M, Monsieurs P, Duprez F, Vanhoecke B, Van de Wiele T.

Head And Neck Cancer Patients (2016), Oral Oncology (submitted).

Abstract: Nowadays, researchers investigate host-microbe interactions to gain new insights in the pathobiology of different diseases in search for new treatment strategies. Also in case of oral mucositis, a severe side effect of cancer therapy, interest in the potential role of the oral microbiota is growing. In this study, oral microbial changes in cancer patients treated with radiotherapy were further explored. Samples from the buccal and tongue microbiota were taken in head and neck cancer patients treated with radiotherapy. By use of Illumina sequencing analysis, the shifts in oral microbiota during therapy were investigated and potential links with the severity of mucositis were investigated. The relative abundances of Bacteroidetes and Fusobacteria spp. colonizing the buccal mucosa increased due to radiotherapy, while the tongue microbiota appeared to be more stable. Furthermore, microbiota like Bifidobacterium, Fusobacterium, Peptostreptococcus, Porhyromonas and Prevotella spp. were shown to increase with the severity of oral mucositis. Moreover, we were able to identify certain microbiota like Atopobium spp. which were significantly higher in the oral cavity of patients that would develop severe mucositis, pointing to its' potential use as a marker for severe mucositis.

Contribution: DNA samples were sequenced by LGC genomics (Berlin, Germany). 300 bp paired-end reads (Illumina MiSeq V3) of the bacteria 16S DNA were obtained covering the V3-V4 region. The sequences were processed following (http://www.mothur.org/wiki/MiSeq_SOP) with the phylotype approach to obtain the OTU-table. The reference alignments were SILVA-based and the RDP trainset was used to classify the sequences. Sequences were assembled via the *make.contigs* command. Contigs containing ambiguity and/or with unexpected length were removed via the *screen.seqs* command, followed by dereplication via the *unique.seqs* command. Next sequences were aligned via *align.seqs* command using the SILVA-based reference database. To make sure that the amplicons are corresponding to the same region, the outliers were removed via the *screen.seqs* command, as well as the contigs having homopolymers with a length higher than eight. The alignment was simplified via the *filter.seqs* command and sequences were denoised via the *pre.cluster* command, followed by chimera removal (*chimera.uchime* in *de novo* mode) and classified using the RDP classifier trained on 16S rRNA gene trainingset, thereby removing those sequences classified as Eukaryotes or Archaea. Phylotyping was performed via the *phylotype* command. Three mock samples were used to validate the results, and fine-tune the processing parameters. I also provided comparative analysis comparing the result via applying my developed algorithms with other existing pipelines (mothur and UPARSE). Finally, I wrote the part of the manuscript covering my contribution and revised the manuscript.

Plants Rather than Fertilizer Drive Rhizosphere Bacterial Community in Organic Growing Medium Blended with Recovered Nutrients

Grunert O, Robles A.A, Aguilar R, Reheul D, et al

2016 (in preparation).

Abstract:

Intensive agriculture and horticulture depend on fertilizer input to sustain food production. However, synthetic fertilizer production is associated with a high carbon footprint and pollution. The use of recovered nutrients, such as organic fertilizers and struvite are promising green fertilizers to contribute to a more sustainable crop production in combination with organic growing media. These recovered nutrients can replace the common used compound fertilizers in growing media, mostly containing guaranteed contents of N, P, and K being the elements most widely applied in greenhouse industry. So far, very little is known regarding the effects of different types of recovered nutrients on the microbial community in the roots and bulk zone and its activity. We hypothesized that recovered nutrients, such as an organic fertilizer and ammonium struvite can be used as novel plant compound fertilizers and that the plant microbiome composition is affected by agricultural practices such as fertilization and by plant in an organic growing medium and that the activity and abundance of the ammonia oxidizing bacteria is increased in the rhizosphere. In this study, young lupine, known for its ability to fix nitrogen in symbiosis with bacteria and tomato chosen for its high root exudation capacity, played a significant role on the structure of the rhizosphere microbial community, in organic growing medium with high N input. The use of struvite as N fertilizer is promising, however the blend needs

to be balanced with a surplus of nitrate. Multivariate analysis showed plants rather than fertilizer drive rhizosphere bacterial community in organic growing medium blended with recovered nutrients. Ammonia oxidation rates in the rhizosphere were decreased in comparison with those in the bulk zone in combination with tomato plants. Moreover, the microbial community inhabiting the tomato rhizosphere seems to be unique. Fertilizer type and the interaction between plant and fertilizer type impacted species richness, diversity, and evenness measured at the second harvest. This study reveals that the interplay between the plant and the microbiome in the growing medium drives functional microbial communities and impacts nitrification dynamics.

Contribution:

High-throughput amplicon sequencing of the V3 – V4 hypervariable region was performed with the Illumina MiSeq platform. Contigs were created by merging paired-end reads based on the Phred quality score (of both reads) heuristic as in Mothur SOP. Contigs were aligned to the SILVA database and filtered from those with (i) ambiguous bases, (ii) more than 8 homopolymers, and (iii) those not corresponding to the V3–V4 region, which resulted in a removal of 75 % of the sequences. The sequencing errors were removed using IPED, Chimera removal was performed using the CATCh tool in *de novo* mode, which resulted in another removal of 3% of the sequences. Sequences were classified using the RDP trainset(Cole et al. 2007)(Cole et al. 2007)(Cole et al. 2007), removing those with unexpected/undesirable classification as Eukaryota, Mitochondria or Chloroplast classification. The sequences were clustered into operational taxonomic units at 97% identity level with UPARSE on default settings via the *sortbysize*, *cluster_otus*, *and usearch_global* commands. Quality of the sequencing and post-processing pipeline was verified by mock samples (12 species) in triplicate into the same sequencing run. The resulting OTU table were used to perform various biodiversity analysis. The pipeline applied here was further modified to produce OCToPUS.

APPENDIX C
CURRICULUM VITAE

Mohamed Mysara Ahmed

Personal Data

Born	17th Feberuray 1986
Phone	+32 484 78 86 35
E-mail	mohamed.mysara@gmail.com

Education & Scientific career

2012 – Present	Joint PhD student
Thesis:	From Sequencing Reads to Microbial Diversity: Bioinformatic Algorithms For Processing Amplicon Sequencing Data
Promotor:	Prof. Dr. Danial. Charlier, Prof. Jeroen Raes, Dr Pieter Monsieurs
Entities:	Vrije Universiteit Brussel, department of Bio-Engineering Sciences, KU Leuven, Microbiology and Immunology and the Belgian Nuclear Research Centre (SCK.CEN), Expertise Group Molecular and Cellular Biology, Microbiology Unit.
2011 –2012	Junior Scientist
Promotor:	Dr. Mohamed Abouelhoda
Entity	Nile University– Bioinformatics research group (Egypt)
2010 –2011	Assistant Researcher
Promotor:	Dr. Mahmoud ElHefnawi
Entity	National Research Center, Bioinformatics group (Egypt)
2009 –2010	Master in Information Technology
Thesis:	MysiRNA: Automation of siRNA Design Considering Multi-score Filtration
Promotor:	Prof. John. Garibaldi
Entities:	University of Nottingham, School of computer science (UK)
2008 –2009	Postgraduate Diploma in Clinical Pharmacology
Entity	Ain Shams University, Faculty of Pharmacy (Egypt)
2003 –2008	Bachelor of Science.in Pharmaceutical Science
Entity	Cairo University, Faculty of Pharmacy (Egypt)

Publications

Grunert O, Robles A.A, Aguilar R, Reheul D, **Mysara M**, et al (2016) *Plants rather than fertilizer drive rhizosphere bacterial community in organic growing medium blended with recovered nutrients* (in preparation).

Mysara M, Njima M, Leys N, Raes J, Monsieurs P. (2016). *From Reads to Operational Taxonomic Units: Comparative Analysis of Amplicon Sequences Processing pipelines*, GigaScience (submitted)

Mysara M, Vandamme P, Leys N, Raes J, Monsieurs P. (2016). *Reconciliation between Operational Taxonomic Units and Species Boundaries.* FEMS Microbiology Ecology (submitted).

Props R, Monsieurs P, **Mysara M**, Clement L, Boon N. (2016). *Measuring the Biodiversity of Microbial Communities by Flow Cytometry.* Methods in Ecology and Evolution.

De Ryck T, Kerckhof F.M, **Mysara M**, Monsieurs P, Duprez F, Vanhoecke B, Van de Wiele T. (2016) *Microbial Biomarkers For Radiation-induced Oral Mucositis In Head And Neck Cancer Patients*, Oral Oncology (submitted).

Mysara M, Leys N, Raes J, Monsieurs P. (2016). *IPED: a highly efficient denoising tool for Illumina MiSeq Paired-end 16S rRNA gene amplicon sequencing data.* BMC Bioinformatics 17:192.

Mysara M, Leys N, Raes J, Monsieurs P. (2015). *NoDe: a fast error-correction algorithm for pyrosequencing amplicon reads.* BMC Bioinformatics 16:88.

Mysara M, Saeys Y, Leys N, Raes J, Monsieurs P. (2015). *CATCh, an ensemble classifier for chimera detection in 16S rRNA sequencing studies.* Appl. Environ. Microbiol. 81:1573–84

Michaud L, Giudice A, **Mysara M**, Monsieurs P, Raffa C, Leys N, et al- *Snow Surface Microbiome on the High Antarctic Plateau (DOME C).*-PLOS one, 9:8(2014).- ISSN 1932-6203

Moors H, Cherkouk A., **Mysara M**, Bleyen N., Boven P, Selenska-Pobell S, et al (2013).*BN Experiment: Intermediate results of the microbiological analyses*, Belgium: Mont Terri consortium. 68 p(Technical Note)

Mysara M, ElHefnawi M, Garibaldi J. (2012). *MysiRNA: Improving siRNA Efficacy Prediction by a Machine-learning Model Combining Multi-tools and Whole Stacking Energy (*ΔG*),* Journal of Biomedical Informatics, pp. 528-534. doi: 10.1016/j.jbi.2012.02.005.

Mysara M, Garibaldi J, ElHefnawi M. (2012). *A Workflow for Efficient siRNA Design*. PLoS ONE e25642. doi:10.1371.l.

ElHefnawi M and **Mysara M** (Ed),(2012), *Recurrent Neural Networks and Soft Computing*, InTech (Book Ed.).

ElHefnawi M and **Mysara M** (2011), *In-silico Approaches for RNAi Post-Transcriptional Gene Regulation: Optimizing siRNA Design and Selection, Bioinformatics* - Trends and Methodologies, Mahdavi M (Ed.), InTech. (Book Chapter)

Oral presentations

Mysara M., Raes J., Leys N., Monsieurs P.- *Bioinformatics Tools for Accurate Analysis of Amplicon Sequencing Data for Biodiversity Analysis*.- Benelux Bioinformatics Conference (BBC2015).- Antwerp, Belgium, 7-8 Dec 2015.-

Mysara M., Raes J., Leys N., Monsieurs P.- From Reads to OTUs: *Improved Algorithms for Preprocessing Amplicon Sequencing data*.- Recent Computational Advances in Metagenomics (RCAM'15).- Paris, France, 6 Oct 2015.-

Mysara M., Raes J., Leys N., Monsieurs P.- Bioinformatics *Algorithms to get accurate 16S rRNA high throughput sequencing data*.- RSG Belgium Symposium 2015 , Hasselt, Belgium, 28 May 2015.-

Mysara M., Wouters K., Saeys Y., Moors H., Leys N., *et.al*.- NoDe and CATCh: *Two Algorithms to Get More Accurate Metagenomics Results*.- "Microbial Diversity for Science and Industry". Brussels, Belgium, 26-27 November 2013.

Mysara M., Wouters K., Saeys Y., *et.al*.- *NoDe and CATCh: Two Algorithms to Get More Accurate Metagenomics Results*.- 2nd Thünen Symposium on Soil Metagenomics.- Braunschweig, Germany, 11-13 December 2013.

Mysara M., ElHefnawi M, Garibaldi J, "MysiRNA-Designer: *A workflow for Efficient siRNA Design Considering splice-variants, Multi-score filtration, Target Accessibility and both perfect and seed-matching off-targets*, RECOMB11..

Poster presentations

Mysara M., Vandamme P., Leys N., Raes J., Monsieurs P.- *Reconciliation between Operational Taxonomic Units and Species Boundaries*..- " Microbes and the Global Change " .- Brussels, Belgium, 11 Dec 2015.

Mysara M., Raes J., Leys N., Monsieurs P.- *Bioinformatics Algorithms to get accurate 16S rRNA high throughput sequencing data*.- FEMS 6th congress of European Microbiologists.- Maastricht, Netherlands, 7-11 June 2015.

Mysara M., Raes J., Leys N., Monsieurs P.- *Bioinformatics Algorithms to get accurate 16S rRNA high throughput sequencing data*.- Cell Signaling in Host-Microbe Interactions.- Brussels, Belgium, 18-18 November 2014.

Mysara M., Monsieurs P., Leys N., Raes J.- *CATCh: an ensemble classifier for chimera detection in 16S metagenomics studies*.- Netherlands Bioinformatics Conference 2013.- Lunteren, Netherlands, 16-17 April 2013.

Mysara M., Saeys Y., Leys N., Raes J., Monsieurs P.- *NODE and CATCH: two algorithms to get more accurate 16S rRNA sequencing data*.- Bioinformatics at all scales of life.- Brussels, Belgium, 9-10 December 2013.

Mysara M., ElHefnawi M, Garibaldi J, *MysiRNA: siRNA Prediction Model Combining Whole Stack Energy and Multiple-scoring Tools*, ISCB Conference, Feb 2011 South Africa.

Mentor of students

Shahin M. (2016)– *Impact of alignment algorithms on 16S rRNA gene amplicon sequencing data analysis*, Vrije Universiteit Brussel, Master thesis.

Ramakrishnan A.- *High throughput comparison between 16S metagenomics analysis pipelines*, KU Leuven, Master level apprenticeship: 1 July to 15 September (2015).

Njima M.-*Assessing and optimizing a bioinformatics pipeline for long reads 16S metagenomics*, Master level apprenticeship: 15 June to 15 December (2015).