Algorithms and Tools for Genome Assembly and Metagenome Analysis

Dissertation

der Fakultät für Informations- und Kognitionswissenschaften
der Eberhard-Karls-Universität Tübingen
zur Erlangung des Grades eines
Doktors der Naturwissenschaften
(Dr. rer. nat.)

vorgelegt von
Dipl.-Inform. Daniel C. Richter
aus Mülheim an der Ruhr

Tübingen
2009

Tag der mündlichen Qualifikation:	16.12.2009
Dekan:	Prof. Dr.-Ing. Oliver Kohlbacher
1. Berichterstatter:	Prof. Dr. Daniel H. Huson
2. Berichterstatter:	Prof. Dr. Stephan C. Schuster

Erklärung

Hiermit erkläre ich, daß ich diese Schrift selbständig und nur mit den angegebenen Hilfsmitteln angefertigt habe und daß alle Stellen, die im Wortlaut oder dem Sinne nach anderen Werken entnommen sind, durch Angaben der Quellen kenntlich gemacht sind. Eine detaillierte Abgrenzung meiner eigenen Leistungen von den Beiträgen meiner Kooperationspartner und von Implementierungsleistungen, die im Rahmen von mir betreuter Studien- und Diplomarbeiten erbracht worden sind, habe ich explizit in Anhang B vorgenommen.

Tübingen, September 2009 *Daniel Richter*

Zusammenfassung

Um einen umfassenden Einblick in die genetische Vielfalt und molekularbiologische Funktionalität eines Organismus zu bekommen, ist die Sequenzierung dessen Genoms unabdingbar. Allerdings erlaubt keine der gegenwärtigen Sequenziertechnologien, das gesamte Genom in einem einzigen Schritt "abzulesen". Stattdessen wird eine große Menge an kurzen Fragmenten (Reads) produziert, die um ein Vielfaches kürzer sind als das ursprüngliche Genom. Um letztendlich die vollständige Genomsequenz zu erhalten, werden die Reads mittels Algorithmen der Genomassemblierung möglichst optimal miteinander verknüpft. Die maschinelle Automatisierung der DNA-Sequenzierung basierte lange Zeit ausschließlich auf einer Methode, die in den siebziger Jahren von Frederick Sanger entwickelt wurde. Seit dem Jahr 2005 jedoch kommt eine neue Generation von Sequenziertechnologien auf den Markt, die es nun ermöglichen, in kürzerer Zeit eine große Menge Sequenzierdaten bei reduzierten Kosten zu produzieren. In dieser Arbeit werden verschiedene Methoden und deren Implementierungen vorgestellt, die solche Sequenzdaten verarbeiten und für die biologische Interpretation aufarbeiten.

Obwohl die neuen Sequenziertechnologien vielfältige Optimierungen versprechen, bleibt die Genomassemblierung eine ernstzunehmende Herausforderung für Bioinformatiker und Biologen. Eines der hier vorgestellten Programme ist OSLay. Es berechnet unter Einbeziehung eines verwandten Referenzgenoms sogenannte Scaffolds. Diese Scaffolds, eine definierte Menge von geordneten assemblierten DNA-Fragmenten, sind später hilfreich für die korrekte Zusammensetzung und somit auch für die abschließende Fertigstellung der Genomsequenz.

Der Einsatz von Hochdurchsatz-Technologien fördert die Erschließung und den Ausbau neuer molekular-biologischer Forschungsfelder. So profitiert zum Beispiel der junge Forschungszweig der Metagenomik stark von diesen neuen Entwicklungen. Dessen Schwerpunkt ist die genomischen Analyse von nicht-kultivierbaren mikrobiellen Organismen, die in diversen Habitaten (Biotopen) gefunden werden. In dieser Arbeit werden Methoden vorgestellt, die einerseits die Häufigkeitsverteilungen von Spezies visualisieren und die andererseits die Analyse mikrobieller Eigenschaften innerhalb eines Metagenoms ermöglichen. Hauptaugenmerk liegt jedoch auf einer neuartigen Me-

thode, die, basierend auf einer Homologiesuche, Reads mit Hilfe der Gene Ontology funktionell klassifiziert. Die intuitive Graphvisualisierung von GOAnalyzer ist Teil der MEGAN Software und erlaubt die effiziente Analyse von einem, sowie den Vergleich der gefundenen Genprodukte von mehreren metagenomischen Datensätzen.

Die sich rasant entwickelnden Sequenziertechnologien erfordern innovative Softwarelösungen, die die Hochdurchsatz-Daten nicht nur verarbeiten, sondern auch helfen, sie nutzbar machen. Um das Testen und Bewerten von Software zu erleichtern, wurde MetaSim, ein Simulationsprogramm für DNA-Sequenzen, entwickelt. Basierend auf einer Datenbank bekannter Genomsequenzen generiert MetaSim simulierte Readsequenzen, die parametrisierbaren Fehlermodellen unterliegen, welche die Fehlerraten und -typen bekannter Sequenziertechnologien widerspiegeln. Zusätzlich können Spezieshäufigkeiten festgelegt werden, um ganze Metagenome zu modellieren.

In dieser Arbeit werden neben OSLay, GOAnalyzer und MetaSim weitere Methoden und Erkenntnisse vorgestellt, die die Auswertung und Interpretation von genomischen und metagenomischen Datensätzen unterstützen.

Abstract

The sequencing of the genome is the first step to gain profound insights into the genetic diversity and the molecular-biological functions of an organism. The existing approaches to sequence DNA do not allow to "read" a whole genome sequence at once in a single step. Instead, many short fragments (reads) are produced that are actually orders of magnitude shorter than the original genome. To finally obtain the complete genome sequence, genome assemblers try to piece the reads back together. For a long time, the automatized and machine-based sequencing of DNA was dominated by an approach originally conceived by Frederick Sanger in the 1970s. Since 2005, several new ("next-generation") sequencing technologies appeared on the market that are able to generate much more sequencing data in shorter time and at lower costs compared to the Sanger sequencing. This thesis introduces several computational methodes that process and structure this sequencing data to assist in their biological analysis and interpretation.

Despite the improvements of the new sequencing technologies, genome assembly still poses serious challenges for (computational) biologists to obtain a finished genome sequence. In this work, a software (OSLay) is described that computes so-called scaffolds by ordering and sorting large fragments (contigs) of an unfinished genome assembly with regard to a related reference genome. The computed ordering of fragments later facilitates the successful completion of the final genome sequence.

The application of high-throughput technologies accelerates biological research and enables new sorts of large-scale genome investigations. One emerging research discipline that strongly benefits from these advancements is metagenomics. It is the study of uncultured microbial organisms directly derived from their natural environment. In this work, methods are presented to facilitate the visualization of species abundances and to enable the analysis of microbial properties of a metagenomic sample. Furthermore, a major focus is given to a novel homology-based approach for the functional annotation of metagenomic reads based on the Gene Ontology. Incorporated into the MEGAN software and provided with an intuitive graph visualization, the GOAnalyzer can be used to efficiently explore and compare the gene products of one or more metagenomic data sets.

The fast-evolving sequencing technologies demand for innovative soft-

ware concepts that are able to efficiently deal with high-throughput data. To support the testing and benchmarking of computational methods, a sequencing simulator software is introduced. Based on known genome sequences, MetaSim simulates sequencing reads that may serve as verifiable test data sets for any type of read processing software. The synthetic reads are generated according to adaptable error models reflecting the typical error characteristics of various sequencing technologies. Additionally, species abundance profiles can be determined to model realistic metagenome data sets.

Beside the introduction of OSLay, GOAnalyzer and MetaSim, additional methods and findings are presented in this thesis that support the analysis and interpretation of genomic and metagenomic data sets.

Acknowledgements

First and foremost, I want to thank my supervisor Prof. Dr. Daniel H. Huson for giving me the opportunity to explore fascinating research topics and for providing an excellent working environment. I would like to express my deep gratitude for his constant support, the kind advice and many constructive suggestions throughout my PhD study. I would also like to thank my co-advisor Prof. Dr. Stephan C. Schuster for his candid support and many inspiring and stimulating discussions, especially during my research stay at his research group at Penn State University, USA.

Thanks to current workmates at the *Algorithms in Bioinformatics* department for their companionship and for providing a friendly atmosphere, namely Alexander Auch, Marine Gaudefroy-Bergmann, Johannes Fischer, Juliane D. Klein, Suparna Mitra, Jan Schulze, Regula Rupp, and Andreas Szillus as well as other colleagues at the WSI: Magdalena Feldhahn, Holger Gast, Kay Nieselt and Julia Trieflinger among many others. I am especially glad that Alexander shared the room with me during the last year of my PhD. Numerous discussions and the exchange of ideas about scientific and non-scientific matters were very enriching and a great pleasure for me. Not to forget to thank former members of the department that made my early thesis life livable: Tobias Dezulian, Tobias Klöpper, and Christian Rausch.

I am also grateful to my collaborators for pleasant and fruitful cooperations: Felix Ott, Tilmann Weber, Ji Qi, and Fangqing Zhao.

Many, many thanks to my and Julia's family for their support and care throughout these years. And, of course, big thanks to my wife, Julia, who gave me endless encouragement, advice and the time that I needed. Thanks for being such a great source of strength during my studies.

And finally, but not least, special thanks go to my two little sons, Fabian and Malte, who reminded me every day that playing with LEGO bricks or toy cars is at least as important as bioinformatic software development.

I dedicate this thesis to the memory of my mother.

In accordance with the standard scientific protocol, I will use the personal pronoun “we” to indicate the reader and the writer or (as explained in Appendix B) my scientific collaborators and myself.

Contents

Chapter 1

Introduction

Every living organism on our planet possesses a genome that is composed of one or several DNA (deoxyribonucleotide acid) molecules determining the way the organism is built and maintained. The DNA molecules are divided into sets of discrete sequence stretches which encode for the genetic information that is translated into proteins.

Since 1970, researchers have made an effort to develop strategies to "read" the DNA sequence because the knowledge of an individual genome sequence provides deep insights into the biological functions and the evolutionary history of this organism. Moreover, a couple of diseases are caused by genetic disorders that are only diagnosable by studying sequence variations within certain regions on the DNA sequence. A widely applied strategy for DNA sequencing is the "chain-termination method" introduced by Frederick Sanger in 1977 (Sanger et al., 1977). The importance of this method over the last 20 years can be explained with its applicability to the machine-based automatization of DNA sequencing. This automatization step accelerated the development towards other high-throughput sequencing technologies, which eventually, opened the flood-gates to an extensive amount of sequence data.

Still, the sequencing of whole genomes turns out to be very challenging and time-consuming because only short sequences called reads (<1000 base pairs) can be produced. The *genome shotgun sequencing* approach, devised by Sanger (Sanger et al., 1982), deals with the problem to reconstruct longer sections of a genome: the idea is to sequence cloned and randomly sampled fragments of the genome to obtain short, overlapping reads. The random sampling and the cloning step ensures to cover each position of the original genome sequence with a sufficient number of reads. Using specialized algorithms, these reads are pieced back together to obtain the final genome sequence. Today, there exists a couple of genome assembly strategies and tools that try to resolve this "jigsaw-puzzle" problem efficiently. The genome shotgun strategy has already been successfully applied to many organisms, such as the bacterium *Haemophilus influenzae* (Fleischmann et al., 1995),

the fruit fly *Drosophila melanogaster* (Fleischmann et al., 1995) and even parts of the human genome (Venter et al., 2001).

However, due to some characteristics of the DNA (e.g. repeat regions) and some technical aspects, the assembly problem is far from being solved. For example, the complete reconstruction of a single contiguous sequence in the end of the assembly process sometimes fails and results in DNA fragments resembling a gapped genome sequence. To fill these gaps with sequence data, the fragments need to be ordered and orientated, i.e. they need to be set into the context of the original genome. Several procedures exist to obtain a consistent scaffold of fragments; one of them is presented in this thesis.

The Sanger technology dominated the sequencing market for several years. Eventually, since 2005, new ("next-generation") sequencing technologies have been commercially launched which promise to produce much more sequencing data in shorter time and at lower cost compared to the Sanger technology. The development of the 454, Genome Analyzer and SOLiD platforms (Margulies et al., 2005; Bentley, 2006; Shendure et al., 2005) was enabled by advancements in microfluidics, surface chemistry, and enzymology, and led to a significant increase of diverse (re-)sequencing projects. Also, for the first time, the ultimate goal, the cost-effective and standardized sequencing of human genomes, becomes likely to be accessible in the near future. Although the new technologies are able to produce significantly more data per sequencing run, the mentioned assembly problems still remain. Therefore, innovative algorithmic solutions are of great need that are capable of processing the new data flavors.

A relatively new research field, called metagenomics, also benefits from the advancements of the new technologies. As a major part of the free-living microbes (>99%) is assumed to elude the cultivation under laboratory conditions, new methods are required to enable the direct sequencing of organisms contained in environmental samples. In contrast to the Sanger sequencing, the emerging next-generation sequencing technologies avoid library preparation issues and therefore, enable researchers to generate unbiased sequencing data directly from the environment. However, the computational analysis of metagenomic data sets is a challenging task because the initial sequencer output consists of large volumes of short, anonymous reads, i.e. the species origin of a read is unknown. These reads are structured (optionally assembled) and taxonomically classified to gain knowledge about the complex species composition of the studied habitat.

Another focus is given to the functional analysis of the metagenome by detecting coding sequences (genes) on the DNA fragments. These genes are either annotated to known functions and gene products or classified as hypothetical proteins if no homologous sequence can be found in reference databases. Obviously, there is great hope to discover unique biosynthetic capabilities and pathways that are encoded in genomes of still unnoticed

microbes. For example, striking insights have already been obtained by studying different environments such as seawater (Rusch et al., 2007), soil (Daniel, 2005), air (Tringe et al., 2008) and various biofilms (Tyson et al., 2004). Furthermore, researchers apply metagenomic methods to understand the complexity of the human microbiome that is composed of a large number of microorganisms "whose collective genome contains at least 100 times as many genes as our genome" (Gill et al., 2006).

The resulting excess of data is remarkable: For example, the Global Ocean Sampling project sampled water probes in the Atlantic and Pacific and predicted about 6.21 million hypothetical proteins, almost doubling the number of known proteins present in databases at that time (Rusch et al., 2007). Hence, this example points out the need for novel algorithmic approaches and innovative software tools that allow the efficient organization and interpretation of metagenomic data sets.

Overall, the advent of next-generation sequencing technologies have had a strong bearing on several (new) research fields like the high-throughput sequencing of human genomes (personalized medicine), genome assembly, paleogenomics (Hofreiter, 2008) and metagenomics. It will certainly lead to further exciting discoveries and insights which, until recently, seemed to be hardly achievable. However, this promising era of "flowing" sequencing data actually poses a lot of challenges regarding the efficient handling and interpretation of the data.

This thesis describes several novel approaches and software tools that support the analysis, the interpretation, and the simulation of sequencing and metagenomic data.

Chapter 2 gives an introduction into the historical and theoretical background of DNA sequencing, assembly and metagenomics. It highlights the main historical developments towards modern sequencing technologies and its computational challenges regarding genome assembly. Furthermore, it includes an overview of representative metagenomic projects and typical analysis pipelines for the computational analysis of the taxonomical and functional content of an environmental sample.

Chapter 3 outlines an algorithmic approach addressing the scaffolding problem of a genome assembly. Here, an heuristic strategy is described that is able to detect the layout of unordered DNA fragments in reference to a closely-related genome sequence. This concept has been implemented as software tool called OSLay.

In chapter 4, the technical aspects of the MEGAN (MEtaGenome ANalyzer) software are presented. This software allows to organize and interpret complex metagenomic data sets. Special focus is given to a new algorithmic approach for the functional metagenome analysis aiming at the classification of reads according to Gene Ontology terms (Ashburner et al., 2000).

Many read processing tools for genome assembly or metagenomics are published these days, trying to keep pace with the fast-evolving sequencing

technolgies. To assist the efficient development, benchmarking and comparison of such software solutions, a new tool is described in Chapter 5. MetaSim provides functionality to generate verifiable test data sets based on known genome sequences. Reads can be simulated according to adaptable error models reflecting the typical error characteristics of different sequencing technologies.

Chapter 6 concludes the topics of this thesis and reviews the achievements of this work in the context of the current developments in the research fields of genomics and metagenomics.

Chapter 2

Background and Theory

In the following chapter, a few historic notes and fundamental aspects of genomics and metagenomics are outlined. Besides the description of the main biological principles, some technical and computational issues about DNA sequencing and assembly are covered as well. Of course, focus is primarily kept on concepts that are important for this thesis.

2.1 The Discovery of DNA

The discovery of DNA (deoxyribonucleic acid) is occasionally associated with two famous names: Watson and Crick. But their work, published in 1953 (Watson and Crick, 1953), was actually based on previous experiments and findings of other, rather unknown people. In fact, the first time that a person identified and isolated DNA, was many years before the 1950s. In 1869, Johann Friedrich Miescher, a Swiss doctor, conducted several experiments on pus found on wound dressings. His laboratory was part of the former kitchen in the castle of Tübingen, Germany (Dahm, 2008). He extracted a mysterious substance from white blood cells (leukocytes) which he later called *nuclein* because he assumed that this molecule is derived from the nucleus. Miescher could show that the properties of this substance differ significantly from proteins. But still, the common opinion to that time was that proteins are responsible for the inheritance of genetic information.

In 1919, Phoebus Levene, a Lithuanian biochemistrist, discovered the constituents parts of the DNA (Levene, 1919): the four organic nucleotides (bases adenine, guanine, cytosine and thymine), as well as the sugar and phosphate groups. He also proposed a chain structure in which the nucleotides are repeatedly connected by the phosphate groups. But still, the importance of the DNA was not apparent to that time. Later, the description of the DNA structure succeeded in the year 1953 when Francis Crick and James Watson published their work (Watson and Crick, 1953) based on x-ray analyses performed by Rosalind Franklin (Franklin and Gosling,

1953). They proposed a double-stranded DNA molecule which is shaped like a double-helix. Nine years later, Watson and Crick received the Nobel prize for this pioneering work (Franklin died of cancer aged 37 and could not be awarded posthumously.)

Based on these and many other studies, it could be shown that DNA is the carrier of the genetic information and that DNA can be inherited by the offspring of an organism. In eukaryotes (e.g. animals, plants or fungi), the DNA is located in the cell nucleus as linear chromosomes. By contrast, prokaryotes (bacteria or archaea, mostly uni-cellular) which lack a cell nucleus, contain the DNA, in most cases, as circular chromosome(s). The sequence of the four bases determines coding and non-coding stretches of DNA. In the process of transcription, the coding regions are transcribed into RNA (ribonucleic acid) which is then subsequently translated to gene products (proteins). The term "genome" means the full set of all inheritable genetic information of an organism coded in the DNA.

2.2 DNA Sequencing

The process of DNA sequencing is the (partial) reading of the base sequence of an organisms' genome. Determining the order of the nucleotides is crucial to get profound insights into the genetic diversity and organization of an organism. For example, by extracting the coding regions of the DNA (open reading frames, ORFs), the primary sequence of a protein (sequence of amino acids) can be determined which then provides information about the protein's functionality. The process of ORF finding and function assignment is also called gene prediction and annotation, respectively. Non-coding regions are occasionally responsible for the regulation of the expression of genes, i.e. regulatory proteins or transcription factors bind to these regions to control the synthesis of proteins. Another motivation for sequencing is the detection of mutations in the nucleotide sequence. The exchange or absence of bases may reveal genetic disorders which might lead to diseases. Further, based on the sequence similarity of multiple genomes (or specific stretches of sequence, also called marker genes), assumptions can be made concerning the evolutionary relationships between organisms. For example, the morphology-based classification models (e.g. in Haeckel (1866)) have difficulties to discriminate between analogy and homology (i.e. similarity due to evolutionary ancestry) when only comparing observed characteristic features of several organisms. So, the objectivity of sequence-based, phylogenetic analyses are able to refine the systematic classification of organisms based on morphology.

In general, one potential application for sequencing has significantly spurred the improvement of sequencing technologies: the cost-effective deciphering of the human genome, accessible and affordable for everyone. With

the personal genome sequence at hand, people anticipate to be able to accurately predict and even to cure diseases more efficiently in the future. The concept of such a *personalized medicine* fascinates researchers (as well as the pharmaceutical industry) and has led to the establishment of the Archon X Prize for Genomics. In 2006, the Archon foundation announced a $10 million prize for the team that decodes 100 human genomes in less than 10 days for less than $10,000 per genome (`http://genomics.xprize.org`). According to J. Craig Venter, the overall probability that this goal will be ever achieved is "close to 100%" (Pennisi, 2006).

To conclude this brief list of motivations for sequencing, it becomes clear that modern biology strongly relies on the extracted sequence of the nucleotides. In any case, sequencing represents the very starting point for subsequent molecular analyses that are fundamental for research studies in modern biology. It is the basis for understanding the wide range of molecular processes, the evolutionary classification of living organisms and the genetic differences that, e.g., makes us humans different from mice. Finally, being able to read the DNA sequence is indispensable for actually its "editing" or even "writing": DNA sequence modifications (cutting, copying and insertion of nucleotides) nowadays are standard techniques widely applied in molecular research.

The following subchapters outline the history of DNA sequencing and the main technologies, and protocols for the (high-throughput) data collection. Note that the field of emerging sequencing platforms is quickly moving, so the following list is only a snapshot of this particular time. The chronological introduction of the technological advancements may help to understand the tremendous increase of sequence data uploaded into the databases (Figure 2.1).

2.2.1 History of DNA Sequencing

In this section, a brief summary is given covering the developments of the early, manual approaches towards the first, automatic sequencing platforms that automated DNA sequencing. This overview is mainly based on a comprehensive survey (Hutchison, 2007) that characterizes all technological advances in more detail.

The description of the double-helical structure of DNA by Watson and Crick (Watson and Crick, 1953) was an important milestone, but, the first determination of the DNA sequence occurred almost fifteen years later. Reasons for this delay were for example the length of the DNA which is significantly longer than protein sequences or the high chemical similarity of the four bases. In contrast to amino acids whose chemical properties could be easily differentiated, the detection of the four nucleotides turned out to be challenging at that time. Another hurdle was the unavailability of DNAases, enzymes that cut DNA sequences at specific bases. In case of proteins, the

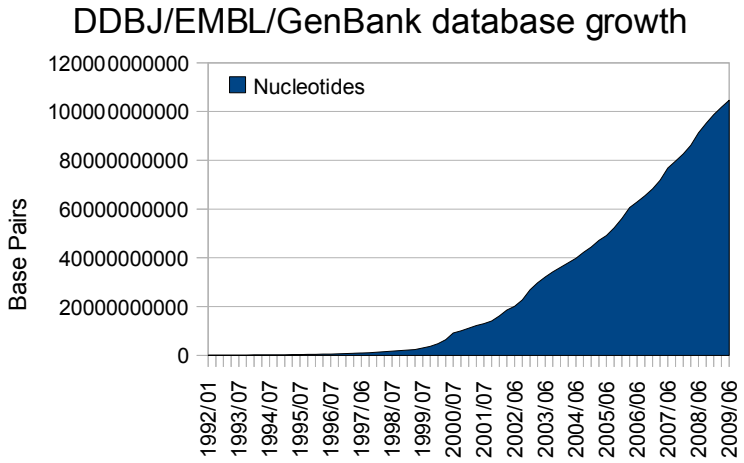


Figure 2.1: DDBJ/EMBL/GenBank database growth. At the beginning of 2009, the number of stored nucleotides passed the 100 Gbp mark. Data from `http://www.ddbj.nig.ac.jp/breakdown_stats/dbgrowth-e.html`.

precise cutting using proteases was already feasible, thereby enabling researchers to at least partially study protein sequences.

The first publication of a partial DNA sequence was accomplished by Wu and Kaiser (Wu and Kaiser, 1968). They first reported the partial sequencing of the phage lambda DNA in 1968, but the completed 12 base sequence was not presented until 1971 (Wu and Taylor, 1971). Eventually, the type II restriction enzymes were discovered by Hamilton Smith (Smith and Wilcox, 1970). These enzymes made it possible to cut large DNA molecules into smaller pieces which could be then analyzed more easily. The next milestone in DNA sequencing was the description of a novel approach by Sanger and Coulson in 1975, called the 'plus-and-minus' method (Sanger and Coulson, 1975). This work influenced many upcoming technological approaches in the following 30 years. For the first time, polyacrylamide gels were used to separate fluorescently (or radioactively) marked DNA-fragments by size using gel-electrophoresis. This method was relatively fast and simple and could read DNA stretches of length $\approx$50 bp (base pairs). However, only single-stranded DNA sequences could be sequenced and some difficulties arose when determining the length of homopolymers (single base repetitions). In February 1977, Maxam and Gilbert (Maxam and Gilbert, 1977) presented a similar method that even was applicable to double-stranded DNA and which produced fairly accurate sequences. Unfortunately, some of the chemicals used for this approach were toxic and hazardous to the health of the scientists.

Eventually, in December 1977, Sanger improved the 'plus-and-minus' method by reporting the 'dideoxy' approach (Sanger et al., 1977), also called the chain-termination method. Its success was based on the basic conception that later allowed the automatization in large-scale sequencing projects. The main idea is to determine the DNA sequence by an enzymatic elongation of complementary nucleotides. Some of these nucleotides cause the termination of the sequence synthesis at specific base positions.

First, the DNA fragment is cloned and a primer sequence is hybridized to one end of a single strand of the DNA molecule. Starting from the primer, the complementary strand is elongated by adding the enzyme DNA polymerase and the four deoxynucleotides (dATP, dGTP, dCTP and dTTP). The trick was then to include small amounts of dideoxynucleotides (ddNTP) that are similar to dNTPs except of lacking the 3'-OH group. Without the 3'-hydroxy group, the polymerase is not able to further elongate the DNA sequence, therefore the sequence synthesis is terminated at this position. Such, fragments are obtained that contain the primer and a stretch of nucleotides ending with a specific base. This is done for each of the four nucleotides and the results are displayed in four lanes on a polyacrylamid-gel by conducting a gel-electrophoresis separating the chains by their lengths. In contrast to the 'plus-and-minus' method, a band is visible for each nucleotide in a consecutive run of bases. To that time, a continuous DNA sequence of about 100 bp could be read. At the end of the 90s, the outcome of gel-based sequencing could be significantly improved by different labeling of nucleotides and by enhancements of the used gels. The development of 96-well plates led to sequence lengths up to 400 bp. Additionally, the amount of sequence data achievable by a single person per day grew up to 30 kbp.

To further increase the sequencing output, people started thinking about the machine-based automatization of the sequencing process. In 1986, the first report of automated sequencing came from a collaboration of Leroy Hood (California Institute of Technology) and Applied Biosystems (ABI) (Smith et al., 1986). Basically, a sequencing primer was fluorescently end labeled using four different dyes for the four sequencing reactions of the dideoxy method, respectively. The DNA samples then travel through a single gel. When reaching the level of the scanner unit, the fragments could be detected by these fluorescent markers. This data was then directly stored on computers that determined the base sequence of the reads (base-calling). In 1992, Craig Venter founded The Institute of Genomic Research (TIGR) which was later renamed as J. Craig Venter Institute in 2006 (JCVI, `http://www.jcvi.org`). Venter's plan was to establish "sequencing factories" with 30 ABI 373A automated sequencers to conduct parallel DNA sequencing. After sequencing large sets of expressed sequence tags (ESTs), TIGR completed several viral and organelle genomes and even the first bacterial genome sequences (*Haemophilus influenzae* and *Mycoplasma genitalium* (Fleischmann et al., 1995; Fraser et al., 1995)). With larger

Figure 2.2: ABI 3700 DNA Sequencer. Picture taken from `http://www.labcentraal.com`.

genome sizes, innovative algorithmic solutions were required for obtaining complete genome sequences. The whole genome shotgun (WGS) method (first applied in the *H. influenzae* genome project) and the application of the 'paired-end' strategy, had a wide influence on the future developments of sequencing technologies and upcoming assembler software. (The WGS method and other concepts of assemblies are outlined in more detail in Section 2.3). Upon these achievements, many genomes had been sequenced, for example *E. coli*, *S. cerivisiae*, *B. subtilis* and the first animal genome, the worm *C. elegans* (see Table 2.1).

After 1996, DNA sequencing was highly automated when new sequencing machines appeared on the market (ABI prism 310 and ABI prism 3700, Figure 2.2). The tedious work of pouring slab gels was replaced with capillaries which were filled with a polymer matrix. Such, samples could be automatically loaded from the 96-well plate and subsequently analyzed. The separation of bands became easier and, likewise, longer reads could be generated. The read length at 99% accuracy was between about 480 bp and 750 bp. In 1998, Venter, as head of the new company Celera Genomics, acquired 300 machines to approach the next big goal: the sequencing of the human genome (≈3 billion bp). The sequencing factories accomplished to sequence ≈50 Mbp per day. This high-throughput approach together with sophisticated assembly algorithms were first tested on the genome of the fruit fly (*Drosophila melanogaster*) (Myers et al., 2000). Eventually, after a competitive race with the Human Genome Project, a public consortium of sequencing centers from the United States, Europe and Japan, both teams presented the first, official draft of the human genome in 2001 (Venter et al., 2001; Lander et al., 2001). Considered as a huge milestone for humanity,

this event received much public attention.

However, the "biologists' hunger for even greater sequencing throughput" was still not satisfied (Schuster, 2008). The race for more and more sequence data has just begun (Figure 2.1. and Table 2.1) In the meantime,

Year	Origin	Mbp
1977	Phage *phi x174*	0.0054
1981	Human mitochondrium	0.0165
1982	Phage lambda	0.0485
1984	Epstein-Barr virus	0.172
1991	Human cytomegalovirus	0.237
1995	*Haemophilus influenzae*	1.83
1997	*Escherichia coli O157:H7*	4.6
1997	*Saccharomyces cerevisiae*	12
1998	*Caenorhabditis elegans*	97
2000	*Drosophila melanogaster*	120
2001	*Homo sapiens sapiens*	3,000

Table 2.1: Selection of sequenced genomes using Sanger sequencing. This listing of genomes indicates the amazing progress and improvement of sequencing technologies within 24 years.

the 'dideoxy method' (Sanger sequencing) which had been successfully employed for the last 30 years, has reached its limits. Although Sanger sequencing is still used, its days are likely to be numbered. Faster and more-efficient sequencing technologies promise to increase parallelism and throughput at reduced costs.

The next sections give an overview of the concepts of the so-called "next-generation-sequencing" technologies.

2.2.2 Next-Generation Sequencing Technologies

After dominating the last 30 years, Sanger sequencing is not able to keep pace with the ambitious plans in current genomic research. Although the performance of Sanger sequencing has been steadily improved over the years (e.g., read lengths up to 1000 bp), the costs for maintaining and running the machines are rather high ($\approx$ \$500 for 1 Mbp). Moreover, the system could only produce 50-100 Kbp per run. These numbers obviously indicate that high-throughput projects like, for example, large-scale resequencing projects or the decipherment of multiple human genomes can not be easily accomplished. However, the Sanger technology is still used in sequencing laboratories if rather high sequence accuracy is required instead of large volumes of sequence data.

Beginning in 2005, several new sequencing technologies appeared on the market that promise to revolutionize genomic studies. They all appeared in the post-Sanger era and are described as "Next-generation sequencing" (NGS) technologies. Other innovative systems, already announced for the following years but still under development, are called "3rd generation" or "Next-next-generation sequencing" (NNGS) technologies. The primary goal of all new methods is the reduction of costs while dramatically increasing the output of sequence data per run. With new developments in chemistry and image based processing methods, the reading of the DNA sequence has been massively parallelized to gain large volumes of data. These achievements opened the door for new genomic approaches in molecular biology and environmental studies (e.g. metagenomics). It is even anticipated that the new sequencing technologies will become part of every-day medical practice (personalized medicine). As a consequence of the low sequencing costs (e.g. due to a reduced reagent volume), large-scale sequencing projects can now even be conducted by small research institutes in addition to the major sequencing centers like the Wellcome Trust Sanger Institute (`http://www.sanger.ac.uk`) or the DOE Joint Genome Institute (`http://www.jgi.doe.gov`). Although major sequencing centers are still contributing a considerable amount of sequences to the databases, their influence becomes less dominant. Hence, in Marguerat et al. (2008), the authors state that NGS technologies have the capability to "democratize science".

Although the NGS technologies rely on different principles and biochemistries, they share some important characteristics. For example, the cloning step of the source sequence is done *in vitro*, to avoid any cloning bias (Sorek et al., 2007). Moreover, in contrast to the Sanger method, all technologies produce (ultra-)short reads (35-500 bp) which pose difficulties for subsequent analyses (for instance regarding genome assembly). This drawback is fairly compensated with an unprecedented scale of produced sequence data which leads to a high sequence coverage (i.e., the average number of overlapping reads per base position). The large output of read sequences is achieved by attaching millions of DNA fragments onto chips. During the sequencing phase, the fixed fragments are then "read" in parallel based on imaging data generated within each iterative cycle. Section 2.2.2 will introduce the concepts of three NGS technologies in more detail.

According to MacLean et al. (2009), some of the (new) applications for NGS technologies are for example:

- *de novo* genome assembly (e.g. Farrer et al. (2009))
- Assembly by alignment (resequencing)
- Gene expression analysis (transcriptomics)
- SNP (single nucleotide polymorphism) detection

- Resequencing of individuals (e.g. the diploid "Watson" genome reported in Wheeler et al. (2008a))
- Non-coding RNA characterization
- Identification of protein binding sites (e.g. Jothi et al. (2008))
- Metagenomics (more details on this is presented in Section 2.4)

Besides the ordinary approaches, some new applications became feasible or could be improved. For example, the gene expression analysis or RNA-centered transcriptomic approach may outperform the standard microarray in the near future. Instead of measuring the fluorescent spots that may be biased to a certain extent, only the reads have to be counted to infer how much of a particular RNA molecule is in a sample. Moreover, the high sequence coverage significantly increases the accuracy for SNP detection and resequencing projects because sequencing errors can be identified more easily. Eventually, the research field of environmental genomics or metagenomics strongly benefits from these technologies because the library construction and the cloning-host bias known from traditional Sanger sequencing is avoided (Schuster, 2008). This issue also applies to the analysis of ancient DNA of, for instance, mammoth DNA, the mitochondrial genome sequence of neanderthals, or the extinct Tasmanian tiger (Poinar et al., 2006; Green et al., 2008; Miller et al., 2009a).

Still, the ultimate goal and driving force for the steady improvement of NGS technologies is the high-throughput sequencing of the human genome. Table 2.2 indicates how the costs decreased since 2001. At last, the company that will be able to win the Archon X Prize (see Section 2.2) has a valuable sales argument.

To obtain an overview of the main differences, Table 2.3 shows a comparison of Sanger sequencing and the four major NGS contenders. The NGS technologies produce much more sequencing data for less dollars, whereas the advantage of Sanger sequencing is based on the long read length and the high read accuracy (up to 99,999%). However, once the NNGS technologies will hit the market within the next five to ten years, the benefit of long read sequences will not be uniquely subjected to the Sanger sequencing anymore. In the next subsections, the three NGS technologies are introduced.

454 (Roche)

In 2005, the 454 platform was the first next-generation sequencing technology commercially available on the market. The sequencing-by-synthesis approach (Margulies et al., 2005) introduced by the company 454 Life Science (Branford, CT, USA) (now marketed by Roche Applied Science) comprises several pioneering solutions for the library and template preparations and for sequencing. These approaches have later been partly adopted by other

Technology	Costs ($Mio)	Year	Reference
Sanger	300	2001	Venter et al. (2001)
Sanger	100	2001	Lander et al. (2001)
Sanger	10	2007	Levy et al. (2007)
Roche (454)	2	2008	Wheeler et al. (2008a)
Illumina	1	2008	Ley et al. (2008)
Illumina	0.5	2008	Wang et al. (2008)
Illumina	0.25	2008	Bentley et al. (2008)
Helicos	0.048	2009	Pushkarev et al. (2009)

Table 2.2: Costs for sequencing human genomes. Human genome: ≈3 Gbp. The Sanger sequencing projects assembled the human genome *de novo*, whereas the NGS technologies used resequencing methods, The company Applied Biosysystems estimates costs of <$0.01 in the near future for their SOLiD platform. Table adopted from Pushkarev et al. (2009).

	Read Length (bp)				
Technology	formerly	now	Reads/Run	bp/Run	$/Kbp
Sanger	100	800-1000	100	50-100 Kbp	1
Roche 454	100	400-500	1.25 Mio	1 Gbp	0.05
Illumina GA	35	50-125	140 Mio	28 Gbp	0.002
ABI SOLiD	25	50	750 Mio	40 Gbp	0.002
Helicos	24	32	1075 Mio	37 Gbp	<0.0005

Table 2.3: Comparison of sequencing technologies. Information derived from (Hutchison, 2007; Check Hayden, 2009; Perkel, 2009; Pushkarev et al., 2009) and the company web sites as of August 2009.

NGS technologies that followed 454 sequencing to the market (Rothberg and Leamon, 2008). The first 454 NGS machine was the GS20 (Figure 2.4). Remarkably, it was able to produce a throughput of 50 single ABI 3730XL capillary sequencing machines at a fraction of the costs (Schuster, 2008). Moreover, it was the first non-Sanger technology to sequence an individual human (Wheeler et al., 2008a).

The main principles of the 454 technology are an *in vitro* sample preparation and a miniaturization of sequencing chemistries that enables to generate sequencing data in a massively parallel manner. Long DNA samples are randomly sheared into small fragments (e.g., by nebulization) of sizes between 500 and 1000 bp. For the library preparation, an adaptor is added to each end of these fragments, respectively (Figure 2.4 a). The single stranded DNA fragments (sstDNA) are then used for the subsequent emulsion PCR (emPCR) (Dressman et al., 2003). Therefore, the fragments are mixed with beads carrying oligonucleotides complementary to the adaptors.

Figure 2.3: Roche 454 Genome Sequencer 20. Picture taken from `http://www.roche-applied-science.com`.

The bead-bound library is then set up inside aqueous microdroplets within an oil emulsion together with amplification reagents (Figure 2.4 b). During the subsequent amplification step $\approx$ 10 million copies of each sstDNA template are produced and bound to single beads. This compartmentalized enzymatic amplification technique is essential because it allows for a clonal production of templates without the need for host cells, like *E. coli*. Hence, the potential loss of sequence coverage due to the cloning bias is avoided. At the same time, the DNA amplification ensures later to obtain high levels of signal differing from the background noise. For the sequencing phase, the beads carrying sstDNA are deposited into 3.5 million wells of a fibre-optic PicoTiterPlate^{TM}(Figure 2.4 c and e). The high amount of wells (formerly 1.6 million in 2005) is a key factor to accomplish the high density and parallelism of the sequencing reactions. The wells are small enough to allow only one bead placement per well (55 μm in depth and a diameter of 44 μm) and such, they act as individual reaction vessels where each reaction is isolated from other wells. In addition to the sstDNA beads, smaller beads carrying immobilized enzymes (ATP sulfurylase and luciferase) required for the pyrophosphate reaction are loaded into the wells (Figure 2.4 d).

The sequencing itself is carried out by flowing sequencing reagents (nucleotides and buffers) over the plate. Individual nucleotides are sequently flowed (e.g., A-C-G-T-A-C-G-T...). If the current nucleotide is complementary to the template strand, the DNA polymerase incorporates one or more consecutive nucleotides. The number of incorporated nucleotides is proportional to the intensity of the emitted light signal generated by the sequencing reaction (Figure 2.5 and Figure 2.4 f). The light signal for each flow is cap-

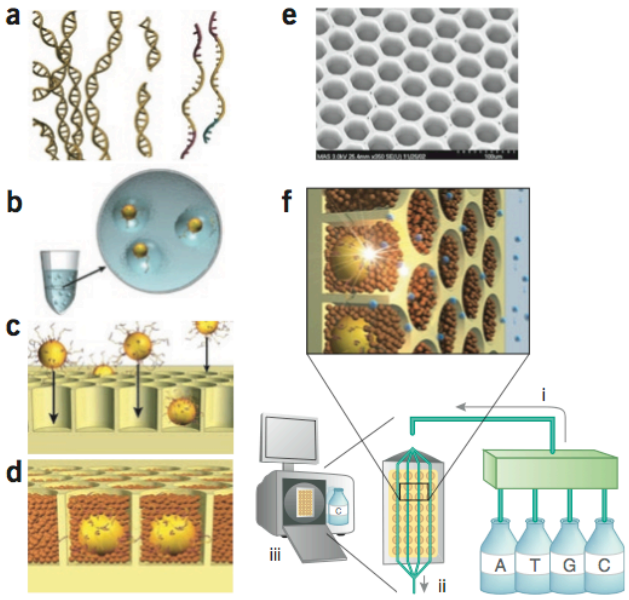


Figure 2.4: 454 sequencing technology. a) Genomic DNA is randomly sheared. Fragments (sstDNA) are ligated to adaptors. b) The single stranded DNA is attached to beads. An emPCR reaction amplifies the fragments within water-in-oil reactors. c) Beads are deposited into wells and d) layered with smaller beads carrying enzymes. e) Scanning electron micrograph showing part of a PicoTiterPlate f) By incorporating the sequentially flowed nucleotides, a light signal is emitted. It is captured by a CCD camera. Picture taken from (Rothberg and Leamon, 2008).

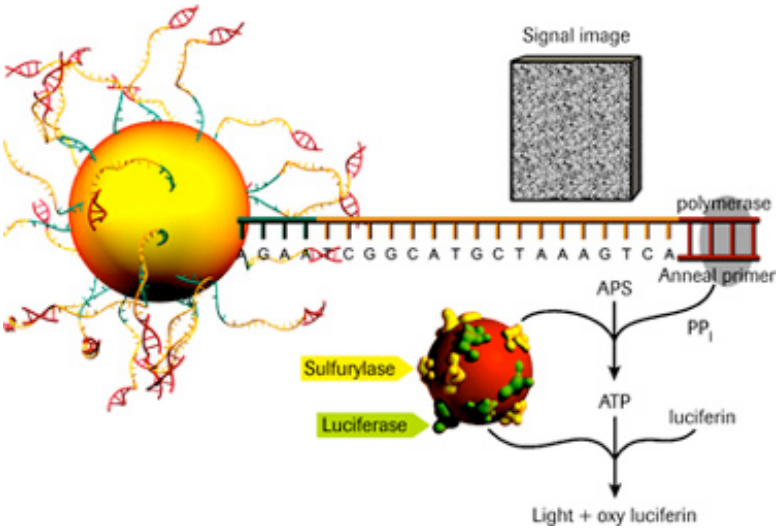


Figure 2.5: 454 pyrosequencing reaction. Incorporation of a complementary base generates pyrophosphate (PPi) which is converted to ATP by the sulfurylase. Luciferase uses the ATP to convert luciferin to oxyluciferin, thereby producing light. Picture taken from `http://www.454.com`.

tured by a CCD camera. The resulting flow grams for each sstDNA bead in a single well are then translated into sequence space, i.e. the base sequence is derived from the intensity of the light signals for each nucleotide flow.

The overall read accuracy could be constantly improved over the years. In Margulies et al. (2005) an accuracy of 96% has been reported for the GS20 platform whereas Droege and Hill (2008) mentioned a single-read accuracy of >99.5% over the first 200 bases. In the meantime, the web site of Roche 454 says that this accuracy is now (August 2009) applicable for the first 400 bases for the GS FLX Titanium Series (`http://www.454.com/products-solutions/system-features.asp`). These improvements are mainly due to advancements in fluidics, surface chemistry and enzymology (Rothberg and Leamon, 2008). Additionally, with each machine and/or chemistry upgrade, the read length could be significantly improved (Table 2.3). Interestingly, the average read length depends on the AT and GC content of the source genome. AT- and GC-rich genomes yield longer sequences than genomes with neutral AT/GC content (Droege and Hill, 2008). Anyhow, the 454 technology is still the only platform able to generate long reads up to 500 bp and is therefore, the first choice for *de novo* genome assembly and metagenomics.

Sequencing errors occasionally result from the misinterpretation of homopolymer runs, i.e. stretches of the same base (e.g., TTTTT or AAA). This leads to single-base insertion or deletions ("overcalling", "undercalling"), rather than to substitutions which occur rarely (rate down to 10^{-6}) (Droege and Hill, 2008). This limitation is due to the lack of a reversible terminator nucleotide preventing the incorporation of more than one nucleotide within a single cycle as stated in MacLean et al. (2009). Moreover, the phenomena of leftover nucleotides lead to asynchronous sequencing of single templates on a bead interfering the polymerase activity. These leftover nucleotides remain to be in the well, though they are unincorporated leading to 'carry forward' or 'incomplete extensions' that may cause sequencing errors (Margulies et al., 2005).

In 2007, the preparation of a paired-end library for 454 sequencing was introduced (Korbel et al., 2007). With the mate-pair information, short-reads limitations could be partially omitted, e.g., it enables to assign unique positions to previously non-unique reads. Also, mate-pairs help to span repetitive regions in *de novo* assembly projects or they may assist in the scaffolding of contigs (for a detailed description refer to Section 2.3.1, p. 29).

To sum it up, the 454 system was the first NGS technology on the market setting the standards for upcoming technologies. Its key advantage is the generation of long (optionally paired-end) read sequences enabling the fast sequencing of whole genomes without the need for a reference genome. Other applications are metagenomics, genome resequencing or whole transcript sequencing studies. (A list of corresponding publications can be found in

Figure 2.6: Illumina Genome Analyzer. Picture taken from `http://www.agrf.org.au`.

Droege and Hill (2008)). However, the relatively low sequencing throughput per run (compared to other NGS technologies) results in the highest cost per base of any NGS systems (Table 2.3).

Genome Analyzer (Illumina)

In 2006, one year after the introduction of the 454 technology, the Solexa 1G sequencer appeared on the market (Bentley, 2006). The technology is now marketed by the company Illumina (Hayward, CA, USA). For the first time, a sequencing technology was capable of generating ≈1 billion bases (1 Gbp) per sequencer run which significantly decreased the cost per base. Obviously, this unprecedented raise of data output heated the race of the sequencing challenge. The current sequencer model is the Genome Analyzer$_{IIx}$ (`http://www.illumina.com`) which is able to generate even more sequence data per run (see Table 2.3 and Figure 2.6). Based on previous works (Turcatti et al., 2008), the technology features certain characteristics which differ from the 454 approach: Instead of applying the bead approach of the emulsion PCR to amplify the DNA templates, the Illumina technology uses the so-called bridge PCR (Adessi et al., 2000; Fedurco et al., 2006). Furthermore, read lengths are considerably shorter than 454 reads (between 36 and 125 bp). By incorporating chain-terminating nucleotides, complications regarding the homopolymer detection are avoided.

The first preparation step comprises the random shearing of the source DNA into smaller fragments. Then, adaptors are ligated *in vitro* onto each denaturalized template DNA of the shotgun library. One end of the ligated products is covalently tethered to the planar surface of a solid glass slide (flow cell). Primers on the slide complementary to the other end of the tem-

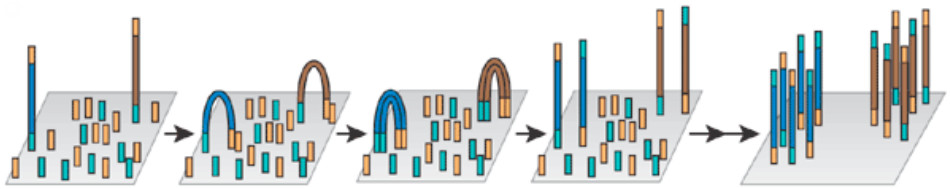


Figure 2.7: DNA Amplification via bridge PCR. The DNA shotgun library is covalently attached to a planar surface where it forms bridge-like structures. After the PCR step, several million clonal clusters of DNA templates exist on the chip. Picture taken from (Shendure and Ji, 2008).

plate force the template DNA to bend down to form bridge-like structures. Finally, the *in situ* amplification (bridge PCR) generates clonal clusters each consisting of about ≈ 1000 amplicons. These amplicons remain immobilized and clustered to a single physical location on the slide (see Figure 2.7). This amplification step is essential to provide a strong fluorescent signal resulting in several millions of isolated spots of DNA all over the microfluidics chip (100 million samples per cm^2).

For the subsequent sequencing, the template DNA is linearized and all necessary primers, nucleotides and a polymerase are directly added to the flow cell. The four nucleotides are fluorescently labeled depending on the type of base. Additionally, they are "reversible chain terminators" which ensures that the template strand is extended by only one base per cycle. After each incorporation of a fluorescent base, the flow cell is interrogated with a laser at several locations. This results in several image acquisitions at the end of a single synthesis cycle. After the chemical cleavage of the terminator and the fluorescent label, the process is repeated for the next nucleotide. Thus, the amount of consecutive cycles determines the read length.

On the one hand, the usage of chain terminators avoids the misinterpretation of homopolymer lengths, as known from the 454 technology. On the other hand, complications arise due to the incomplete removal of the fluorescent label or the terminator. This results in substitution errors, rather than in insertions or deletions. The error rates of the Illumina technology are comparable to the 454 system. Novel system enhancements provide protocols for the generation of mate-pairs (2 x 75 bp) with insert sizes between 0.2 and 5 Kbp (`http://www.illumina.com`).

The decreased sequencing costs and high volumes of data output proved to be ideal for resequencing projects, targeted sequencing, SNP detection and gene expression studies. Mentioning the new mate-pair capabilities, the company even suggests to use the system for analyses such as sequence assembly, *de novo* sequencing, and large-scale structural variation detection,

Figure 2.8: SOLiD platform (Applied Biosystems). Picture taken from (Blow, 2007).

which, in fact, are application fields for rather long reads technologies, like 454.

One disadvantage that, in a sense, applies to most NGS technologies is the data-intensive output. Different to 454, the Illumina sequencer generates more than one image per cycle because more than one location on the chip has to be scanned per base incorporation. For one sequencing run generating 36 bp reads and one flow cell, ≈800 GB of (temporary) image data is produced. This demands for efficient transfer and storage solutions on the part of the selling company but also on the part of the consumer who faces the problem of effective and cost-saving long-term storage of the data.

SOLiD (Applied Biosystems)

The SOLiD (Supported Oligonucleotide Ligation and Detection) platform is based on methods described in Shendure et al. (2005). The company Agencourt Personal Genomics which was later taken over by Applied Biosystems (Foster City, CA, USA) developed a sequencing technology that uses a "hybridization-ligation-chemistry". The first machines commercially available were introduced in 2007 (Figure 2.8). Although SOLiD relies on comparable techniques for the library and template preparation (emulsion PCR) like 454, the sequencing process requires the enzyme DNA ligase instead of polymerase. The volume of the sequencing data output and the length of the reads are similar compared to the Illumina system.

At the beginning, the template DNA is amplified by using emulsion PCR.

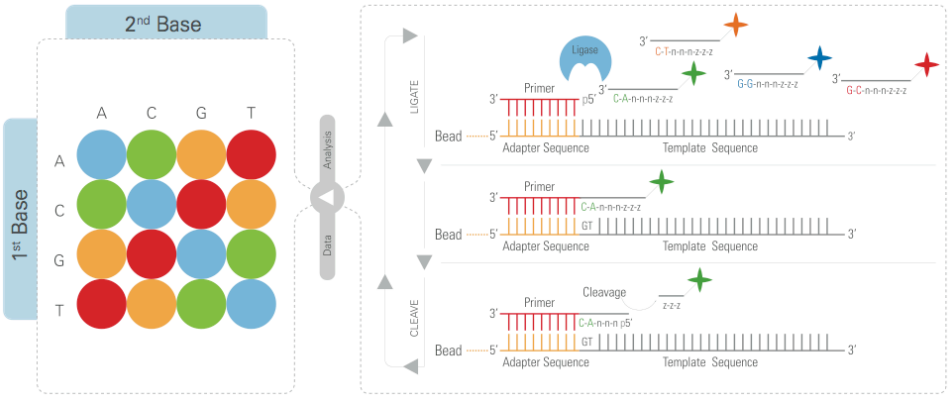


Figure 2.9: SOLiD sequencing process. Oligonucleotides which are fluorescently labeled are used for the synthesis of the complementary strand. The two-base-encoding and the repeated cycles covering each base twice improves the quality of the base-calling. Picture taken from the official SOLiD brochure found at `http://www.appliedbiosystems.com`.

The beads of the SOLiD system are smaller than those of 454, so a higher density on the array is accomplished. Covalently anchored on the glass slide, primers bind to the adaptor sequences of the templates. Then, a set of four fluorescently labeled di-probes are added to the mix. These di-probes are short oligonucleotides of length 8 bp (octamers) with random sequence, except known dinucleotides at the 3' end (Figure 2.9). The fluorescent dye at the 5' end of the octamer corresponds to the type of dinucleotide. In case an octamer is complementary to the template, it will be ligated, and the two specific nucleotides can be called. Therefore, an image is acquired after the ligation for the same position for all beads on the array. Then, the fluorescent dye is removed so that other octamers can be ligated. After several ligations (e.g., 7 ligations for a 35 bp read), the synthesized DNA is removed and the primer is denaturalized. Then, a new round of ligation cycles starts by adding a new universal primer positioned one or more bases back on the template DNA (compared to first cycle). This process is repeated from different starting points, therefore, covering each base position twice. This technique is called *two-base-encoding* and acts like an error-correction scheme to ensure accurate base-calls (Shendure and Ji, 2008). Currently, an accuracy of 99.94% is reported (99.999% for 15x coverage).

On the one hand, the ligase-based approach helps to avoid polymerase-induced errors. On the other hand, the length of the reads remains rather short (up to 50 bp). However, a protocol for the generation of mate-pairs is also available (2x50 bp).

Because of the short read length, the range of applications for SOLiD

is similar to Illumina's technology, such as (targeted) resequencing, small RNA and transcriptome analysis and SNP detection. The *de novo* genome sequencing is only reasonable in conjunction with a long-read technology.

2.2.3 3rd Generation Sequencing Technologies

The 3^{rd} generation or next-next-generation sequencing (NNGS) technologies share a characteristic feature which distinguish them from the 2^{nd} generation (NGS) technologies: the "direct" and fast sequencing of single DNA molecules. This makes the time consuming sample preparation and, in particular, the amplification step dispensable. Consequently, the amplification bias and therefore, synchronization problems during the sequencing do not play a role anymore. As a consequence, high-quality sequence data can be generated in relatively short time at lower costs.

At present, three different technologies are of importance, whereas two of them are still experimental, i.e. the commercial launch is scheduled within the next years. For a comprehensive list of companies offering sequencing products, please refer to an article of Blow (2008a).

Heliscope (Helicos Biosystems)

Heliscope is the first NNGS platform brought to the market in 2008. It is marketed by the company Helicos Biosystems (Cambridge, MA, USA). Its "true single-molecule sequencing" technology (tSMS) was first described in (Braslavsky et al., 2003). Although the commercial launch has started already, the company has been "dogged by sequencing errors" (Check Hayden, 2009). The first sold machines have been returned back by unhappy customers.

However, the idea of the tSMS technology is straight-forward: The randomly fragmented DNA (which does not have to be clonally amplified) is immobilized onto a glass slide by attaching poly(A)-tails at the templates' ends. The poly(A)-tails are captured by poly(T) adaptors on the slide. One fluorescently-labeled base at a time is incorporated to create complementary strands. Reversible terminators avoid the incorporation of more than one nucleotide. After the image acquisition using a high-resolution optical microscope, the fluophore is removed, and the process is repeated.

The typical read length is rather short (about 24-32 bp). Main error types are deletions (2%) and insertions (1.2%). Substitutions occur only rarely (0.38%) (Pushkarev et al., 2009). The key feature of this technology is the cost-efficiency as recently stated in Pushkarev et al. (2009). In this article, the first single-molecule sequencing of a human genome using the Helicscope technology is described. The authors claim that their sequencing has been accomplised at lower cost ($48,000) than previous human sequencing projects. However, the acquisition costs of the sequencer ($750,000-

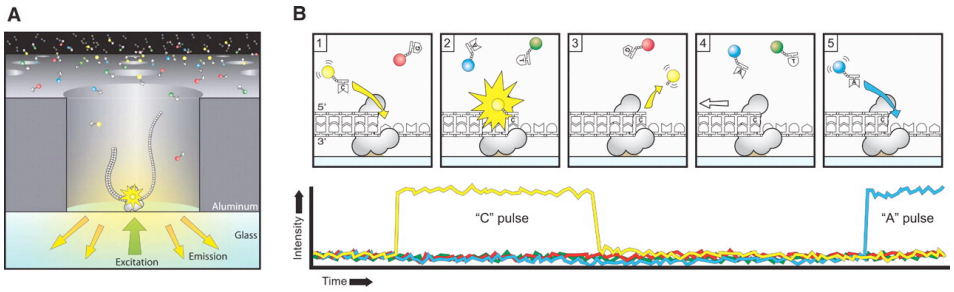


Figure 2.10: Single Molecule Real Time Technology (Pacific Biosciences). The sequencing reaction takes place in a zepto liter well enabling the real time monitoring of base incorporations. Picture taken from (Eid et al., 2009).

1,000,000) and its reagents are still relatively high (Check Hayden, 2009).

Pacific Biosciences

The company Pacific Bioscience (Menlo Park, CA, USA) promotes its "Single Molecule Real Time" (SMRT) technology (Eid et al., 2009) (`http://www.pacificbiosciences.com`). The sequencing occurs in zepto liter (10^{-21}) wells which contain the immobilized single DNA molecules (Figure 2.10). After adding a high concentration of fluorescently labeled nucleotides and the polymerase, the sequencing begins. As the complementary strand grows, the fluophores are cleaved and the light signal is captured by a highly focused detection system. Due to the small well and the high nucleotide concentration, the sequencing rate is expected to be fast (10 bases/sec, (Rothberg and Leamon, 2008)). In Check Hayden (2009) it is stated that the SMRT technology will be able to sequence a human genome in less than three minutes in 2013. Further, the company claims to "generate reads that are thousands of nucleotides long, at the expense of overall output" (MacLean et al., 2009). Currently, average read lengths of 568 bp (max: 2,805 bp) are reported (Check Hayden, 2009) which would really take (meta)genomic research to the next level. The first commercial release is scheduled for late 2010.

Oxford Nanopore Technologies

Nanopore sequencing is a fairly different approach to read the DNA molecule (Clarke et al., 2009). Instead of capturing light signals with CCD cameras to detect base incorporations, the company Oxford Nanopore Technologies (`http://www.nanoporetech.com`) modified an α-hemolysin nanopore (inner distance: 1 nm) for the detection of individual nucleotide monophosphates (BASE technology). DNA is digested by an exonuclease attached to the pore. Each nucleotide "falling" through the pore blocks the electrical current that runs through this pore. Because each nucleotide has a characteristic

current amplitude, one is able to read the sequence base-by-base. Similar to the SMRT technology of Pacific Biosciences, the nanopore technology has the potential to generate long reads (up to several thousand base pairs). As stated in Rusk (2009), read accuracy is quite high (99.8%), and the error correction is expected to be straight-forward. Furthermore, the massive parallelization of this technology (hundreds of thousands of pores on an array) will be able to produce large-volume sequencing data in short time. As of August 2009, no official release date has been announced for this technology.

2.3 Genome Assembly

As already described in Section 2.2, the developments of DNA sequencing technologies have had an enormous impact on genomic research. At first glance, this is quite surprising regarding the technical shortcomings of all technologies: instead of the complete genome sequence, the typical output of sequencer machines is a set of DNA fragments, orders of magnitude shorter than the original genome sequence (read lengths between 25 and 1000 bp, see Table 2.3). So, to be able to interpret the data, specific computer programs, called genome assemblers, are employed which piece the DNA fragments (sequencing reads) back together to obtain the original sequence. A completely assembled sequence is essential for an in-depth investigation of an organism's genome and its genetic features. Several different assembly approaches have been reported in the last years depending on the chosen sequencing technology, and therefore, depending on read length, typical error characteristics and the amount of sequence output. (For a comprehensive overview of current genome assemblers, please refer to Scheibye-Alsing et al. (2009)). Currently, the "ever-changing technology landscape" (Pop, 2009) leads to the need for improved algorithmic solutions to keep pace with the tremendous advancements in sequencing technology.

The "shotgun sequencing" technique was introduced by Sanger in 1982 (Sanger et al., 1982). Its main idea is the random fragmentation of the original genome into smaller DNA fragments which are multiply cloned and then partially sequenced to generate the read sequences. A subsequent (computer-based) assembling process combines all reads to reconstruct the original genome sequence. In 1995, Fleischmann *et al.* applied this strategy the first time in a large-scale experiment when sequencing the genome of the bacterium *Haemophilus influenzae* (Fleischmann et al., 1995). Over the last couple of years, a lot of genomes could be assembled by using the shotgun method including the 120 MB genome of the fruit fly (Myers et al., 2000) and parts of the human genome (Venter et al., 2001).

The current status of sequencing and genome assembly projects (Table 2.4) indicates that Prokaryotes have been sequenced and assembled by far

most frequently, followed by Eukaryotes, Fungi and Plants. This dispar-

Organism	Complete	Draft Assembly	In progress	Total
Prokaryotes	939	1028	877	2844
Animals	4	75	60	139
Plants	2	11	47	60
Fungi	10	71	42	123
Protists	6	24	24	57
Total	**961**	**1209**	**1050**	**3220**

Table 2.4: Genome seqencing project statistics. Information derived from `http://www.ncbi.nlm.nih.gov/genomes/static/gpstat.html` as of August 2009.

ity is mostly due to the simple genome structures of Prokaryotes (genomes sizes <15 Mbp, high gene coding density, mainly haploid chromosomes and plasmids) and their relevant role in many human diseases and for molecular research. Another crucial observation can be made: The amount of projects classified as "Draft Assembly" always exceeds the number of "Completed" projects. "Completed" here means that the final and complete genome sequence has been successfully assembled and that no more sequencing has to be done. In contrast "Draft Assemblies" are actually not yet finalized, i.e. the reconstruction of the original genome sequence could not (yet) be finished completely: the genome sequence is still fragmented. This may be due to two reasons: On the one hand, some projects did not aim at a complete genome assembly due to financial limitations or just because the whole reconstructed sequence was of no interest for the specific research. On the other hand, and this is the more likely explanation, most of these projects had profound difficulties to come up with the finalized genome sequence.

What are the obstacles and difficulties that complicate a successful assembly? First of all, genome assembly is like a huge "jigsaw puzzle": millions of DNA reads have to be assembled into a complete picture of a genome (Pop et al., 2002). But, as already described in Section 2.2, every sequencing technology produces data with a underlying error rate which is specific for each technology. Consequently, the input data (read sequences) for the assembler software contains errors. Secondly, the characteristics of the DNA itself pose some serious challenges to the assembly process: for example, repetitive regions, called *repeats*, complicate the positioning of reads. (For example, more than 50% of the human genome are repeat sequences not coding for any proteins.) Further, due to some reasons, specific stretches of DNA can be hardly sequenced, thus, some regions of the original genome are underrepresented leading to a fragmented (draft) assembly.

In the next subsections, the typical genome assembly pipeline is described starting with the generation of the reads and ending with the finishing of the genome sequence. New concepts for genome assembly are intro-

duced which account for the specific characteristics of the high-throughput data derived from new sequencing technologies (resequencing and hybrid assembly).

2.3.1 Whole Genome Shotgun Assembly

In this subsection, the *de novo* sequencing of a genome using the Sanger technology (Section 2.2.1) is described. Note, that the methodology is not directly applicable to the new sequencing technologies.

To sequence a genome *de novo*, the DNA is first isolated from the organism, amplified and then physically sheared (e.g. by sonification or nebulization) into random-sized fragments, called *inserts* (Figure 2.11 a-c). To clonally amplify the fragments, they are inserted into stable *vectors* like plasmids or BACs (bacterial artificial chromosomes) whose sequence is known and which accept the insertion of foreign DNA. The vectors are then cloned in host cells (*in vivo*) such as *E. coli* (Figure 2.11 d). There exist several sorts of vectors each capable of propagating clones of different sizes (e.g. 2, 5, 10 and 50 Kbp). The mixture of different cloning systems later facilitates the resolution of certain structural variants of the DNA like repeats. According to Scheibye-Alsing et al. (2009), the million copies of the clones should have the following properties:

- The clones should be highly redundant, i.e. the fragments optimally cover the entire genome multiple times. For example, a clone coverage of 10 means that, on average, 10 clones span each base position.
- The clone coverage should be random and even, i.e. no specific region of the genome is covered by significantly more or less clones (no bias).
- The clones should be stable within the host cell. No recombination or reorganization during the propagation process should occur.

Unfortunately, in practice, these requirements are rarely satisfied causing a non random clone distribution and therefore incomplete sequencing. For example, if an insert sequence is toxic for the host cell, this specific insert will be missed in the subsequent sequencing and gaps will complicate the assembly. In contrast, the new sequencing technologies replace the vector cloning with *in vitro* approaches (Section 2.2.2), avoiding any cloning bias. Single molecule sequencing technologies (as described in Section 2.2.3) even completely omit the cloning step.

Following, the sequencing of the clones is performed obtaining a set of so-called *reads*. Depending on the chosen sequencing technology, the reads differ in length (compare Table 2.3) and error rate distribution. In many projects, the clones are sequenced from both ends ("double-barreled shotgun sequencing") obtaining mated reads (also called mate-pairs or paired-end reads) (Figure 2.11 e) The information of mate-pairs are extremely valuable

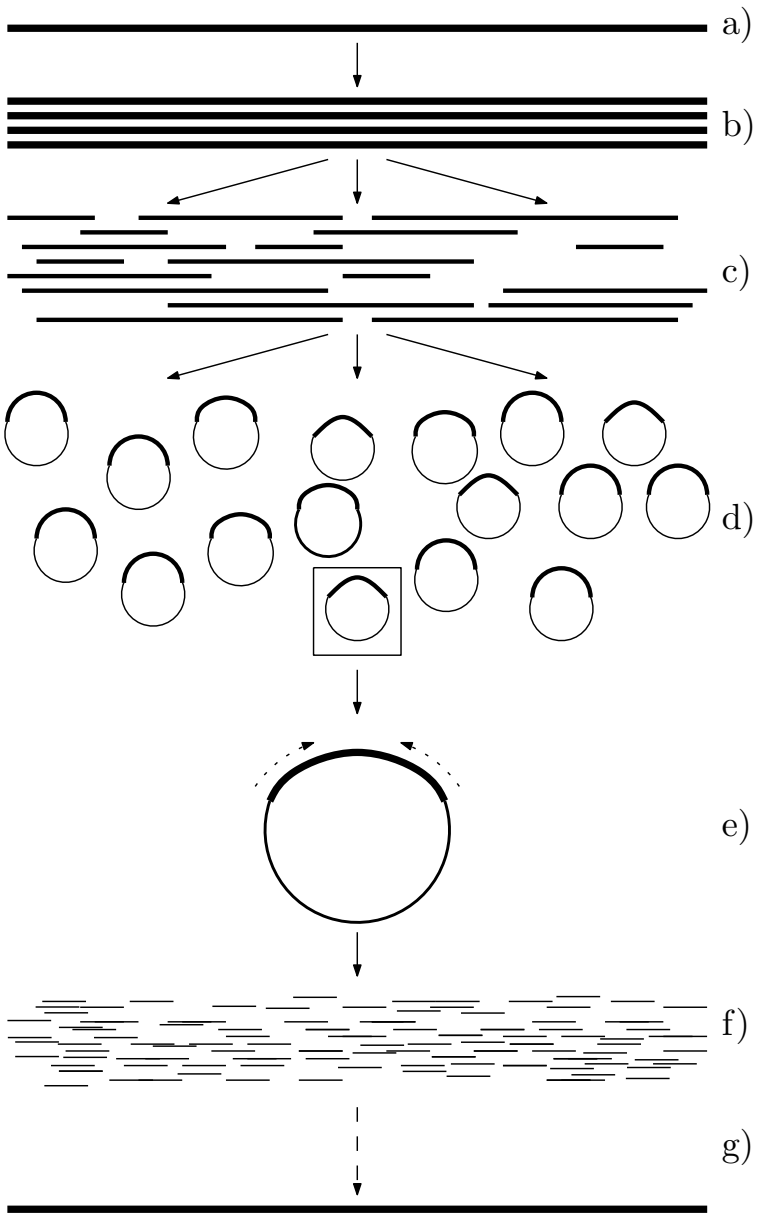


Figure 2.11: Whole Genome Shotgun Assembly. Several steps are needed to obtain a large collection of reads. a) The original DNA sequence is isolated and b) amplified. c)+d) The sequences are randomly fragmented into inserts which are then incorporated into different vectors depending on their lengths. e) After the cloning in host cells, the fragments are sequenced. f) The set of obtained (paired-end) reads optimally covers the entire genome multiple times. g) Several assembly steps are conducted to reconstruct the whole genome sequence.

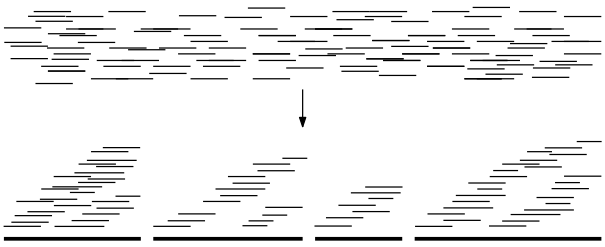


Figure 2.12: Overlap and consensus phase. Reads are overlapped and assembled into contigs.

for the subsequent assembly steps. The known distance (the size of the clone) and orientation of the pairs of reads later helps to order the assembly and to reveal structural variants like repeats.

At the end of the sequencing phase, a large set of reads has been generated that ideally are distributed over the entire genome and that statistically cover each genome position several times (Figure 2.11 f). A high over-sampling of reads is important to ensure that each base position in the genome is sampled by at least one read (Pop, 2009). After cleaning the reads from unspecific DNA like vector or *E.coli* sequence fragments, the main assembly pipeline begins with the search for read overlaps.

Overlap-Layout-Consensus

The first step of the genome assembly pipeline is the combination of all short read sequences into so-called *contigs* (contiguous sequences). Hence, to compute the read overlaps, the assembler employs the sequence similarity to compute all pairwise alignments between the reads. Many assemblers use heuristic variants of the Smith-Waterman algorithm (Smith and Waterman, 1981). This phase is often called the "computational bottleneck" which implicates that it is computationally intensive and thus, very time consuming. In general, each pair of reads is checked for overlaps which means that the suffix of a read matches the prefix of another read. The quality of an overlap is determined by its length and the number of shared base pairs (level of identity). The required overlap quality depends on the average read length produced by the sequencer.

The contig sequence is derived from the consensus sequence of all reads covering specific base positions (Figure 2.12). Base quality values provided by the sequencer ensure to determine the most reliable nucleotide at each contig position. The more high-quality reads cover a position, the higher the confidence is that the consensus nucleotide is the correct one. The amount of reads generated by new sequencing technologies is often an order of magnitude higher than the amount of reads generated with Sanger technology. Thus, parallelized or grid-based software solutions help to decrease the com-

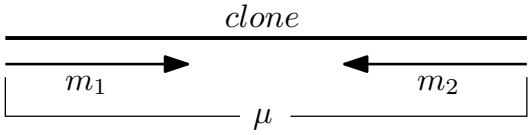


Figure 2.13: Mate-pair. Two mated reads m_1 and m_2 are sequenced from a single clone. Their distance is given by the mean clone length μ and a standard deviation σ.

puting time. Also, so-called *k-mer* algorithms based on indexing strategies are employed (Pop, 2009). As alternative to the greedy overlap method, the Eulerian fragment assembly was introduced in Pevzner et al. (2001). By avoiding the costly computation of pairwise alignments between reads, it is better suited for next-generation technologies, such as Illumina or SOLiD, that generate Gbp of short reads per run (see Section 2.2.2). According to MacLean et al. (2009), the Eulerian assembly approach (also called de Bruijn graph assembly) is able to "put together bigger contigs and can handle sequencing errors and complex genomes better than their counterparts".

In practice, the overlap phase suffers from the existence of repeats in the DNA. If reads are sampled from different regions harboring the same repeat sequence, they will be erroneously assembled together because they share the same repeat sequence. Hence, they will form *repeat clusters* that are composed of more overlapping reads than would be expected by chance. Such misassemblies lead to a fragmentation of the final assembly. One possibility to resolve repeat-induced errors is to identify and mask repetitive regions *a priori* using, for example, the software RepeatMasker (Smit et al., 1996-2004) or a close reference genome whose repeat regions are already known. Other approaches try to reveal repeats by analyzing the overlapped reads within repeat clusters or by using the mate-pair information: If one read of a mate-pair is located within a repeat and the other read is located outside of this repeat, it becomes feasible to distinguish between different repeat regions. Reads within identified repeat regions may then be initially excluded from the assembly.

Scaffolding

The result of the overlap-layout-consensus phase is a set of larger fragments, called contigs which represent local islands of the original genome. The goal of the scaffolding phase is to detect the ordering and orientation of the contigs, thereby, forming larger structures called scaffolds or super-contigs. By using the mate-pair information (orientation and distance of paired reads sequenced on a single clone, Figure 2.13), it is possible to infer a partial contig layout. Assuming that one end of a mate-pair is located in one contig c_1 and the other end in another contig c_2, the relative orientation of c_1 and c_2 can be easily determined. Obviously, the more mate-pairs

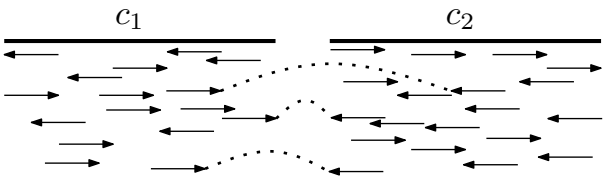


Figure 2.14: Mate-pairs spanning a gap. The relative orientation and ordering of contigs c_1 and c_2 can be inferred by the presence of mate-pairs that span the gap between them. If the mate-pair distances confirm each other, the length of the gap separating the two contigs can be predicted as well.

span a gap of two contigs, the more likely these contigs will be neighbored (Figure 2.14). It is important that the relative orientation of the mates and their distance corresponds to the known clone length. If the distances of all present mate-pairs confirm each other, the gap distance between two contigs can be predicted as well. For this approach, the use of clone libraries with different lengths is advantageous. Of course, sequencing errors or a lack of mate-pairs may complicate the scaffolding.

Three other (complementary) approaches are used in practice: first, an optical mapping method uses known restriction site cuts along a genome to infer a contig layout (Samad et al., 1995). Alternatively, contigs can be reordered by laborious, manual approach: Therefore, primers are synthesized which are complementary to short regions at the ends of single contigs. Then, a PCR is conducted hoping to reach another contig end to close the gap. Unfortunately, this strategy is very time-intensive. Another possibility to order contigs is the homology method: by using the sequence similarity between two genomes of closely related organisms, the unfinished assembly can be arranged according to a (already complete) reference genome. Of course, one has to be aware of any genome rearrangements that affect the homology data. An algorithmic approach and a software solution for this problem is presented in Chapter 3. Generally, it is recommended to use complementary information for the scaffolding process.

In practice, the final contig layout is represented by multiple scaffolds separated by gaps of unknown size.

Gap-Closure and Finishing

To obtain a one-piece genome sequence (in case of an unichromosomal organism), all remaining gaps have to be filled with sequence data. Similar to the scaffolding step, PCR runs are conducted starting at the ends of neighbored contigs to produce new sequence data that potentially cover the missing sequence information between them. If this lab approach and the subsequent assembly runs are successful, a single contig may be obtained (e.g. in case of prokaryotic genome) which eventually represents the final

genome sequence: the genome has been finally "closed". To achieve this goal, the procedure of combined PCR and reassembly commonly has to be repeated several times.

The term "finishing" describes the efforts to reexamine the assembly regarding misassemblies and low coverage regions. Again, the mate-pair information is highly valuable for the validation of the assembly quality: if, for instance, the distance between paired-reads significantly deviates from the stored clone size, this may indicate the presence of errors in the assembly. A common software tool for an in-depth assembly analysis is the tool Consed (Gordon et al., 1998). It provides a graphical user interface for the manual inspection of mate-pairs or ambiguous regions in the assembly. To complement the mate-pair information, other validation methods like whole genome comparisons to closely-related reference genomes may be helpful.

However, this phase of the assembly is not always crowned with success: due to unresolved misassemblies or other technical limitations, many genome assemblies can not be finalized. Such "draft assemblies" remain fragmented (compare Table 2.4). Many draft assemblies are/were based on Sanger sequencing and therefore, they might suffer from the mentioned cloning bias. Still, people are optimistic to close such genomes in the near future by employing the NGS technologies that avoid a lot of shortcomings of the Sanger sequencing (see Section 2.2.2).

2.3.2 Resequencing

Read sequences produced by NGS technologies, such as Illumina GA or ABI SOLiD are rather short (25 - 125 bp). This size limitation complicates the sequencing and *de novo* assembly of unknown genomes. Alternatively, NGS projects make use of (parts of) reference genome sequences of closely-related organisms or strains. The idea is to find the corresponding placement of each read within the reference sequence. If a unique stretch of sequence is detected which is very similar or even equal to the read sequence, the read is "mapped" onto or aligned to the reference genome. The allowed number of unmatched base positions per read is usually very small (e.g. 2 bp) and depends on the average read length generated by the sequencer.

Large-scale resequencing projects benefit from the massive data output of NGS technologies because a high sequence coverage ensures a high mapping accuracy. This mapping accuracy is needed for effective genotyping analyses, such as the search for single nucleotide polymorphisms (SNPs) or other structural variants. In particular for disease and other mutation analyses, resequencing brings great potential for genomic medicine. An example of a human resequencing project is described in Wheeler et al. (2008a) (the "Watson" genome). In this study, about 106 million reads have been mapped to the human reference genome, revealing 3.32 million SNPs.

The mapping of reads onto a reference genome poses a lot of computa-

tional challenges: A large volume of read data has to be efficiently processed and aligned to a large reference sequence. Typical alignment programs like BLAST (Altschul et al., 1990) ("seed-and-extend approach") are way too slow and, therefore, they are inappropriate for this task. Several software solutions have been recently developed that use novel, computational concepts (for example Maq (Li et al., 2008) and Bowtie (Langmead et al., 2009) (for a comprehensive listing refer to Trapnell and Salzberg (2009)). These tools allow the fast mapping of millions of short reads to a reference sequence on a single desktop computer.

Obviously, repetitive regions and sequencing errors pose serious challenges to the alignment process. Here again, the consideration of sequence quality values, as well as the mate-pair information (Figure 2.13) helps to resolve ambiguities regarding the read placement.

2.3.3 Hybrid Assembly

Each sequencing technology has individual characteristics concerning the read length, the incorporation of errors and the generation of mate-pairs (among others). For example, the "long" 454 reads suffer mostly from insertion and deletions, whereas the short Illumina reads rather contain substitution errors. Hence, the presence of several technologies on the market provides the opportunity to combine their strengths in a hybrid assembly approach. Such an complementing strategy may help to overcome the shortcomings of each individual technology. As stated in Pop (2009), the direct integration of input data from different technologies is not always possible. For example, the combination of differing read lengths will pose difficulties in the overlap phase. Consequently, both data inputs have to be adapted to fit each other. This has been done for instance in a study described in Goldberg et al. (2006): in this project a Sanger/454 hybrid assembly was carried out. To adapt the different types of sequencing data, the 454 contigs were first fragmented into Sanger-like reads. Subsequently, the Celera assembler used both, the Sanger reads and the chopped 454 contigs, to assemble a bacterial genome. A similar approach has been described in Reinhardt et al. (2009): in this study 454 and Illumina reads were used to assemble the genome of *Pseudomonas syringae* : Illumina reads were first solely assembled into contigs to obtain larger fragment lengths similar to 454 reads. The final assembly produced several large scaffolds "at a fraction of the traditional cost and without the use of a reference sequence" (Reinhardt et al., 2009).

The hybrid assembly is a promising strategy for a *de novo* genome assembly. However, basic guidelines have to followed as discussed, for example, in Jeong and Kim (2008). In this article, the authors recommend different assembler and analysis tools depending on the mixture of the input data (e.g., if more 454 than Sanger data is available or vice versa). Another complication is the mixture of base qualities that may differ be-

tween the platforms. For example, the 454 assembler ("Newbler") reports quality values for contigs which cannot be easily correlated with quality values of Sanger reads as described in a hybrid-assembly pipeline for the Arachne assembler (Batzoglou et al., 2002) at the Broad Institute (`http://www.broadinstitute.org/crd/wiki/index.php/Hybrid_assembly`).

As sequencing technologies and assembler tools are further developed, the support for hybrid assemblies is expected to be improved in the near future.

2.4 Metagenomics

2.4.1 Motivation and Goals

Biologists estimate that microbes make up more than one-third of earth's biomass (Whitman et al., 1998). Although more than ten million different species of bacteria colonize our planet (Eisen, 2007), theirs roles in the biosphere is still far from being completely understood. Further, within the last decades only a minor part (<1%) could be already described and classified (Eisen, 2007). In contrast, the knowledge about higher animals and plants is more advanced: approximately 90-99% of all species are already known and classified (Snyder et al., 2009).

The advent of the automated DNA sequencing technology help to obtain a glimpse of the rich set of the diversity and functional capabilities of microbial species. For example, in traditional sequencing projects, single organisms are isolated and cultivated under laboratory conditions. Although, such genome projects provide an in-depth analysis of the genome of a single organism, they fail to completely clarify the role of that organism within its habitat or ecological niche and the interaction to other species. Eventually, people noticed that the vast diversity of the microbial world can not be comprehensively studied by such segregated approaches. The acceptance of the fact that most naturally occurring microbial species can not be cultured under laboratory conditions led to the emergence of a new exciting research discipline called *metagenomics*.

The term "metagenomics" (also: environmental or community genomics) was first coined in an article in 1998 (Handelsman et al., 1998). It is described as the sequencing and analysis of DNA from environmental samples while bypassing the need for culturing and cloning individual organisms. The direct study of microbial species in their natural environments has been made possible by several novel cultivation-independent, molecular methods and the recent developments in high-throughput DNA sequencing (see Section 2.2.2). The analysis of metagenomic data promises to reveal new life forms and unique biosynthetic capabilities. The gained knowledge is expected to have an impact for human health, agriculture, food production and even alternative energy.

However, to "close the gaps between genotype, phenotype and environment" (Harrington et al., 2007), the study of metagenomes demands for capable and intuitive bioinformatic solutions and tools. Considering the primary output of metagenomic projects, large volumes of sequencing data which are highly fragmented and noisy, have to be efficiently processed and structured. The sheer mass of anonymous DNA fragments of a heterogenous, environmental sample presents a major challenge for computational biologists. In particular, the main tasks that are addressed in typical metagenomic projects are:

- *Taxonomical analysis*
  To obtain an overview about the species diversity within the sample, the individual community inhabitants are detected and characterized.

- *Quantitive analysis*
  The abundances of the contained species are estimated.

- *Functional analysis*
  The functional analysis comprises methods to detect (new classes of) protein coding genes or to reconstruct metabolic processes and networks.

Since the DNA fragments obtained from the sequencer are short, anonymous reads (i.e. the species origin of each read is unknown), their characterization and assignment to known (or unknown species) is one of the "hottest topics" in metagenomic research. Likewise, the prediction and annotation of open reading frames (ORFs) on metagenomic reads generally suffers from the limited sequence lengths. The unprecedented complexity of environmental data means that traditional software tools, like genome assemblers or gene finding programs, are not applicable to the full extent. As a consequence, several approaches to analyze a metagenome have been made available that try to tackle these problems (For a comprehensive overview, please refer to Kunin et al. (2008)). Frequently, projects use a mixture of different tools and web-services that allow to interpret the complex data sets. Often, this leads to an undesirable situation that the results of different projects can be hardly compared.

The next sections describe the mentioned aspects of a metagenome analysis in more detail.

2.4.2 Representative Metagenomic Projects

Since 1998, a couple of metagenomic analyses have been reported providing striking insights into diverse ecological systems, such as seawater, soil, air, biofilms and the human body. In this subsection, a brief (and incomplete) overview of the broad range of environmental studies is presented.

One of the pioneering works was the analysis of ocean surface water in 2004 conducted by a team led by J. Craig Venter (Venter et al., 2004). During this first large-scale project, researchers sampled water probes in the Sargasso Sea near Bermuda. Using whole-genome shotgun sequencing (Sanger technology), they produced a total of approximately one billion base pairs. Moreover, more than one million unknown protein sequences have been predicted. But that boat trip was only a "primer" of the subsequent Global Ocean Sampling (GOS) project as reported in 2007 (Yooseph et al., 2007; Rusch et al., 2007). During 2004 and 2007, Venter's team collected bacteria at 41 sites in the Atlantic and Pacific, particularly near the Galapagos islands. The data output was stunning (7.7 million reads consisting of approximately 6.3 billion bp) and required the application of "new comparative genomic and assembly methods" like "fragment recruitment" and "extreme assembly" (Rusch et al., 2007). This "was easily the largest DNA sequencing of environmental samples ever accomplished" (Bohannon, 2007) and certainly, it was also the most expensive one: the funding amounted to $10 million. The large-scale Sanger sequencing and gene prediction resulted in 6.12 million hypothetical proteins, almost doubling the number of known proteins present in databases.

Other projects focussed on viruses rather than bacteria in marine metagenomes (e.g. Breitbart et al. (2002), Culley et al. (2006) and Williamson et al. (2008)). Besides the study of species communities in seawater, many projects focussed on soil samples. It turned out that the typical community structure (number and abundances of different species) of soil metagenomes is far more complex compared to other habitats (Handelsman et al., 1998; Tringe et al., 2005; Urich et al., 2008)). In addition, the analysis of soil metagenomes is of interest for the pharmaceutical industry, "as soil organisms have been the main sources of new natural products, including antibiotics" (Daniel, 2005).

It is a well known fact that the number of cells in the human body (10^{13}) is far less than the number of microbial cells actually inhabiting our body (10^{14}) (Berg, 1996). Consequently, a global initiative was started to characterize the interaction between microbes and the several parts of the human body (Turnbaugh et al., 2007). The Human Microbiome Project (HMP) employs metagenomic strategies to gain knowledge about the complexity of human microbial communities that help to understand how microbes "contribute to normal physiology and predisposition to disease" (`http://nihroadmap.nih.gov/hmp`). Several results have been already reported (Turnbaugh et al., 2006; Gill et al., 2006).

Further exemplary projects focussed on an airborne metagenome (Tringe et al., 2008), microbes in honey bee colony collapse disorder (Cox-Foster et al., 2007), the hindgut microbiata of termites (Warnecke et al., 2007), and on an acid mine drainage biofilm (Tyson et al., 2004). A large-scale study using the 454 technology and comparing the metabolic profile of several

different environments like marine, freshwater, coral-associated, terrestrial-animal-associated and many others is reported in Dinsdale et al. (2008).

Interestingly, there is connection between metagenomic and the recently emerging paleogenomics research which is described as the study of ancient DNA from extinct organisms (Hofreiter, 2008). Several projects have been reported that employ methods similar to typical metagenomic approach to obtain and characterized DNA from bone (Poinar et al., 2006; Green et al., 2006; Ramírez et al., 2009) or hair probes (Gilbert et al., 2007; Miller et al., 2009b; Zhao et al., 2009).

An up-to-date list of completed, draft and planned projects can be found at the GOLD database (`http://www.genomesonline.org/gold.cgi?want=Metagenomes`).

2.4.3 Workflow and Methods

In the following subsections, the technical and methodical strategies for the processing and analysis of environmental sequencing data will be described and discussed.

Sequencing Technologies

The environmental shotgun sequencing process produces a plethora of DNA reads that are the starting point for subsequent analyses like assembly or gene calling.

Many projects used the Sanger technology to generate the read sequences (e.g. Tyson et al. (2004), Rusch et al. (2007), and Warnecke et al. (2007)). The known benefits are the high sequence quality and the length of the read sequences which is more informative and therefore, attractive for a robust annotation or assembly. However, the cloning bias (Sorek et al., 2007) might prevent the sequencing of certain genes, promoters and viral DNA (Kunin et al., 2008). Due to this reason, high-throughput technologies like 454 (Roche) that avoid the *in vivo* cloning step slowly replace the Sanger sequencing (Blow, 2008b) in metagenomic studies. Since the launching of the Titanium series, the long 454 reads (up to 500 bp, see Table 2.3) are even suitable for (partial) ORF finding or efficient similarity searches.

Other platforms, such as Illumina's Genome Analyzer and ABI's SOLiD (see Section 2.2.2) with read lengths only up to 100 bp, have not yet been produced for metagenomic studies. But this may change soon: In an unpublished article (Mitra et al., 2010), the authors state that the usage of mate-pairs (paired-end reads) produced by the Illumina sequencer might be suitable for the taxonomical classification of environmental reads. Anyhow, the promise of next-next generation sequencing technologies (longer reads at lower costs, see Section 2.2.3) could be a great leap forward for metagenomics.

Assembly

As introduced in Section 2.3, based on sequence similarity, the assembly process combines the read sequences into longer fragments called contigs. Commonly, the sequenced reads obtained from an environmental sample consist of heterogenous DNA, i.e. the reads are derived from various microbial species that are, in general, only partially sampled. Because of the data fragmentation and the incomplete genome coverage, the assembly of environmental reads is disputable, since the likelihood of a *coassembly* of similar reads derived from different species is very high. The resulting chimeric contigs hardly contribute to an accurate community composition analysis. On the one hand, coassembled reads may actually originate from related genomes. On the other hand, it could be observed that even reads from phylogenetic distant taxa are assembled together because they may share an almost identical (conserved) stretch of sequence. Due to this reason, it is not recommended to assemble high complexity microbial communities with low coverages (Kunin et al., 2008; Mavromatis et al., 2007), because they consist of many different highly abundant species complicating the assembly of unique species populations. To circumvent the problem of chimeric contigs, an initial binning of the data, prior to the assembly, proved to be useful (Eisen, 2007): The separation of the reads into different phylogenetic groups, based on species-specific word frequencies (as reported in Warnecke et al. (2007)) or similarity searches, helps to unravel the subsequent assembly process (Binning methods are explained in more details in Section 2.4.3). Besides all the drawbacks and the mentioned barriers, the assembly of environmental sequences is actually reasonable for the discovery of genes (independent of the species origin).

The evaluation of the sequence quality differs from single genome assembly: Normally, the sequence coverage per base position indicates whether the inferred consensus nucleotide is likely to be correct. In contrast, the coverage in metagenomic assemblies may also reflect the distribution of the nonuniform sequence coverages of multiple organisms. As a consequence, most currently available assemblers, designed for individual genomes, fail to assemble metagenomic data correctly. For example, atypical high read coverages that normally represent repetitive regions in single genome assemblies, may consequently lead to a fragmentation of the contig sequence (Kunin et al., 2008). Due to this reason, researchers are not able to simply reuse their traditional assembly software pipelines. Instead, alternative paths have to be followed: for example, a comparative assembly can be performed using a known reference sequence. Another option is to merge the results of multiple assembler tools (hybrid assembly). So far, no specific assembler software for metagenomics has been published.

Community Composition Analysis

The exploration of the community composition (taxonomical analysis) aims at an association of sequence data (reads, contigs) to phylogenetic groups or even species and is therefore crucial for gaining knowledge about the species diversity within a sample.

One of the first molecular methods to count and classify microbial species used rRNA (ribosomal RNA) marker genes (Woese, 1987) and the rRNA-PCR technology. The background is that all cell-based organisms share the same rRNA genes (e.g. the 16S rRNA, a small subunit of the ribosome) with slightly different base sequences. Based on these differences and reference databases of known sequences, a species classification can be performed. Frequently used databases are GreenGenes (`http://greengenes.lbl.gov`) and the Ribosomal Database Project (`http://rdp.cme.msu.edu`). According to Kunin et al. (2008), several marker genes have been tested so far, such as 16S and 23S (rRNA), RecA (DNA repair protein), EF-Tu, EF-G (elongation factors) and HSP70 (heat shock protein). Conducting a rRNA-PCR on a metagenomics sample will ideally amplify all rRNA genes in the sample. After sequencing these genes, they can be placed onto a reference phylogenetic rRNA tree indicating the "phylotype" they belong to (Eisen, 2007). However, several issues complicate the application of this procedure: the rRNA-PCR approach suffers from amplification bias and from problems due to varying 16S copy number variations between species (Raes et al., 2007). Additionally, the diversity of viruses can not be detected by this technique because no conserved marker genes are known for these life forms. Obviously, incomplete reference databases biases the result of comparative analyses since only a few microbial lineages are present in current databases. In general, metagenomic data sets have a "low overall incidence of marker genes ($\approx 1\%$)" (Kunin et al., 2008). Thus, the classic 16S rRNA analysis represents only one option among others to explore and to characterize the community composition of a metagenomic sample.

Other taxonomical binning approaches are based on whole sequence comparisons of reads/contigs to genome and protein databases. Such methods employ the homology to known reference sequences to infer the phylogenetic origin of the anonymous, environmental DNA fragments. Typical similarity searches are conducted using the BLAST algorithm (e.g. BLASTN or BLASTX) (Wheeler et al., 2008b). Although this type of analysis uses more sequence information compared to single marker genes, it comes with some disadvantages that one has to be aware of: All similarity searches suffer from incomplete reference databases that may bias the result (see Section 2.4.4). Organisms that are not present in the database can not be found. As mentioned in Valdivia-Granda (2008), only 2% of the Sargasso sea sequences could be overlapped at 90% identity with sequences from existing databases. Further, commonly applied "Best-BLAST-Hit" analysis meth-

ods do likely lead to a misinterpretation of the data set because the best BLAST hit is often not the nearest phylogenetic neighbor due to, for instance, horizontal gene transfer across species borders (Koski and Golding, 2001). To address the problem of the interpretation of BLAST outputs, more sophisticated analysis methods had been reported in Huson et al. (2007) and Monzoorul Haque et al. (2009) (See Section 4.3 for an introduction of the MEGAN analysis pipeline.).

Another software tool, CARMA (Krause et al., 2008a), obtains a phylogenetic classification of reads by searching for protein families in the database PFAM (Finn et al., 2008). Note that a homology-based, taxonomical analysis is only reasonable for reads, since (chimeric) contigs can hardly be assigned to unique species because of the mentioned coassembly problem (see Section 2.4.3). In most cases, a reference database consisting of protein sequences (e.g. NCBI-nr) is used for comparison because coding sequences are more conserved than the plain nucleotide sequences. Hence, the length of the reads should not decrease below a certain threshold (e.g. 100 bp), because longer reads are more informative, i.e. long reads likely cover longer stretches of coding sequences than short reads, consequently yielding more true-positive hits (Richter et al., 2007).

A third binning approach makes use of the sequence composition differences between phylogenetically distant organisms. For all metagenomic fragments, the frequency of all oligonucleotide signatures (*k-mers*) is analyzed and clustered (unsupervised approach) and/or compared to a reference set of sequences (supervised approach). Unsupervised methods obviously have the benefit to not suffer from the mentioned database bias. (The computation of the GC content of DNA sequence is a simple sequence composition-based method with word size 1.). Existing tools use word sizes between 2 and 8 base pairs whereas longer words fairly increase the accuracy of the results but lead to higher computational costs. The best results can be obtained with a word size of 3-6 bp (Kunin et al., 2008).

Prior to the actual analysis, supervised methods train a model with related reference data to "learn" how to classify the metagenomic fragments. Exemplary tools following this principle are Phylopythia (McHardy et al., 2007) (support vector machine approach) and TACOA (Diaz et al., 2009). A semi-supervised method has been implemented using a "seeded growing self-organizing map", S-GSOM (Chan et al., 2008). It is comparable to Phylopythia but does not require any knowledge of completed genomes. Instead, it uses the flanking sequences of highly conserved 16S rRNA from the metagenome as seeds to classify other sequences based on their compositional similarity. A second semi-supervised method is CompostBin (Chatterji et al., 2008) which uses hexamer word sizes and other phylogenetic information to guide the clustering algorithm. Unsupervised methods do not require any training set: the anonymous fragments are clustered according to the word signatures in the sequence composition. Accordingly, this

method allows to predict phylogenetically novel species groups sharing the same sequence composition. A mentionable tool is TETRA, a web service for the analysis and comparison of tetranucleotide usage patterns in DNA (Teeling et al., 2004).

One major drawback of all sequence-composition-based methods is that they require very long input sequences (>1 Kbp). However, such sequences are only available if the environmental reads are assembled into contigs or if an adequate set of long reads could be obtained from Sanger sequencing accompanied with all the complications as discussed in Section 2.4.3. Hence, the composition-based binning of short reads (<500 bp) produced by next-generation-sequencing technologies is unfeasible.

Another development for a phylogenetic classification is called bar-coding. It is still experimental but already caused some controversy (Valdivia-Granda, 2008). The idea is too focus on individual motifs with lengths about 50 bp instead of using common marker genes. However, it is still unclear if this approach proves to be successful.

In conclusion, the taxonomical binning process is crucial, not only for an efficient assembly of fragments only belonging to single phylogenetic group, but also for the deep understanding of the community composition. However, all approaches show specific limitations, so a strategy complementing different methods is advisable.

Functional Analysis

To gain knowledge about the functional potential of a microbial community, two subsequent analyses are performed that are, in general, similar to procedures known from single genome studies: first, open reading frames (ORFs) have to be identified on the fragments (gene prediction). Then, in a second step, these genes are annotated, i.e. they are assigned to known biological functions.

The limiting factor that complicates the functional metagenome analysis is, again, the highly fragmented nature of the input data. Gene finding on short reads is challenging because they likely cover only partial ORFs. Additionally, the assembly of environmental reads into longer fragments (contigs) occasionally lead to chimeric assemblies or frame-shifts and therefore to incorrectly predicted gene boundaries (Kunin et al., 2008). However, there exist two strategies to identify stretches of sequence that code for proteins: The homology-based ("evidence-based") method finds ORFs by performing a similarity search against reference databases (e.g., by using BLASTX to search against the NCBI-nr database). This approach can be used for unassembled reads and contigs. A general drawback is the already mentioned database bias that limits the search only to the known fraction of sequences: novel genes without any database homolog can not be identified.

A second strategy for gene finding is the "ab-initio" method that aims

at detecting certain sequence features (like start/stop codons or the Shine-Dalgarno sequence pattern and others) to infer the location of an ORF. Tools employing this strategy are either trained on typical gene features (e.g., fgenesB, `http://www.softberry.com`) or self-trained on the actual data set (e.g., GLIMMER (Delcher et al., 1999)). A software called MetaGene (Noguchi et al., 2006) assumes "correlations between GC content and the di-codon frequencies" to predict genes. The successor of MetaGene, the program MetaGeneAnnotator, additionally contains models of prophage genes and ribosomal binding sites (Noguchi et al., 2008). Another tool, called Orphelia, provides specific pattern models depending on the read lengths (Hoff et al., 2009). Using machine-learning techniques, the authors claim to find genes on fragment lengths <300 bp. To enhance the accuracy of the gene prediction, gene neighborhood approaches are also employed: A search for gene patterns (described as "genes sharing a distinct genomic neighborhood, initiated by a user provided key gene") has been implemented in the program MetaMine (Bohnebeck et al., 2008).

Facing all the computational challenges one has to keep in mind that an environmental sample contains genes derived from a set of various organisms. Thus, in general, the detection of genes and their annotation is fairly independent of a unique species. Hence, to study the functional potential of a single phylogenetic group or even a single species, a taxonomical binning of the reads prior to gene calling is advisable. This has been accomplished, for example, in the analysis of the termite hindgut (Warnecke et al., 2007).

To annotate the set of ORFs, a homology search is performed comparing the potential genes with reference databases containing known and previously annotated sequences (e.g. PFAM (Finn et al., 2008), TIGRFAM (Selengut et al., 2007), COG (Tatusov et al., 2003), SEED (Overbeek et al., 2005), STRING (von Mering et al., 2005), Gene Ontology (Ashburner et al., 2000), NCBI-nr (Wheeler et al., 2008b)). According to Kunin et al. (2008), future homology-based annotation approaches will incorporate more context information (gene neighborhood, gene fusion) to enhance the quality of the results. Moreover, the authors recommend to use both, reads and contigs, to obtain a comprehensive picture of the biological functions contained a metagenomic sample. The final step of the functional analysis (not further discussed here) is the reconstruction of metabolic pathways and networks by mapping the protein coding genes onto reference pathway collections (such as KEGG (Kanehisa and Goto, 2000) and SEED (Overbeek et al., 2005)). A recent work has been published describing a "parsimony approach" for the inference of pathways for metagenomes (Ye and Doak, 2009).

2.4.4 Metagenomic Analysis: Open Issues

The fast-evolving research field of metagenomics has led to an enormous increase of sequence data. Over the last couple of years, researchers tried to

obtain first insights into earth's microbial biodiversity by sampling probes form various habitats like seawater, soil, air, biofilms, or human and animal intestinals. Comparing these diverse ecologic systems, researchers revealed a lot of differences regarding the taxonomic compositions and functional annotations (see Section 2.4.2). Still, the number of interesting and promising sample sites on our planet is high as suspected by the number of ongoing or planned sequencing projects (GOLD database: `http://www.genomesonline.org/gold.cgi?want=Metagenomes`).

However, after the first "gold rush" of the metagenomics era, the research field is currently in a crucial phase of reconsidering the design goals of metagenomic projects. People start to realize that some methods used to analyze and to compare metagenomic data need to be reviewed or improved. As being symptomatic for a young research discipline, common infrastructures and new computational methods as well as universal data standards have to be created or adapted.

This section briefly introduces some of the current findings about the database bias and the comparability of metagenomic projects.

Database Bias

The term database bias describes the imbalance of biological databases covering only a fraction of all present forms of life and biological functions. For example, one estimates that to date, less than 1% of all microbial species can be cultivated in the lab at all and thus, only a negligible part is actually represented in biological databases (Hugenholtz, 2002). Obviously, this imbalance, favoring only a few cultivatable species, biases homology-based methods which depend on sequence comparisons. As of 2009, 82% of the 3,000 reported bacterial genome projects only focus on Proteobacteria, Actinobacteria and Firmicutes (Kyrpides, 2009). Consequently, this imbalance is also applicable to protein and annotation databases. To address the problem of the incomplete representation of bacterial lineages, a project has been initiated in 2007, called GEBA (Genomic Encyclopedia of Bacteria and Archaea), that tries to systematically fill the gaps in the taxonomy by sequencing representative organisms of underrepresented clades. The importance of this ambitious project is undoubted, but it will take a while until a broad clade coverage is achieved.

Another issue about databases has to be kept in mind when interpreting or comparing analysis results: the content of major databases is ever-changing. Data is permanently added, removed or adapted which may lead to different results when comparing metagenomic sequences to the database at different times. This has been studied in a work of Pignatelli and coworkers (Pignatelli et al., 2008). They tested how taxonomical assignments and classification of ORFs change after conducting similarity searches on different database releases. Remarkably, significant changes could be observed

after the addition of newly sequenced genome sequences which emphasizes the urgent need for sequencing representative organisms within underrepresented lineages.

Needless to say that metagenomic studies also demand for updated databases which regularly incorporate the latest information and biological findings. One database which is used intensively in (meta)genomic studies but lacks recent updates since 2003 is COG (Tatusov et al., 2003; Kunin et al., 2008).

Comparability of Metagenomic Data Sets

As the number of completed metagenomic studies increases, a large amount of comparative projects are initiated to discover significant differences regarding the taxonomical and functional content of an environmental sample. However, the application of various analysis strategies and the usage of different classification systems often leads to the undesirable situation that a comparison based on the published data is rarely feasible.: Differences between the following list of experimental parameters may hamper direct comparisons of multiple data sets:

- probe sampling (filter size)
- DNA extraction methods
- sequencing technology (read length)
- amount of generated sequence ("coverage")
- applied read quality filter
- type of assembler software (parameters)
- community complexity of habitat
- type of gene prediction method (parameters/version)
- type of annotation method (parameters/version)
- type of reference database (release date)
- etc.

Considering this list, it becomes obvious that the results of metagenomic studies are considerably influenced by a multiple of factors and parameters which often are not well documented in publications or databases. For example, the final product of an assembly is a flat file without any evidence about the actual read coverage and quality (Kunin et al., 2008). By uploading only the sequence file into biological databases, valuable information needed for the interpretation of the data will not be available to the community.

As a consequence, the genomics and metagenomics community made an effort to propose principles on standardization and the specification of "metadata descriptors" using a controlled vocabulary. The MIGS/MIMS standard (Minimum Information about a Genome/Metagenomic Sequence) (Field et al., 2008; Kottmann et al., 2008) has been introduced "to provide a diverse set of descriptors for describing the exact origin and processing of a biological sample" (e.g. pH value, temperature, depth/height and time of DNA sampling, clone library information, geographical information, host habitat conditions, etc.). Providing this valuable information, the community is enabled to manage, structure and, most importantly, to systematically compare different data sets. In particular for environmental metadata, the Genomic Standard Consortium (GSC) proposes a set of terms defining specific habitat attributes and locations ("Habitat-Lite") (Hirschman et al., 2008). To also encourage researchers to use these standards, in 2009, the GSC has launched a genomic science journal "Standards in Genomic Science (SIGS)" (`http://standardsingenomics.org`) for the publication of genome sequencing projects.

So far, only few metadata descriptors are actually used in databases, such as IMG/M (`http://img.jgi.doe.gov/cgi-bin/m/main.cgi`) or CAMERA (`http://camera.calit2.net`). However, it is anticipated that the MIGS/MIMS standard will soon be adopted and incorporated into further sequence databases to facilitate the interpretation of (meta)genomic data.

Chapter 3

OSLay: Syntenic Layout of Unfinished Assemblies

3.1 Introduction

The sequencing technologies introduced in Chapter 2.2 produce large sets of sequence fragments, called reads. Depending on the size of the reads and the achieved coverage, a *de novo* sequencing of a whole genome is feasible. As explained in Chapter 2.3, during the assembly phase, the set of reads is assembled into longer sequences, called contigs. However, these contigs still need to be set into the context of the original genome, i.e. their actual relative ordering is unknown. By using, for instance, the mate-pair information obtained in the sequencing phase, contigs can be ordered and oriented into scaffolds. Ideally, to get rid of gaps separating contigs, additional sequencing runs are conducted producing more reads and, likewise, a higher sequence coverage to fill these gaps.

As a matter of fact, genome finishing and gap closure occasionally are the most challenging tasks in a genome assembly project. Reasons for the inability to reduce the number of gaps may be, for example:

- low level of sequence coverage
- features of the genome sequence (e.g. GC content)
- cloning bias (in case of Sanger sequencing) (Myers, 1999a)
- size and abundance of repetitive sequences in the genome
- short read lengths (as for next-generation sequencing technologies)

One approach to close gaps, is the synthesis of sequence primers that are needed to perform a Polymerase Chain Reaction (PCR) on the original genome sequence. The primers are located on different contig ends to “walk

across the gaps", i.e. to reach another contig end. This lab procedure is a costly and time-consuming process, thus, the goal is to minimize the the number of possible contig links. Such a *layout* of contigs is preferably obtained by analyzing the mate-pair information.

The intention was to provide a software tool that alternatively uses the genome sequence of a related organism to infer the relative order and orientation of the target contigs or scaffolds. In contrast to existing software (at that time), this approach is able to use also a fragmented reference sequence (unfinished assembly) to determine a layout for the target contigs and vice versa (Richter et al., 2007). In times of constantly growing sequence databases, this synteny approach is interesting for research groups performing sequencing and assembly of closely related strains.

The Optimal Syntenic Layout (OSL) algorithm has been implemented in a Java software package called OSLay (Optimal Syntenic Layouter).

3.2 Methods

The main idea of the OSL algorithm is permute and flip contigs (or scaffolds) of a target assembly while keeping the ordering and orientation in the reference assembly fixed (Richter et al., 2007). In case the reference assembly is given as single genome sequence, the algorithm can be applied in the same manner. This scenario facilitates the successful ordering of a target assembly.

Given a target assembly $A = \{a_1, \cdots, a_p\}$ consisting of contigs a_i and a reference assembly $B = \{b_1, \cdots, b_q\}$ consisting of contigs b_i of genome assembly of a closely related organism. Conducting a local sequence comparison (e.g. using BLAST or MUMmer (Altschul et al., 1990; Kurtz et al., 2004)) of both assemblies generates a set of matches $M = \{m_1, m_2, \cdots, m_r\}$. A match m is specified as $(a, x_1, x_2, b, y_1, y_2, o)$, with $a \in A$, $1 \leq x_1 < x_2 \leq |a|$, $b \in B$, $1 \leq y_1 < y_2 \leq |b|$ and $o \in \{-1, +1\}$, where $|a|$ denotes the length of a and x_1, x_2, y_1, y_2 denote relative nucleotide positions within contig a and b. Following these definitions, this means that a match m is called a *direct* match between the interval with indices $[x_1, \cdots, x_2]$ in contig a and $[y_1, \cdots, y_2]$ in contig b, of $o = +1$, or a match in which the sequence of the second interval is reverse-complemented, if $o = -1$. All contigs of both assemblies and their matches can be visualized in a comparison grid Z (see Figure 3.1). The positions and orientations of the single matches are later used to maximize the number of pairs of extended local diagonals. Crucial for this extension approach is the consideration of *informative* matches: these matches are "overlap"- or "containment" matches, but not "end-to-end" matches. Only informative matches guide the layout process of the target contigs. Consequently, both assemblies should not be too correlated, that is, contig boundaries should not coincide (i.e. contigs should not start

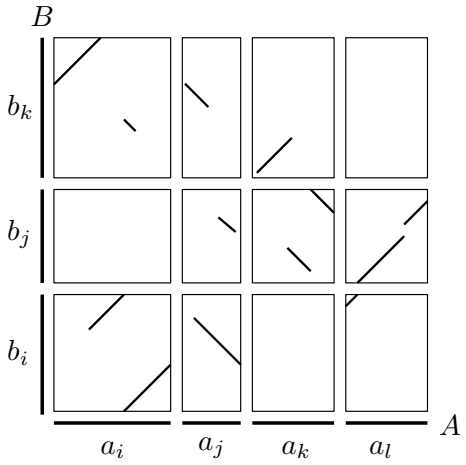


Figure 3.1: Comparison grid. Two assemblies A and B are shown together with their matches M in a comparison grid Z. Cell z_{kk} contains a direct match, whereas cell z_{ji} contains a reverse-complemented syntenic segment inferred by a local sequence comparison. Cell z_{ii} and z_{ji} contain informative matches because both matches overlap a contig boundary of assembly A. Thus, they may be used to obtain a local diagonal extension between contig a_i and a_j.

and end at equivalent positions). So, if a match (syntenic segment) overlaps contig boundaries in assembly A, then this may imply that the concerned contigs of A can be placed next to each other. For an example of informative matches refer to Figure 3.1.

Usually, BLAST matches are relatively short local matches, even if both genomes are closely related. However, prevalently, they lie close to a common diagonal. To facilitate the handling of those matches, a cluster of such matches is substituted by a summarized match m_s that reflects its orientation and total length.

The OSL Problem

The extension of match diagonals is enabled by the use of "anchor points" that indicate where a summarized match m_s hits (or would hit) a contig boundary (i.e. borders of a cell in the comparison grid). These points are specified as *connectors* $c = (y, w, o)$, whereas y is the position where m_s hits the border, w is the length and o represents the orientation of m_s. Consider two cells, z_{ij} and z_{kj}, in the same row of the comparison grid. Let C_{ij}^{right} be the set of all right connectors associated with z_{ij} and C_{kj}^{left} be the set of all left connectors associated with z_{kj}. Two connectors $c = (y, w, o) \in C_{ij}^{right}$ and $c' = (y', w', o') \in C_{kj}^{left}$ form a local extension if $y \approx y'$ and $o = o'$ (see Figure 3.2). Additionally, each extension is assigned a weight $w + w' - |h - h|$ which means that the sum of both connectors is penalized with their position

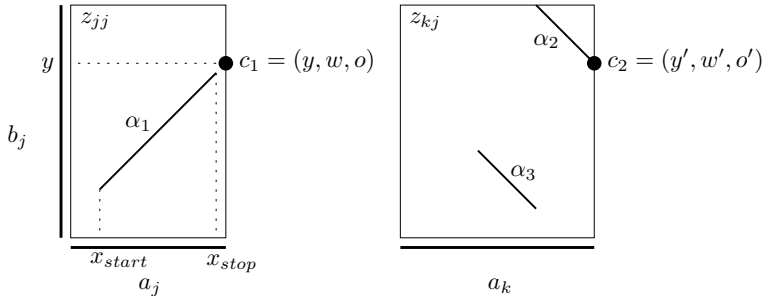


Figure 3.2: Connectors. Two cells are shown with three summarized matches α_1, α_2 and α_3. In contrast to α_3, the ends of α_1 and α_2 touch (or almost touch) the contig boundaries, thus generating two connectors c_1 and c_2. By flipping contig a_k, c_1 and c_2 denote the possibility of extending α_1 with α_2.

difference (see Figure 3.3 a). In other words, two connectors form a local diagonal extension in the comparison grid, if they are located at nearly the same position at both contig boundaries and if their summarized matches either have 45° or -45° slope. Note, that only either connectors at column or row boundaries are considered for an extension. This implies that the sorting of assembly A is independent of the sorting of assembly B and vice versa.

To order and orientate the complete target assembly, whole row or columns have to be considered. So, for all target contigs every possible side combination (left/right) is examined. Given two columns (rows), the score of matching two contig sides is the sum of weights of all local diagonal extensions obtained for cells contained in the two columns (rows). The OSL problem is then to find an ordering and orientation of columns (or rows) of the grid such that the sum of scores of pairs of adjacent column-sides (or row-sides, respectively) is maximized (Richter et al., 2007).

The OSL Graph

The OSL problem of detecting a layout for a target assembly can be reformulated as a graph theoretical problem. A layout graph $G = (V, E, \omega)$ is defined with a node set V, an edge set E and a weight function ω that assigns positive weights to all edges. For each contig or column a_i in the comparison grid Z, two nodes v_i^{left} and v_i^{right} are defined which represent the left and right side of a_i.

Let v_i^{δ} and v_j^{ϵ} be two nodes representing different columns a_i^{δ} and a_j^{ϵ} with $\delta, \epsilon \in \{left, right\}$. An edge e can then be defined connecting the nodes v_i^{δ} and v_j^{ϵ}, if the score S of matching the δ-side of column a_i with the ϵ-side of column a_j is greater than zero. In this case, the edge e is assigned the weight equal to S. Additional contig edges are added: for every pair of nodes v_i^{left} and $v_i^{right} \in V$ generated for a contig a_i, an edge is inserted into G (see

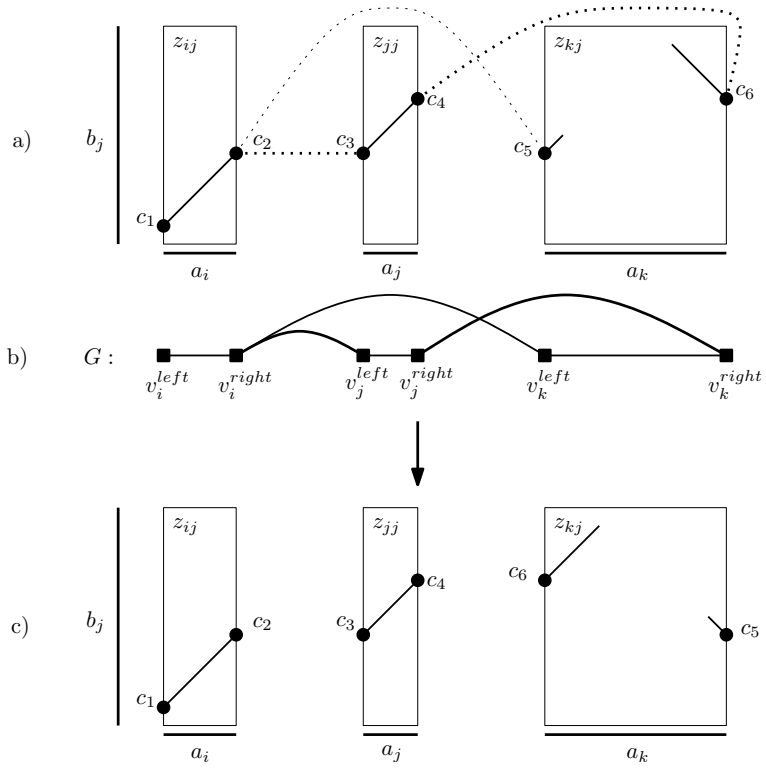


Figure 3.3: Layout graph. a) Shown are three cells with connectors guiding the layout process: Dotted lines represent possible side-by-side connections between three target contigs. b) For the possible contig side connections, the layout graph is built. Additional contig edges between contig nodes v_i^{left} and v_i^{right} for column a_i are inserted. c) After removing cycles, all contig nodes are visited in a greedy fashion to layout the contigs. Here contig a_k needs to be flipped to elongate the diagonals via the connectors c_4 and c_6.

Figure 3.3 b)).

The OSL Algorithm

The OSL problem is NP-hard (Richter et al., 2007). However, the problem of finding a contig layout can be solved efficiently. In Richter et al. (2007), the application of a *maximum weight matching* algorithm (Gabow, 1976) is explained. In the program, a greedy heuristic has been actually implemented that proved to be quite successful. The idea is to traverse the graph greedily, i.e. the edges having highest weights are always considered first when visiting all contig nodes. One precondition is crucial though: occurring cycles in the graph have to be removed prior to the graph traversal (see Figure 3.3 c).

A cycle is "broken" by removing an edge of minimum weight. In this way, each cycle loses less than half of its total weight. Because there may exist

another solution without cycles, breaking cycles may lead to solutions that have only half the weight of an optimal solution (2-approximation) (Richter et al., 2007). In practice, the experience shows that cycles occur only rarely in G, so the algorithm often produce "optimal" results.

Algorithm 1: OSL algorithm

Data: Assemblies A, B and matches M
Result: Syntenic contig layout for A
Let F be the set of contig edges
Construct the layout graph $G = \{V, E \cup F, \omega\}$
foreach *cycle* C *in* G **do**
| Delete the smallest weight edge
end
Greedily traverse G and visit all contig nodes
Infer and report the resulting contig layout for A

3.3 Implementation

The described, theoretical approach has been implemented in an interactive software tool called *OSLay* (Optimal Syntenic Layouter). The Java program is based on a sequence visualization engine provided by *CGViz* (Friedrichs et al., 2003). The processing pipeline of OSLay is shown in Figure 3.4. In addition to the BLAST file, OSLay reads in two FASTA files containing the target contigs and the reference sequence (assembly), respectively. Three dot-plot images are then produced displaying three different stages of the algorithm (Figure 3.5). The visual presentation of the data significantly enhance the quality of the results because the user gains immediate feedback when changing parameters. By using the navigational tools, the user may interactively explore the results.

OSLay provides a list of eight parameters that affect the outcome of the algorithm. For example, the maximal distance between summarized matches and contig boundaries to produce a connector can be set. Further, the maximal height difference between connectors of two contigs sides can be adjusted. Additional features are included for resolving potential assembly errors or evolutionary events that differentiate the target and reference genomes.

1. For instance, inserts may cause unmatched regions in the target contigs. In the case that foreign DNA (e.g. phage DNA) got only inserted into the target genome, there will be no sequence matches between the associated target contig and the reference sequence. If the insert is located near a contig end in the target assembly, connectors might be placed at a wrong position. By offering the option to trim un-

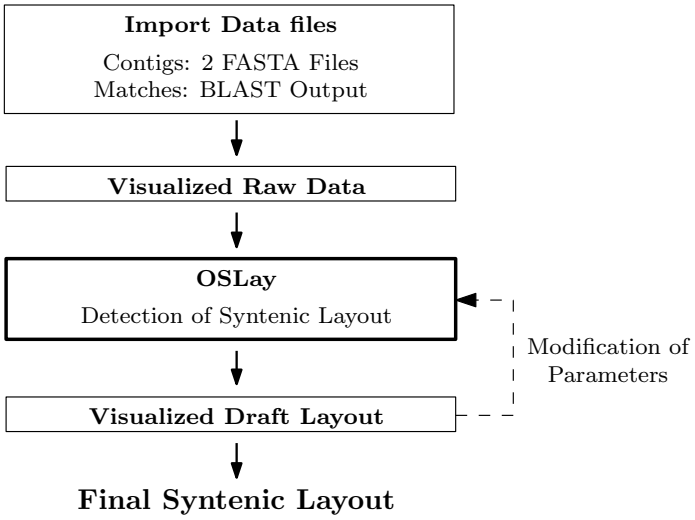


Figure 3.4: OSLay pipeline.

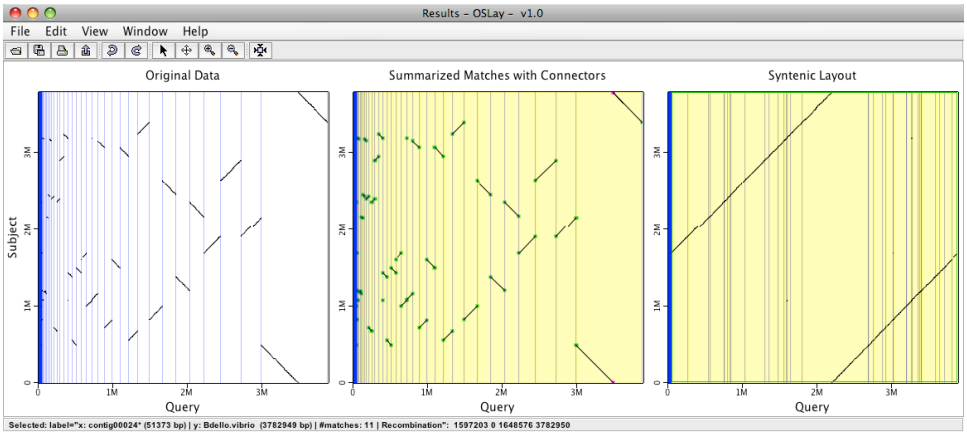


Figure 3.5: OSLay's graphical user interface. The left view (Original Data) displays the target contigs at the x-axis (ordered by size) and all matches that are obtained by a BLAST run against the reference sequence at the y-axis. The center view shows the same match distribution as the first view with one restriction: only summarized matches that give rise to connectors are shown. Connectors are represented as green (red) dots at the vertical (horizontal) contig borders. The last view depicts the final ordering of the target contigs (syntenic layout) after applying the OSL algorithm.

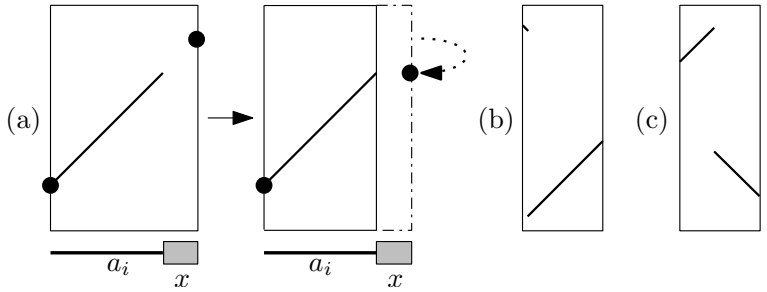


Figure 3.6: OSLay's features. a) The option to trim unmatched contigs ends allows to reset the location of biased connectors. b) Very short matches at contigs ends might mislead the contig ordering. They can optionally be disregarded. c) The dot-plot visualization helps to unveil recombination (and/or misassembly) events. OSLay can optionally mark these "broken matches" in the dot-plot view.

matched contig ends, OSLay allows to reset the connector positions by disregarding the unmatched end (Figure 3.6 a).

2. Bad sequence quality, misassemblies or other artifacts might hamper the correct positioning of the connectors. OSLay provides an option to ignore short matches at contig ends that give rise to "weak connector extensions" (Figure 3.6 b).

3. Repetitive regions that repeatedly map to contig ends may also produce misplaced connectors and, such, they may mislead the ordering process.

To facilitate the application of OSLay, we paid much attention to the integration of OSLay into typical assembly pipelines. Therefore many different output files are produced that are useful for diverse subsequent analyses. Besides a simple list of contig names in correct order put into supercontigs, OSLay exports a list of gap distances. This list contains the distances between succeeding contigs in the layout by measuring the connector height difference. This information is useful when designing primers on both contig ends. A third file lists all positions in the (finished) reference sequence where summarized matches have been mapped to. Further, a multifasta file is exported that contains all sorted and oriented contig sequences. Sequences are reverse-complemented (if required) and concatenated within each supercontig. Gaps between succeeding contigs are filled with N's. The last output is an ace file that can be imported into the assembly viewer software Consed (Gordon et al., 1998). Usually, this file is produced by the Phrap assembly software (`http://www.phrap.org`). In case of an existing ace file, OSLay reads in this file and modifies all read coordinates and other related values according to the computed contig layout. Alternatively, a new ace file can be generated from the scratch. Having an ace file at hand, the primer design

with Consed is simplified because contigs that are adjacent in the contig layout appear as neighbors in the ace file facilitating the primer design.

3.4 Results

In Richter et al. (2007), we presented contig layouts for two different strains of *Bdellovibio bacteriovorus*. The HD100 strain is a predatory Gram-negative bacterium (Rendulic et al., 2004) whereas the host-independet HDHI strain was evolved from strain HD100 (both about 3.78 Mbp). Both genomes are highly collinear. The HDHI target assembly consisted of 376 contigs. To prove OSLay's capabilities of sorting and orienting contigs in dependance of the number of reference contigs, different assembly phases of the reference strain HD100 have been considered (Table 3.1 and Figure 3.7).

Reference Contigs	**Supercontigs (Contigs contained)**	**Total Length of Supercontigs (cmp. to Total Genome Length)**
66	29 (260)	3,513,114 bp (93%)
27	14 (274)	3,697,854 bp (98%)
6	11 (277)	3,704,402 bp (98%)
1	1 (286)	3,748,836 bp (99%)

Table 3.1: Result statistics For the four assembly stages of the reference assembly HD100, the number of ordered target contigs is listed.

The results demonstrate that OSLay is capable of ordering and orientating the target contigs to obtain (partial) local match extensions, i.e. elongations of local diagonals. Figure 3.7 nicely shows that with increasing sequence coverage of the reference assembly, the number of super-contigs decreases, If the reference assembly is already finished, i.e. it consists of only one single contig, the detection of a syntenic layout is likely to be successful.

OSLay has already been successfully applied to several sequenced microbial genomes at Penn State University, USA and the Ludwig-Maximilian University in collaboration with the Max-von-Pettenkofer Institute, Munich, Germany.

3.5 Discussion

The next-generation sequencing technologies (NGS) produce significantly more bases for less dollars and in less time. Compared to Sanger sequencing, the achievable sequence coverage is multiple times higher depending on the number sequencer runs. However, the hurdles of the gap closure phase at the

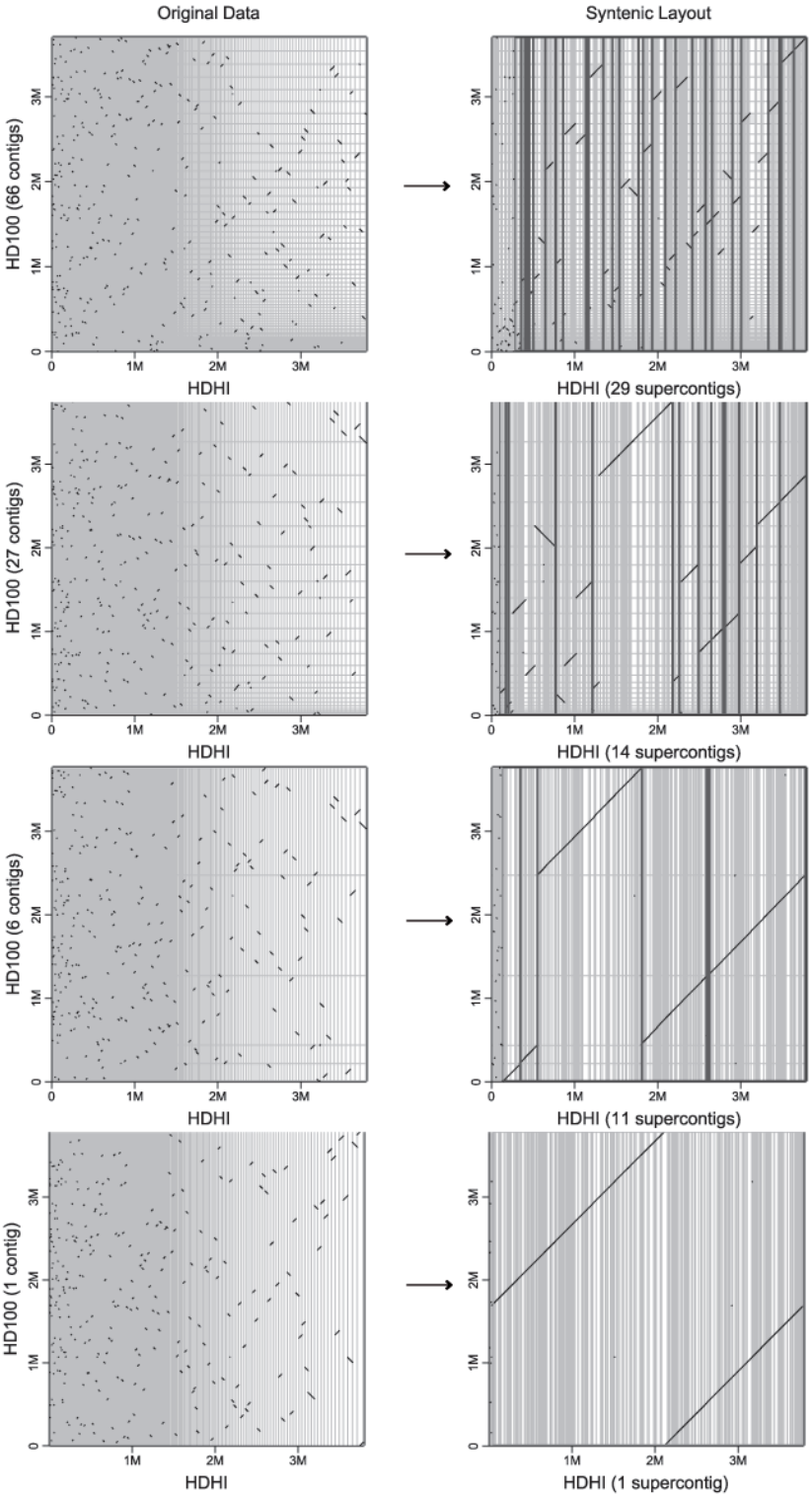


Figure 3.7: Contig layouts for the B. bacteriovorus strain HDHI. Shown are contig layouts of the HDHI assembly at four different assembly stages of the reference strain HD100. Depending on the number of reference sequences, the number of ordered and oriented contigs contained in supercontigs vary significantly. Eventually, the 376 contigs of HDHI can be almost completely laid out (bottom row).

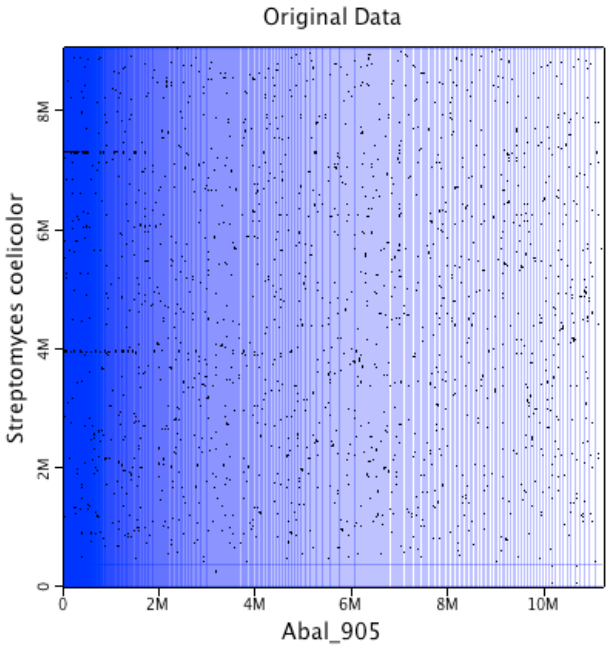


Figure 3.8: A. balhimycina vs. S. coelicolor comparison. Shown are the raw matches obtained from a pairwise BLASTN comparison. No syntenic segments, nor informative matches are apparent in the dot-plot.

end of an assembly still exist because of the tendency towards (ultra-)short read lengths (currently 35-500bp for NGS technology, Sanger: 1000bp)

Thus, for gap-closure in *de novo* sequencing projects, it is advantageous to have a closely related reference genome at hand. As more and more organisms or strains of a species are being sequenced these days, users may profit from the syntenic approach used by OSLay. Obviously, only closely related organism can be used for ordering and orientation contigs. Genomes from a more distant pair of taxa (e.g. from different orders or classes of the taxonomy) may have too few syntenic regions in order to compute sufficient local diagonal extensions. Such a situation arose when contigs of an (Sanger) assembly of *Amycolatopsis balhimycina* ($\approx$ 10-11 Mbp) should be laid out. The genome of the glycopeptide producer *A. balhimycina* had been partially sequenced by the Wohlleben group (Institute of Microbiology and Interdisciplinary Fields, University of Tuebingen, Germany). Unfortunately, no closely related organism or strain was present in the sequence databases. We then tried to detect a syntenic layout by comparing the 905 contigs to the finished genomes of the nearest neighbors in the taxonomy: *Nocardia farcinica* and *Streptomyces coelicolor*. The results were disappointing as shown in Figure 3.8. OSLay was not able to build a reasonable (or even partial) contig layout because no informative syntenic segments appeared in the dot-plot. In such cases, OSLay's approach will obviously fail.

Typical scenarios for the successful application of OSLay are hybrid-assembly approaches: As, for example, explained in (Goldberg et al., 2006) or (Reinhardt et al., 2009), a genome can be *de novo* sequenced by combining two different sequencing technologies, for example Roche's 454 with Illumina or Sanger with Roche's 454 technology (see Section 2.3.3, p. 32). Thus, the target genome is sequenced twice giving rise to two different read data sets. Since different sequencing technologies have dissimilar characteristics and error probabilities, hybrid assemblies are an attempt to overcome shortcomings of the one technology with the benefit of another technology. The differing contig sets obtained from the two approaches can then be used as input for OSLay. Hence, contigs from the two assemblies that overlap each other give rise to informative matches that enable the detection of a syntenic layout.

In general, repetitive sequences pose challenges during assembly, sequence comparison and, for example, the placement of connectors in the comparison grid. Consequently, it is recommended to mask repetitive regions by applying programs like RepeatMasker (Smit et al., 1996-2004). Also, genome rearrangements may lead to erroneous results.

As an outlook, future versions of the OSL algorithm could additionally integrate the mate-pair information derived from the assembly phase of the target genome. On the one hand, this additional linkage information could be used to confirm OSLay's result. On the other hand, OSLay's contig layout may help to determine erroneous mate-pair connections due to misassemblies. Hence, the usage of mate pairs for gap closure is still essential for successful sequencing projects. And it becomes even more valuable with the possibility for obtaining mate pairs with NGS technologies (e.g. Illumina and Roche's 454). Such a synteny-mate-pair approach could turn out to be a great enhancement for future scaffolding tools being part of automated assembly pipelines.

OSLay can be freely downloaded from `http://www-ab.informatik.uni-tuebingen.de/software/oslay`.

Chapter 4

Metagenome Analysis using MEGAN

4.1 Introduction

The analysis of metagenomes does not only require novel laboratory approaches but also efficient computational methods and software tools (see Section 2.4.1, p. 33). The impressive data flood of metagenomic studies spurs bioinformaticians to forge new paths in software and algorithm development. Regarding metagenomic data there are two common issues that have to be addressed by (computational) biologists when analyzing and processing the results: uncertainty and fuzziness. Uncertainty because in the beginning there is only minor knowledge about the content of the sample. Fuzziness because environmental samples are (not only at the first glance) considerably complex.

Over the last years, many software tools have been published addressing the two big questions, "Who is out there" and "What are they doing?", to investigate the species composition and the functional content of a metagenome (see Section 2.4.3, p. 36). The existing tools and methods can be classified into two groups: web services and stand-alone software. Web services comprise databases providing data derived from metagenomic projects and computation services. Examples for this are CAMERA and IMG/M (Seshadri et al., 2007; Markowitz et al., 2008). Both provide functionality to download and to compare existing data from environmental studies. A typical example for a computation service is MG-RAST (Meyer et al., 2008). Users are able to upload their individual data sets that are then analyzed (e.g. compared against each other) on a remote compute cluster. Remote computation of data using public web services is preferable for research groups without access to local high-performance computer clusters.

In contrast to those web services, stand-alone software has the advantage that (confidential) data can be processed locally and hence, does not need

to be transferred to anywhere else. Examples for such software tools are TACOA, Compost Bin, Phylopythia or MetaGeneAnnotator (Diaz et al., 2009; Chatterji et al., 2008; McHardy et al., 2007; Noguchi et al., 2008). One of the first metagenomic stand-alone software tools is MEGAN (Huson et al., 2007). It was first introduced as "GenomeTaxonomyBrowser" in Poinar et al. (2006). MEGAN uses a homology-based approach to study the taxonomical and functional content of environmental sequences. As of August 2009, it is still the only software available that allows users to efficiently explore metagenomic data at all levels of detail on a standard laptop computer. Furthermore, MEGAN's design puts a focus on a user-friendly data visualization. Many projects have already applied MEGAN in their analyses (e.g. Qi et al., 2009; Miller et al., 2009b; Urich et al., 2008; Frias-Lopez et al., 2008).

To internally store and access data parsed from a metagenome BLAST result file, usually several hundreds and even thousands of MB of memory are required. To enable the usage of MEGAN on standard computers (with limited RAM), an open file format, called *ReadMatchArchive* (RMA), has been recently developed (Huson et al., *unpublished*). A RMA file efficiently stores the complete read and match information in an incrementally compressed, binary format. Hence, MEGAN is able to directly access data within the RMA file without allocating too much space in memory. Consequently, the total memory usage is kept to a necessary minimum. Due to the compression, the typical size of a RMA file is $10\% - 40\%$ of the original BLAST file. In the near future, the release of a public Java and C++ library is planned.

In this chapter, MEGAN's main features and algorithms for the taxonomical and functional analysis are described.

4.2 Preliminaries

Let $G = (V, E)$ be a directed graph. A path on V is a sequence of nodes $(v_0, v_1, \ldots, v_k) \in V$ such that $(v_{i-1}, v_i) \in E$ for all $i = 1, \ldots, k$. Such a path is a *cycle* if $v_k = v_0$. We call v_0 the *origin* of the path, $(v_1, \ldots, v_{k-1})$ its *intermediate* nodes, and v_k its *end*. The *length* of path $(v_0, v_1, \ldots, v_k)$ is k. The *shortest/longest distance* from u to v is the shortest/longest length of a path with origin u and end v. A directed graph is acyclic (called DAG) when it does not contain any cycle. A connected graph without cycles is a *tree*. The *leaves* of a graph are its nodes of out-degree 0. The nodes that are not leaves are called *inner* nodes. A DAG is *rooted* if it contains one *root*, i.e. one node with in-degree 0. A node $v \in V$ is called child node of $u \in V$ if $(u, v) \in E$. In this case, u is the parent node of v. For $u, v \in V$, a node w is called the *lowest common ancestor (LCA)* if w is an ancestor of both u and v and if w is the end of a path with the longest distance from the root.

Note that, in contrast to trees, DAGs may contain more than one LCA for a set of nodes.

4.3 Taxonomical Analysis

As explained in Section 2.4.3, the estimation of the community composition is one of the first steps during a metagenome analysis. It involves the identification of taxa (organisms) to obtain a taxonomic profile of the environmental sample. Starting point is the collection of sequenced fragments (reads) generated by various sequencing technologies like Sanger (Sanger et al., 1977) or 454 (Margulies et al., 2005) (see Sections 2.2.1 and 2.2.2). The classification or separation of read sequences into distinguishable sets is also called "binning" (see Section 2.4.3).

MEGAN uses a homology-based method to bin reads taxonomically. The software infers taxon assignments by comparing the given reads with known sequences contained in databases. The assigned reads are then placed into the tree graph of the official NCBI taxonomy (Wheeler et al., 2008b) (alternatively, any other taxonomic system might be used). The NCBI taxonomy contains the names of all organisms (microorganisms and eukaryota) that are represented in the GenBank database (`http://www.ncbi.nlm.nih.gov/Genbank`) with at least one nucleotide or protein sequence. Overall, about 460.000 taxonomic nodes are contained (as of June 2009). Taxa are organized into different ranks of the taxonomy, e.g., Kingdom, Phylum, Class, Order, Family, Genus, down to the species level.

In a preprocessing step, the set of read sequences is compared against a database of known DNA or protein sequences. For example, BLAST (Altschul et al., 1990) in conjunction with NCBI databases may be used to find matches between reads and DNA sequences (BLASTN against NCBI-nt) or between reads and protein sequences (BLASTX against NCBI-nr). (Note that MEGAN is not committed to any particular comparison method or database.) The sequence comparison step is computationally intensive and, thus, very time consuming: In case of conducting a BLASTX comparison against the NCBI-nr database, each read (e.g., assume more than one million of reads in a typical data set) has to be translated in all six possible reading frames and compared to over seven million entries in the NCBI-nr database. This computation usually has to be performed on a compute cluster (grid-computing) to obtain a result in reasonable time. The idea is that the time-consuming BLAST comparison has only to be computed once because MEGAN itself provides functionality to filter the BLAST matches (e.g., by using the *bit-score* parameter). The result of the preprocessing step is then analyzed using MEGAN. Remarkably, the MEGAN analysis of large data sets can be carried out on a standard computer ("Laptop Analysis"). Therefore, the software uses its own efficient data format (RMA-format).

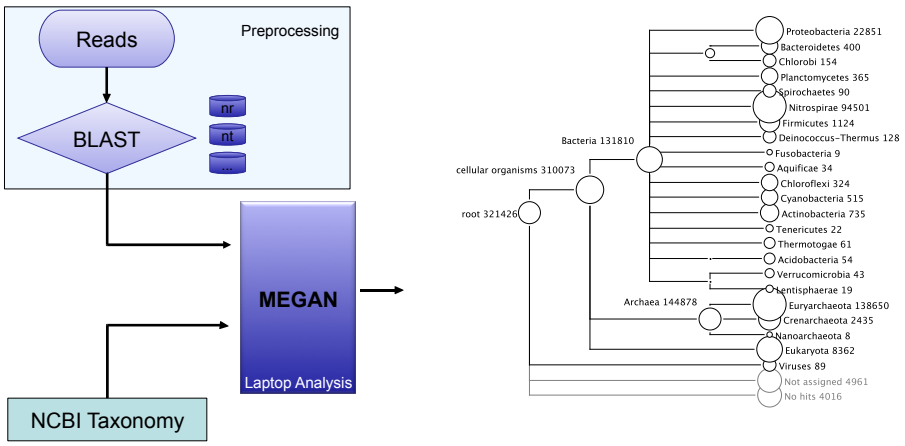


Figure 4.1: MEGAN pipeline and taxonomical assignment. During a preprocessing step, read sequences are compared against a database of known DNA or protein sequences. The BLAST result file and the NCBI taxonomy read in by MEGAN to compute a species abundance profile. The reads are placed onto nodes of the taxonomic tree using a LCA algorithm. The node sizes (and numbers) correspond to the quantity of assigned reads. The "No Hits"-node contains all reads for which no database homologue could be found. The "Not Assigned"-node contains filtered reads with poor bit-scores. (Data set derived from (Tyson et al., 2004))

The result file is interpreted by MEGAN to extract the organism names for each read and all its BLAST matches (see Figure **??**).

In case all BLAST matches of a read are a derived from a single species, this read can be directly assigned to that organism node in the taxonomic tree. In contrast, if matches indicate that a read might be assigned to more than one organism, MEGAN applies a *lowest common ancestor* (LCA) algorithm to resolve a representative taxon. Hence, the read will be assigned to a node at a higher taxonomic level which is the parent node for all organisms actually obtained for the matches (see Figure 4.2 for a simplified example of the LCA algorithm).

By applying this LCA approach, the assignment of reads to taxa reflects the level of conservation of the sequence (Huson et al., 2007). It further helps to avoid false-positive assignments since one considers not only the best match but all matches of a read that passed the bit-score filter of MEGAN. In the original paper (Huson et al., 2007), it has been demonstrated that this approach is able to taxonomically classify reads of different sizes, derived from different sequencing platforms.

To support the inspection and comparison of two or more data sets at the same time, MEGAN is able to process and visualize several data sets simultaneously. Still based on the taxonomy view, different node visualizations

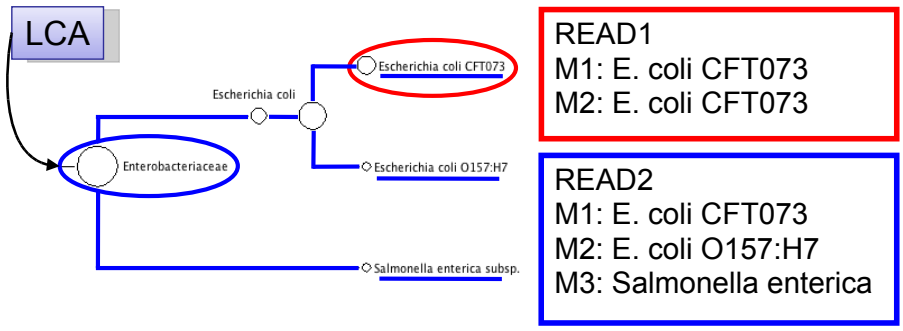


Figure 4.2: LCA algorithm. $Read1$ (red) has two matches $M1$ and $M2$, each of which indicates *E. coli CFT073* as homologue organism. Then, $Read1$ can be directly assigned to E. coli CFT073. In contrast, the matches of $Read2$ (blue) can be assigned to three different organisms. In this case, $Read2$ is assigned to their LCA.

(e.g. pie charts or heat maps) can be selected to detect differences between the taxon profiles. If the data sets vary significantly in the number of reads, the user may conduct a comparison with relative instead of absolute (read assignment) values.

4.3.1 Visualization of Taxon Profiles

Introduction

The default view that is displayed after computing the taxonomic assignment of the set of environmental sequences is the tree view (see Figure **??**). Based on the official NCBI taxonomy, the visual appearance of taxa organized in a hierarchical tree enables to obtain a first impression of the distribution of the assigned reads. The node sizes indicate the number of reads being assigned to a specific taxon. By scrolling the view or by collapsing/expanding and zooming the set of nodes, the user may inspect the taxonomic profile at different levels of detail level and taxonomic ranks.

To extend its visual capabilities, a comprehensive chart feature has been added to MEGAN. The intention was to provide the user with a configurable chart tool that allows the visualization of the taxon profile(s) as bar and pie chart. On the one hand, these views allow the presentation of the data in a more condensed and tree independent way. On the other hand, the charts can be exported as high quality (vector-based) images for presentations and publications.

A further goal was to implement a general "MEGAN2Chart" interface that allows to easily incorporate new chart views for all kinds of analysis purposes. For example, the evaluation of the microbial attributes (see Chap-

ter 4.4) or the COG classification (see Chapter 4.5.1) greatly benefits from the chart visualization.

Result

An existing library was used for the bar and pie chart drawing that provides a lot of convenient methods for presenting charts in 2D and 3D mode (`http://www.jfree.org/jfreechart`). The idea was to build a highly flexible chart tool as part of the MEGAN software. To illustrate the number of read assignments in a chart, the user selects taxa of interest in the taxonomy view. Only these taxa are then displayed as bars or pie sections. Since the selection of taxa may change during an analysis, it is possible to add/hide taxa later to/from the chart.

To improve the flexibility for the user, each MEGAN project may have more than one chart window opened. This is useful if different sets of taxa should be visualized separately. In case of a comparative MEGAN analysis, read assignments for each project are visualized simultaneously in the bar or pie chart. The taxonomy and the chart view are synchronized, i.e. if a data set changes within the taxonomy view (due to new parameter settings), the chart view is notified to update its chart.
Here is a brief list of the implemented functionality:

- Data sets and taxa can be hidden from the view.
- Data sets and taxa can be sorted and renamed.
- Displayed taxa can be filtered to show only leaf or inner nodes (depending on selected taxa).
- Charts can be customized in a lot of ways: font style, font color, chart color, 2D/3D mode, the labeling of the axis and the chart arrangement is adjustable.
- Chart views are scalable.
- Images can be exported to several common image formats (.jpg, .png, .pdf, .svg).

Figure 4.3 shows an example of a pie chart visualization of a single data set whereas Figure 4.4 shows a bar chart illustrating the taxon profile for two data sets. In Huson et al. (2009) we used the chart tool to compare a marine data set (145.000 reads of the *Global Ocean Survey* (Rusch et al., 2007)) with a soil data set (140.000 reads) (Tringe et al., 2005). In Figure 4.5, the differences of the species abundance between both (normalized) data sets can be quickly detected.

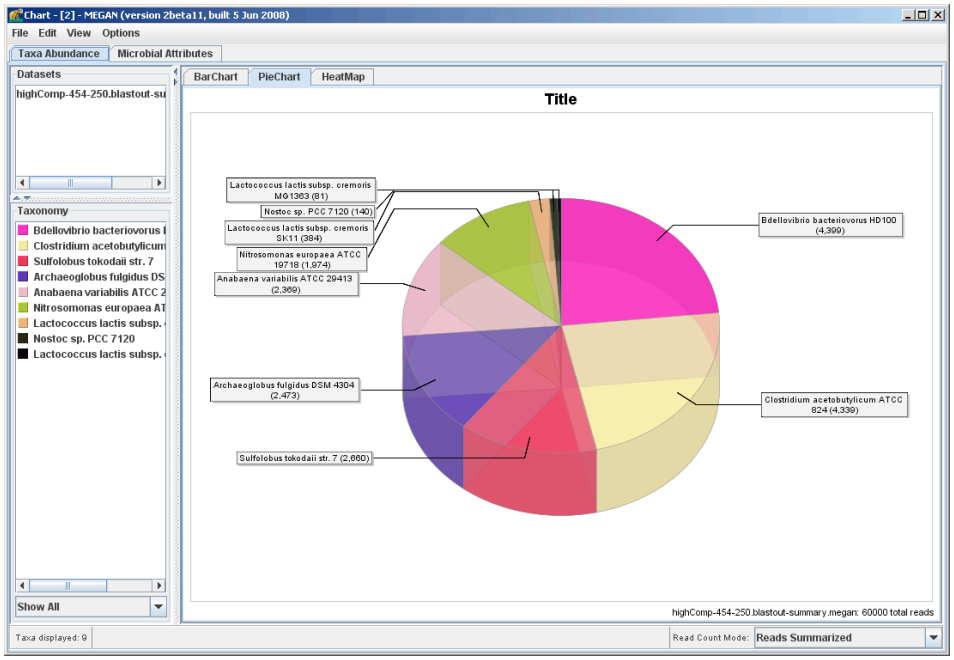


Figure 4.3: Pie chart. A 3D pie chart is shown for a single data set. The upper-left list displays the currently loaded data sets. The bottom-left list contains all taxa that have been selected in the taxonomy view. Using the tabs at the top of the view, the user may switch between different chart types. By double-clicking entries in the lists, single data sets or taxa can be temporarily hidden in the chart. Additionally, the entries can be sorted either by taxon name or by the amount of assigned reads. List entries can also be dragged and dropped to change their sequential order.

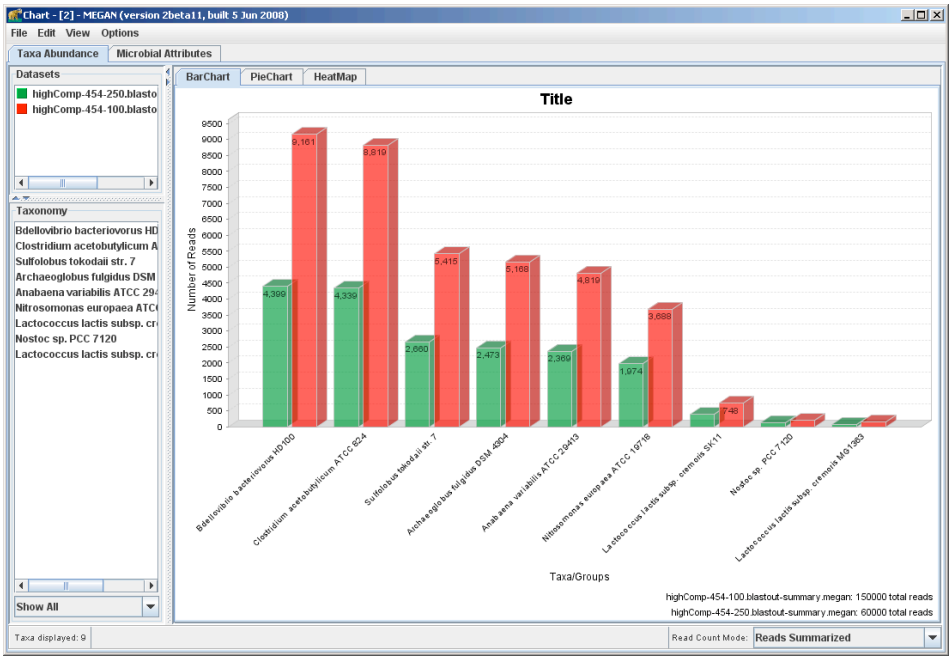


Figure 4.4: Comparative bar chart. Taxa of two data sets (green, red) are displayed in a single chart. Colors can be freely chosen from a color palette.

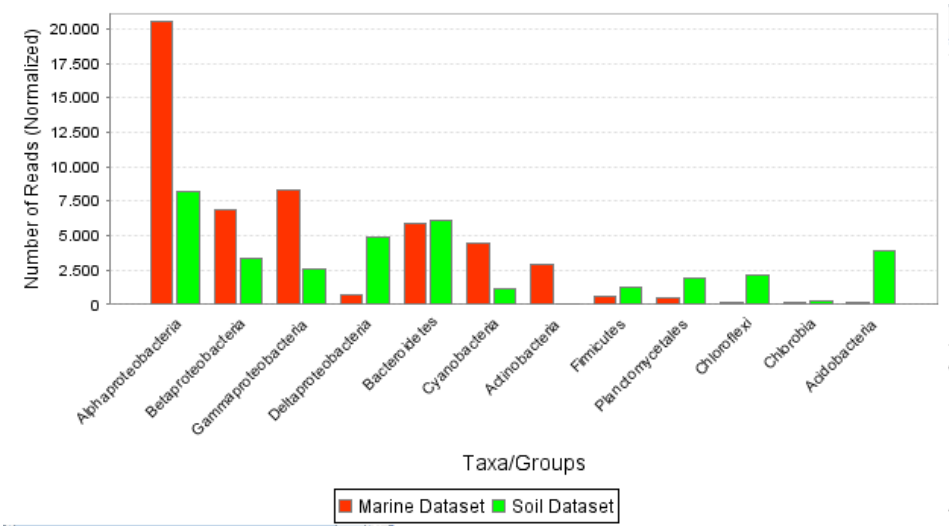


Figure 4.5: Comparison chart. Comparison of a soil and a marine metagenome at different ranks of the taxonomy. Figure taken from Huson et al. (2009)

Discussion

In general, visualization of biological data significantly helps to reveal structures and patterns that may otherwise be difficult to spot. Especially in the research field of metagenomics, the flood of data has to be categorized and visually organized to obtain an overview about the composition of an environmental sample. By providing a chart viewer, MEGAN extends its accessory of visual inspection tools.

Unfortunately, not all described features of the chart viewer are still incorporated into current versions of MEGAN. The primary motivation during the implementation phase of the chart tool was to provide the user the full control over the displayed data sets and its taxa. For instance, the user was able to open several chart viewers for a single project. Any nodes could be selected, independent of their ranks and depth of the nodes in the tree. Taxon and data set could be renamed without changing the original file names or taxonomy nodes. Unfortunately, we noted that the usability of the software suffered from the initially planned broad flexibility. The reason was that the strict consistency between the taxonomy viewer and the chart tool was affected. As a consequence, the handling of the user-interface became complicated. After reviewing the functionality of the chart viewer, only essential features were kept for future software releases. To regain simplified functionality, is was decided to provide only one chart viewer window for each single project.

Another improvement is that the user does not priorly have to select nodes in the taxonomic view to draw a chart. Instead, all current leaf nodes are automatically added to the taxa list of the chart tool when opening the chart tool. By collapsing the tree at the desired level, the user determines which nodes are displayed in the chart. By restricting the user only to leaf nodes, situations may be avoided that, for example, taxon nodes which contain each other do appear in the same pie chart.

Currently, MEGAN uses the chart functionality to illustrate the distribution of read assignments subject to the taxonomical classification as well as to Clusters of Orthologous Genes (COG), Gene Ontology (GO) (Chapter 4.5.1), and to microbial attributes (Chapter 4.4). The current state of the chart view is shown in Figure 4.20 (p. 85).

4.4 Microbial Attributes Classification

4.4.1 Introduction

The taxonomical binning of environmental reads can be further refined by looking at the identified taxa from another perspective. The NCBI web site provides a "Prokaryotic Attributes Table" (`http://www.ncbi.nlm.nih.gov/genomes/lproks.cgi`) that lists microbial organisms and their

various physiological features. It contains only those organisms from which genomic information is present at the NCBI databases. (As of August 2009, 2871 microbial genomes are included.)

The classification is partitioned into five sections (`http://web.ncbi.nlm.nih.gov/genomes/static/gprj_help.html#prok_attributes`): *cellular features*, *environment*, *temperature*, *pathogenicity* and *disease* (see supplemental Tables C.1, C.2, C.3, C.4, p. 117 ff). These attributes describe the detected taxa physiologically and they directly enable the user to obtain an overview of their habitat preferences. Some attributes are determined via a *controlled vocabulary* of predefined tags that simplify the categorization of the data. For example, the gram stain of a bacterium can be either "positive", "negative" or "unknown". The result section describes how MEGAN makes use of this classification scheme.

4.4.2 Results

The official classification and nomenclature of the "Prokaryotic Attributes Table" has been integrated into MEGAN as a separate inspection tool. At start-up, MEGAN reads in the attributes table which is included in MEGAN's installation package. After computing the taxonomical read assignment, the user may open the *Microbial Attributes Window*. Since only discrete organisms (species) are listed in the attributes table, only leaf nodes (at species level) found at the Bacteria and Archaea subtree can be classified.

The result of the attribute binning can be inspected either by browsing a tree-like structure or by displaying the summarized numerical values in a chart view. For example, Figure 4.6 displays the attribute data for the acid mine metagenome (Tyson et al., 2004). Each category (e.g. Oxygen Requirements) is represented as node in tree view. Each category node can be expanded to show its attribute nodes (e.g. microaerophilic in case of the oxygen requirements category). If a taxon has been detected at the species level by MEGAN and if this organism is known to have a certain attribute, it is inserted as child node beneath this property node. Selecting this organism displays a summary of its known attributes.

To obtain a visual summary of the attribute classification, each category based on controlled vocabulary, can be visualized as pie or bar chart. In our paper about methods for comparative metagenomics (Huson et al., 2009), we presented a chart view as summary of the attributes analysis for a soil metagenome (Tringe et al., 2005) (see Figure 4.7).

4.4.3 Discussion

Using the microbial attributes window of MEGAN enables to obtain an broad overview about the physiological and environmental features of microbial organisms within metagenome samples. The included chart view

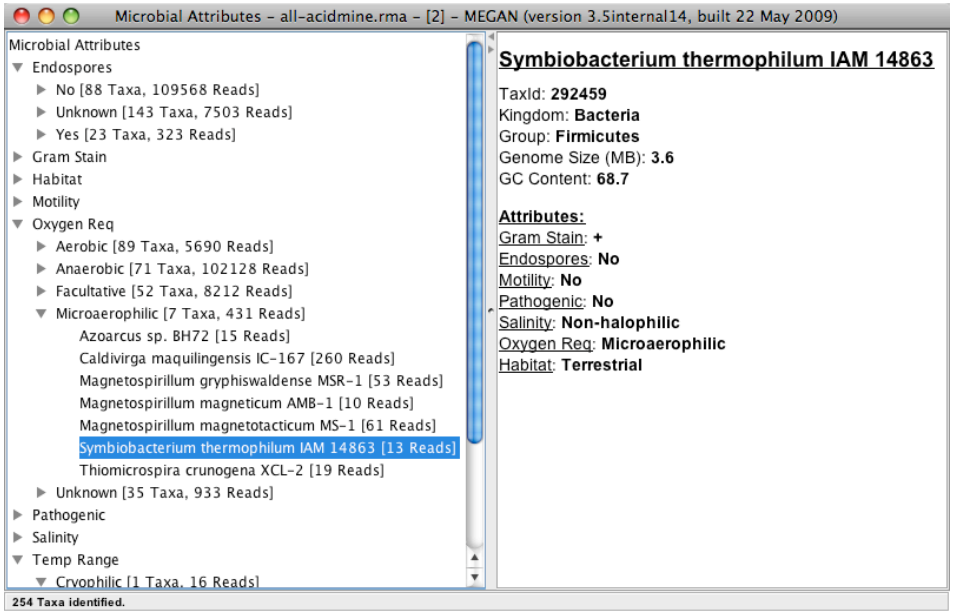


Figure 4.6: Microbial attributes window. This window provides a tree like structure to present the classification of microbial attributes. By selecting an organism, a summary of its attributes is displayed together with further information (e.g., taxonomical classification, genome size and GC content). At total, 254 microbial organisms could be classified for the acid mine metagenome (Tyson et al., 2004).

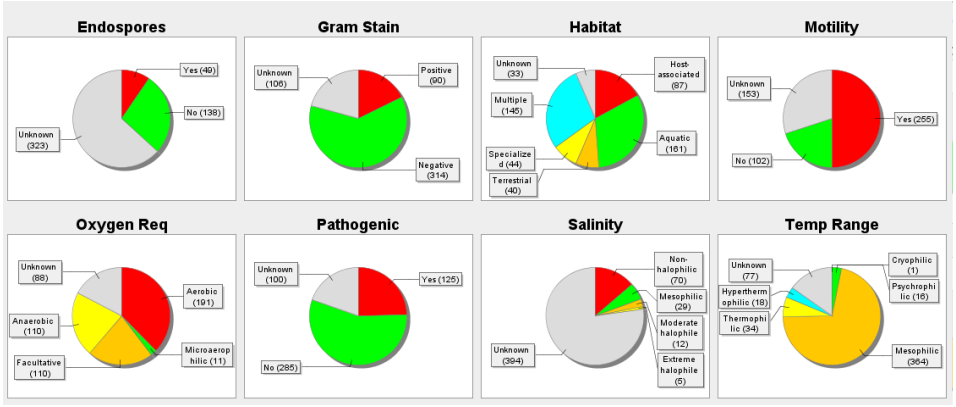


Figure 4.7: Microbial attributes chart. As an alternative to the tree view, a summary of the attibutes analysis is displayed using the chart tool. Here, the analysis of a subsample of a soil metagenome (Tringe et al., 2005) using multiple pie charts is shown.

summarizes the data, whereas the tree view allows for a comprehensive inspection of important characteristics of each microbe.

Apart from the software implementation, it has to be noted that this approach suffers from two issues that evidently bias the results. First of all, it is obvious that only a small subset of all actually existing species of a metagenome sample can be identified by MEGAN. Further, only species (single organisms) can be used for the assignments of microbial attributes. Assignments to inner nodes in the taxonomy are disregarded. Due to MEGAN's LCA approach (see Section 4.3), the amount of leaf (species) nodes may be rather small compared to the total number of nodes in the taxonomy. For example, it is not surprising that the analysis of the acid mine metagenome (shown in Figure 4.6) consists of only 254 detected organisms that can be classified by this approach, although even in low-complexity habitats, one would clearly expect a higher number of different organisms.

Secondly, studies have shown that the constitution of the attributes table and its underlying data may not be yet fully appropriate for comparative analyses. For example, in Huson et al. (2009), we compared a marine with a soil data set using the microbial attribute window. Although, both samples have been derived from distinct environments, the profiles of attributes differ only insignificantly. The reason for this is most likely due to the database bias: Still, only a minor set of all existing microbial species is considered in common (attributes) databases. (See Section 2.4.4, p. 42).

Thus, major efforts have to be made to populate biological databases by discovering and studying new microbial organisms. For example, new (sequencing) technologies will help to overcome this limitation. The development of the last months seems to be promising: at the end of the year 2008, about 1500 organisms were listed in the official NCBI attributes table. Currently, the amount of characterized organisms could be almost doubled (2871, as of August 2009). This rapid increase is expected to last or even to scale up in the next couple of years. Hence, in prospect of more complete databases in the near future, the microbial attributes tool included in MEGAN provides a detailed insight into the microbial characteristics of an environmental sample.

4.5 Functional Analysis

4.5.1 Introduction

MEGANs strenghts are its high-performance, memory efficient processing and analyzing capabilities for the taxonomical classification of reads. However, as already stated in Section 2.4, both, the taxonomical as well as the functional binning of environmental sequences are fundamental steps in a typical metagenomic project.

The first approach to integrate a functional analysis into MEGAN was

based on NCBI's Clusters of Orthologous Groups (COG) classification (Tatusov et al., 1997; 2003). COGs have been developed to cluster annotated genes into functionally related groups. Due to their simple structure and adaptability, COGs are widely accepted and applied in the scientific literature. Standard homology searches using BLASTX report COG identifiers that can be easily used to classify the query sequences. Since MEGAN imports such BLAST result files, a straight-forward COG analysis of the environmental reads is automatically performed. The results can be visualized in the COG chart view.

Our aim was to extend MEGAN by incorporating a more comprehensive analysis tool for the functional analysis of read sequences. Similar to the taxonomical classification, our intention was to provide a fast, (memory-) efficient and user-friendly tool to visualize and structure functional groups of gene products. The idea was not to design another gene calling or ORF finding program (there are already lots of others) but rather to provide the user with a structured overview of the functional gene annotation for the set of reads based on the BLASTX hits (e.g. after a homology search against NCBI-nr). Hence, the new module has been integrated into MEGAN following its data processing principle ("BLAST only once and filter and analyze the results afterwards.").

We decided to use the Gene Ontology (GO) (Ashburner et al., 2000) as classification structure for binning environmetal sequences. GO is regularly updated and has widely been used in many biological databases, gene expression and annotation studies, and it is "the most successful example of systematic description of biology" (Rhee et al., 2008). Unique definitions of gene products and their (hierarchical) relationships among each other facilitate the analysis of functional data. Further, a lot of mapping files among different databases and GO exists and could therefore be used for the comparison between different data sets and projects.

In the light of these benefits, it is remakable that to date, only very few metagenomic projects have made use of this valuable resource (e.g. Yooseph et al., 2007; Szczepanowski et al., 2008; Poretsky et al., 2009) Explanations could be that GO is originally based upon eukaryotic gene annotations (e.g. derived from fruit fly, yeast, mouse, or rice). However during the last years, more and more prokaryotic gene annotations have been added (see `http://www.geneontology.org/GO.current.annotations.shtml`).

Gene Ontology

In general, an ontology is a formal way to represent knowledge in a structured and well-defined manner (Bard and Rhee, 2004). A set of entities or so-called terms are associated with unique identifiers linking to external databases. Terms are (hierarchically) connected through certain types of relationships. In contrast to common databases that solely store data objects,

the purpose of an ontology is to maintain and structure data about certain fields of knowledge.

The Gene Ontology is one of the best known bio-ontologies. It adresses the need for consistent descriptions of gene products in different databases (`http://www.geneontology.org`). Therefore, GO provides three sets of structured vocabularies (ontologies) that describe gene products in terms of their associated biological processes, molecular functions and cellular components. As of August 2009, GO comprises 28089 terms (17025 biological processes, 2430 cellular components, 8634 molecular functions and 1423 obsolete terms). Each GO term is associated with a list of annotated genes derived from more than 40 experimental organisms including animals, plants, fungi, bacteria and viruses (Bard and Rhee, 2004). Such, users are able to find all proteins for a specific GO term or, vice visa, all GO terms for a certain protein. Although the annotated gene products are derived from a couple of model organisms, the biological vocabularies itself are cross-specific. Accordingly, the terms describing different elements of molecular biology are shared by among almost all life forms.

Three ontologies have been developed to describe different attributes of gene products. *Molecular Function* describes what a gene product does at the molecular level. Examples are term definitions like "protein binding" or "kinase activity". *Biological Process* describes a biological objective that is assembled of molecular functions. Examples are "response to stress" and "signal transduction". *Cellular Component* refers to a place in the cell where the gene product is usually found. Examples are "nucleus" and "cytosol".

The terms in GO are hierarchically organized defining parent-child relationships where children can have more than one parent. Therefore, each of the three ontologies can be represented by a directed acyclic graph (DAG) that contains the terms (as nodes) and the relationships among them (as edges). There are five types of relationships (referring to `http://www.geneontology.org/GO.doc.shtml`):

- `<is_a>`: This refers to the case when a child is an instance of its parent. For example, X `<is_a>` Y means that X is a subclass of Y.
- `<part_of>`: This refers to the case when a child is a component of its parent. For example, X `<part_of>` Y means that whenever X is present, it is always a part of Y, but X has not to be present.
- `<regulates>`, `<positively_regulates>`, `<negatively_regulates>`: These relations describe interactions between biological processes and other biological processes or molecular functions. For example, X `<regulates>` Y means that X modulates the occurence of Y.

Figure 4.8 shows a simple example of a DAG with different relations. To work with GO, several files and file types are provided at the GO website

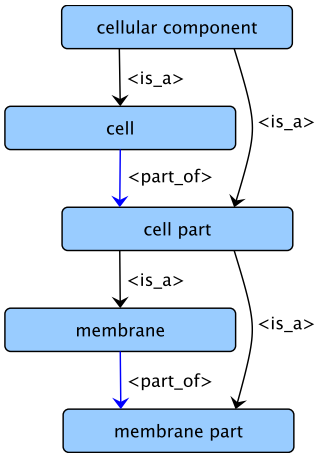


Figure 4.8: Example of GO terms and their relationships. Five terms of the cellular component ontology are displayed. They are hierarchically structured in a directed acylic graph (DAG). Black edges represent `<is_a>`, blue edges `<part_of>` relations. Here, *membrane part* has two parents: first, it is an instance of the term *cell part*. Secondly, it exists as part of *membrane*.

(`http://www.geneontology.org/GO.downloads.shtml`). The ontology itself can be downloaded in OBO format (official text file format for defining and editing ontologies: `http://www.geneontology.org/GO.format.obo-1_2.shtml`) and XML format. Addtionally, the gene association files for all experimental organisms are available at the GO web page.

4.5.2 Implementation

The intention was to implement a user-friendly software to assign read matches derived from a BLASTX comparison against NCBI-nr to GO terms. Generally speaking, MEGANs strategy to bin reads functionally is the following:

1. The BLASTX result file is parsed.

2. A mapping file and the header information of each BLAST hit is used to assign reads to GO terms.

3. The GO terms and their relationships within the ontologies are displayed in an interactive graph view.

4. Tools for deeper analysis (e.g. charts, comparison mode), summarization and export of desired data are provided.

The software module (called GOAnalyzer) and its required resources are completely integrated into MEGAN. No additional data preprocessing or further computations have to be priorly conducted by the user.

Generation of Mapping File

As already mentioned, a BLAST file derived from an BLASTX comparison of read sequences against the NCBI-nr database is used as input for GOAnalyzer. Since GO identifiers are not reported directly in a BLAST result file, a mapping is needed to assign read matches to GO terms.

Typical BLAST reports contain one-line descriptions of the top database matches. These descriptions include different database sequence identifiers (among other information) to uniquely identify a database match. A certain identifier syntax is used to identify the source database, such as GenBank (`gb|accession|locus`), SwissProt (`sp|accession|entry name`) or Protein DataBank (`pdb|entry|chain`). Another important database that will be used in this approach, is the NCBI *Reference Sequence Database* (RefSeq) (`ref|accession|entry name`).

The RefSeq database (Pruitt et al., 2009) is hosted at the NCBI (`http://www.ncbi.nlm.nih.gov/RefSeq`). The current release 36 of RefSeq (July 13, 2009) contains $\approx$ 8.1 million proteins of more than 8600 organisms. In contrast to primary sequence databases like NCBI GenBank, genomic DNA, transcripts and proteins contained in RefSeq are validated and checked for format consistency. Further, RefSeq stores only one example of each natural biological molecule for major organisms ranging from viruses to bacteria to eukaryotes. Sequences are non-redundant and regurlarly curated by NCBI staff and collaborators. Due to these reasons, the RefSeq collection is often employed as a stable reference for genome annotation, gene identification and characterization.

RefSeq accession IDs have the following format: two prefix characters are followed by an underscore character ("_") and by a sequence of digits (e.g. `YP_123456`). The accession prefix represents the type of molecule (Genomic, mRNA, RNA, protein) and the source of the sequence information. The NCBI-nr database contains only RefSeq entries belonging to protein sequences. Table 4.1 outlines a list of protein related RefSeq identifiers that can be found in a BLASTX report.

Due to the aforementioned benefits of the RefSeq database, we decided to use RefSeq accessions found in a BLAST file to map read matches to GO identifiers. A comprehensive mapping file was found at the Protein Information Resource (PIR) web site (`http://pir.georgetown.edu`). The conventional usage of this mapping file is to link UniProtKB accession numbers of the Universal Protein Resource Knowledgebase (`http://www.uniprot.org`) to a variety of other databases such as EntrezGene, NCBI GI number, PDB, PFAM and PIRSF (among others). It contains $\approx$ 3.3 Mio mappings of Ref-

Accession	Note
AP_123456	Alternate protein record. This prefix is used for records that are provided to reflect an alternate assembly or annotation. The AP_ prefix was originally designated for bacterial proteins but this usage was changed.
NP_123456	Primarily full-length precursor products but may include some partial proteins and mature peptide products.
XP_123456	Model proteins provided by a genome annotation process; sequence corresponds to the genomic contig.
YP_123456	No corresponding transcript record provided. Primarily used for bacterial, viral, and mitochondrial records.
ZP_123456	Annotated on collections of whole genome shotgun sequence data for a project. Often via computational methods.

Table 4.1: RefSeq accession numbers for protein products. These RefSeq accessions belonging to protein products are reported in a BLASTX result file. (Descriptions taken from: `http://www.ncbi.nlm.nih.gov/RefSeq/key.html`)

Seq identifiers to sets of GO terms (A single RefSeq might be linked to more than one GO term). Because only two out of 21 entries were of interest for our needs (RefSeq and GO), an individual mapping file was created for GOAnalyzer.

This file, named `ref2go` ($\approx$ 41 MB), has been included into MEGAN's installation package. It is updated regularly.

LCA Approach to Assign Reads to GO Terms

The assignment of read sequences to GO terms follows the general procedures of the lowest common ancestor (LCA) approach of MEGAN (Figure 4.2, p. 61) when classifying reads taxonomically. The main concepts are:

1. The mappings of reads to GO terms follow a many-to-one or $(n : 1)$–relationship. This means that a read is allowed to map to at most one GO term (for each of the three ontologies), but a GO term can be mapped to many reads.

2. In case a read could be assigned to different GO terms, a variant of the LCA algorithm is applied in order to return only one representative GO term. This GO term might be different from the ones that have been indicated by the BLAST matches of this read.

As we will see later, these two concepts significantly facilitate the analysis of the large data sets containing millions of environmental read sequences.

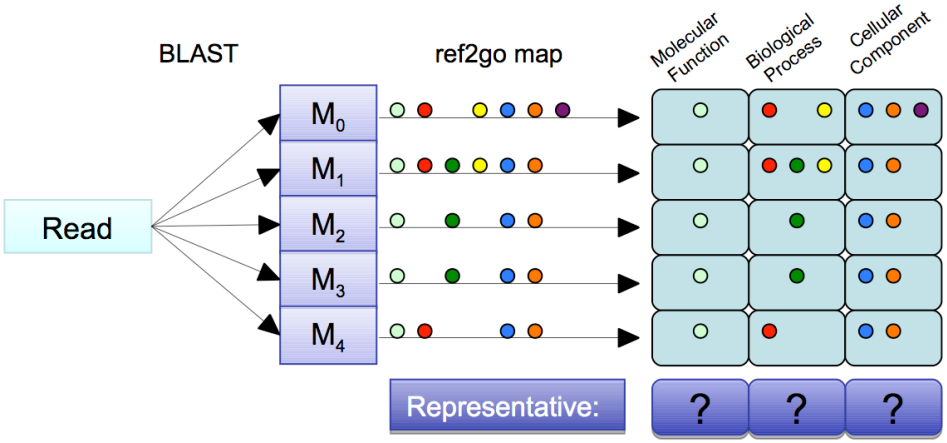


Figure 4.9: Schematic overview illustrating the read to GO term assignment. Each read is assigned to at most one GO term for each ontology. In this simplified example, all 5 BLAST matches ($M_0, ...M_4$) are mapped to different GO terms (colored dots). Each colored dot represents a single GO term (e.g. light green: *protein binding*, red: *response to stress*, yellow: *cell communication*). They are binned into the three ontologies. The LCA algorithm then determines the three GO term representatives (placements) for each read.

At first, the RefSeq identifiers of all BLAST matches that passed MEGAN's *bit-score* filter are mapped to GO terms using the provided `ref2go` mapping file. These GO terms are then binned into three sets representing the three ontologies (biological process, molecular function and cellular component) (see Figure 4.9).

After the binning, the GO terms in each set are sorted in descending order by their number of occurrences. A majority threshold (e.g. 80%) is applied to keep only those GO terms with the highest number of occurrences ("hit" GO term nodes). These terms are then used to build an *induced* DAG for each ontology. See Figure 4.10 for an example of an induced DAG. "Induced" here means that the graph only consists of the given GO term nodes, the root node and any other intermediate nodes being part of paths connecting the GO terms with the root node. Hence, by using the parent-child relationships of GO, the induced GO DAG is built in a bottom-up manner. Starting at the hit GO term nodes (leaves) and following the paths up to the root node, the DAG is generated. Next, inner nodes that have been assigned with reads are disregarded to keep only the GO terms with the most specific biological meaning.

In contrast to tree graphs, DAGs might contain more than one LCA. Due to the fact that only a single GO term is actually needed per read, the most specific LCA has to be selected as ontology representative.

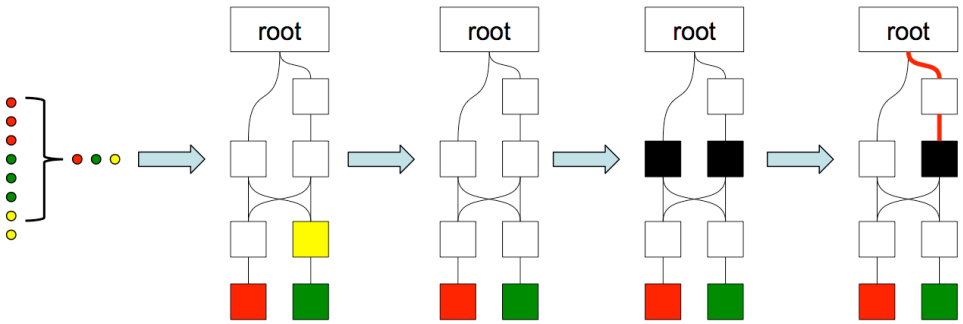


Figure 4.10: LCA approach. This example describes the application of the LCA approach to GO terms contained in the *Biological Process* ontology as shown in Figure 4.9. All three GO terms that passed the majority filter (dark green: *signal transduction*, red: *response to stress*, yellow: *cell communication*) are used for building the *induced* DAG bottom-up. Inner nodes are disregarded keeping only the most specific GO term nodes. In the next step, two LCAs (black nodes) are obtained. If both are assigned with the same specificity score, the LCA with the longest path to the root node is selected as representative (placement) within the *Biological Process* ontology for a single read.

Detection of the Most Specific LCA

The identification of a specific GO term is subject to a three-step filtering approach. First, a specificity score is calculated and the LCA with the highest score is selected. In the case that multiple LCAs have the same specificity, the positions of the LCAs in the DAG evaluated in a second and third step.

The specificity or information content of a GO term is calculated in a preprocessing step resulting in a file named `goid2specificity.map.gz` that maps GO identifiers to a specificity score. The specificity is based on the number of annotated genes for a single GO term and its descendants compared to the total number of annotated genes, as explained in Reference Genome Group of the Gene Ontology (2009) and Alterovitz et al. (2007)). Let $p(V_n)$ be the probability of observing a randomly selected gene to be annotated by term node V_n

$$p(V_n) = \frac{|k(V_n)|}{|\bigcup_{m=1}^{j} k(V_m)|} \tag{4.1}$$

whereas $k(V_n)$ is the gene set annotated to node V_n and j is the total number of nodes in GO. The specificity score $specScore$ for V_n is then

$$specScore(V_n) = -\log_2 p(V_n) \tag{4.2}$$

Equation 4.2 indicates that the informativeness of a GO term decreases as the frequency of annotated genes increases. This means that the nearer a

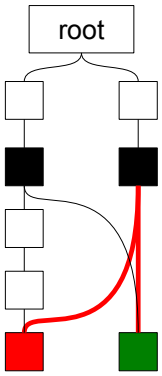


Figure 4.11: Detecting the closest LCA in GO DAGs. This example shows a situation when multiple LCA nodes have the same path distance to the root node. In this scenario, for each of the two LCA nodes (black) the path lengths to all hit GO term nodes (red and green) are summed up (left LCA node: 3+1=4, right LCA node 1+1=2). The right LCA node is the closest LCA because its path length is the lowest.

node is located to the root, the more genes are annotated to this node (and its descendant nodes) and the less informative or specific this node is. This also implies that the (imaginary) "super root" node being the parent node of the root nodes "biological process", "cellular component" and "molecular function" is assigned with $specScore = 0$.

Now, if a single LCA node is found with maximal specificity it is considered as ontology representative for a read. Otherwise the set of LCA nodes with maximal specificity score is the input for the next filtering step that evaluates the location of the nodes in the DAG. The goal is to select the LCA with the longest path to the root node (see Figure 4.10). This strategy assures that GO terms with a rather unspecific biological meaning (located nearer to the root node) are disregarded. However, occasionally, there are multiple LCAs having a maximal path length to the root. Then in a third step, another LCA filtering is applied: for each LCA, the path lengths to all hit GO term nodes is summed up (top-down) (see Figure 4.11). By selecting the LCA with the shortest path length, the closest GO term according to its biological meaning is chosen as representative for a read sequence.

At the end of this binning process, each read sequence is assigned to at most three GO terms. Obviously, situations may occur when a read can not be mapped to GO terms in all three ontologies at all. Reasons for this are for example, the lack of high quality matches in the BLASTX file or missing *ref2go* mappings.

Finally, the read-to-GO assignment is visualized in an interactive DAG view. Figure 4.12 sums up the main steps of the described processing pipeline.

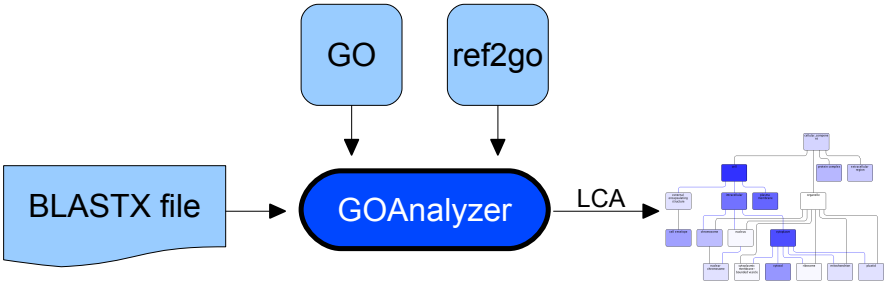


Figure 4.12: GOAnalyzer pipeline. Three inputs are needed for a functional metagenome analysis: the match information of each read from a BLASTX file, the Gene Ontology and a `ref2go` file mapping RefSeq identifier to GO terms. After applying a variant of the LCA algorithm, the found gene products are visualized in a DAG view.

4.5.3 Results

GOAnalyzer has been developed as a module for the MEGAN software. The benefit for the user is that once the BLASTX file is imported, both, taxonomical and functional analyses on the environmental reads can be conducted in one step.

The main window of GOAnalyzer is composed of an overview panel, a GO term listing and the main DAG view (see Figure 4.13). All GO term assignments previously computed for each read, together with their relationships, are visualized as nodes and edges, respectively. Again, the *induced* GO DAG is constructed instead of the whole set of available GO terms. This helps to substantially reduce the number of displayed GO terms. (The yFiles graph library (`http://www.yworks.com`) has been used for graph drawing and graph layout purposes.)

Different edge colorings represent the different types of GO relationships (described in Section 4.5.1) between the terms. By default, a gradient color scheme for the nodes indicates the number of assigned read sequences for each GO term. Nodes without any assigned reads are only inserted into the DAG (as inner nodes) if they are associated with a path from a hit GO term node to the root. Several different node drawers are available (see Figure 4.14).

To facilitate the inspection of the set of GO terms, the user is able to zoom into regions of interest within the DAG. (see Figure 4.15). In addition

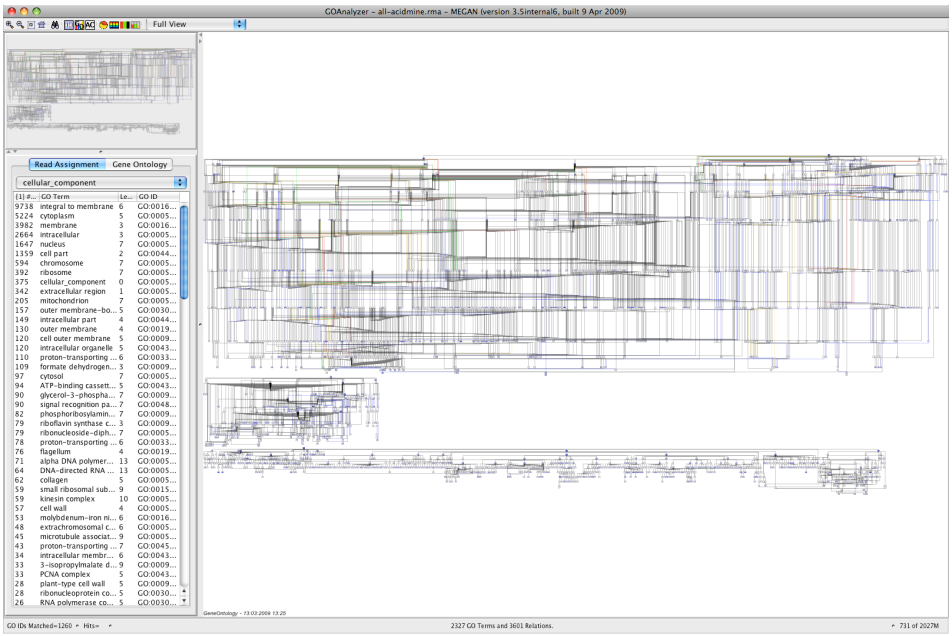


Figure 4.13: GOAnalyzer screenshot. The graphical user interface of GOAnalyzer is separated into three parts: an overview panel (top-left corner), a GO term listing (left) and the scalable main view with the interactive DAG visualization (right). In this example, the three ontologies of GO are displayed as separate subgraphs. Shown is a data set derived from Tyson et al. (2004) (acid mine drainage metagenome) containing 321.426 read sequences which are assigned to 1260 GO terms resulting in 2327 nodes and 3601 edges (relations).

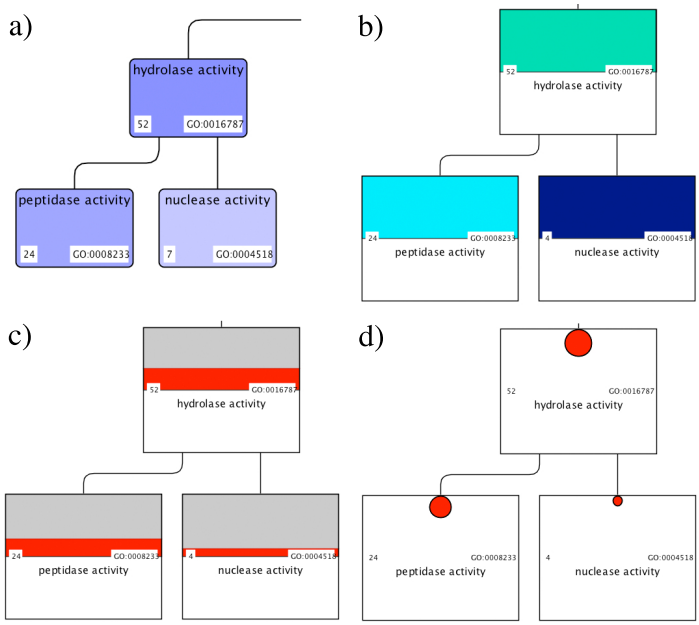


Figure 4.14: Node drawers. Several kinds of node drawers are available. Each drawer indicates the amount of assigned reads differently: a) By default, single data sets are drawn as rounded rectangle using a color gradient (white → no reads assigned to dark blue →many reads assigned). A heat map node drawer is shown in b). Its color gradient comprises all shades between black and red. Other node drawers are c) meters view and d) pie chart view.

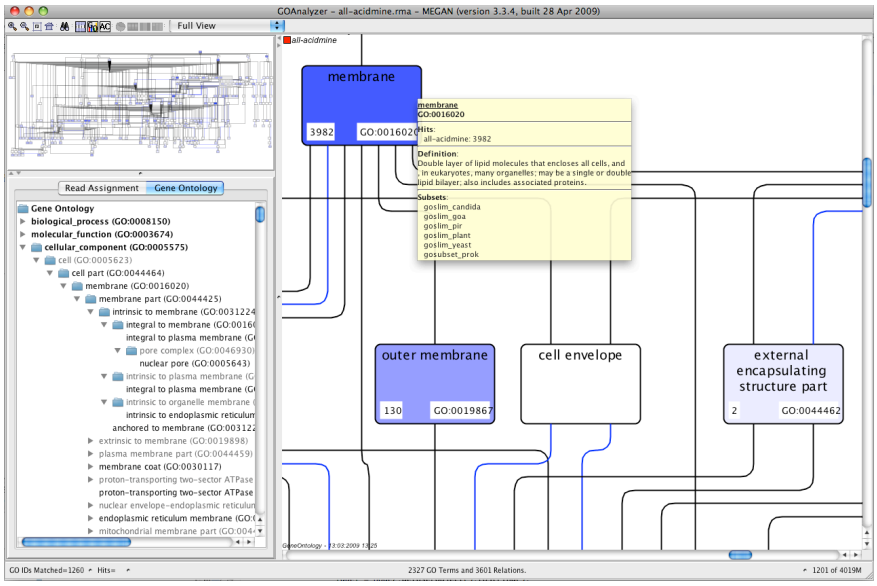


Figure 4.15: Zoom view. By using the mouse wheel, the user is able to zoom into the DAG. Such, the assigned GO term can be inspected. Hovering the mouse over a node brings up a tooltip box with information about the GO term.

to the visualized DAG, the graphical user interface provides different views on the extracted GO term information. In adaption to the official Amigo web interface of GO (`http://amigo.geneontology.org`), GOAnalyzer provides a text-based hierarchy that displays the GO terms in a tree-like manner. Its entries can be collapsed and expanded to browse the content of the GO tree (Figure 4.15).

Alternatively, a tabular listing displays the set of GO terms being exclusively assigned to at least one read. By clicking on an entry in the table or the tree, the DAG view directly zooms and centers the GO term in the DAG view. To allow for a manual examination of the BLAST hits of the assigned read, GOAnalyzer provides an inspector window. It displays all assigned reads together with their matches and BLAST hits for selected GO terms. Furthermore, the user is able to export all read sequences of selected GO terms into a multiFASTA file.

By default, the graph is drawn in antialiased mode giving the visualization a smoother appearance (e.g. for image export). In case of large data sets (e.g. more than 5000 displayed nodes), the user may turn off antialiasing to speed up the processing of the data visualization. Additionally, the detail level can be further decreased by choosing (*Edit* → *Preferences* →*Optimize View for Large Data Sets*).

GOAnalyzer has been already applied for a functional analysis of samples of the woolly mammoth hair and bone. (Zhao *et al.*, 2009, submitted). With the help of the software, several significant GO terms have been identified according to their specificity, the amount of assigned reads and their level in the GO DAG.

High-Level View using GO Slims

By default, GOAnalyzer assigns read sequences to all GO terms available from the official Gene Ontology. Currently, there are more than 28,000 terms contained in GO. Obviously, the number of terms will grow in the future as new definitions of gene products and their relationships will be added.

On the one hand, the wealth of GO terms allows for a distinct and precise consideration of the involved gene products in a metagenome sample, i.e. reads can be assigned to rather specific terms near the leaves of the DAG. On the other hand, reads are likely distributed over the whole GO DAG and, thus, might hamper to detect the abundance of functional groups of interest (such as *response to stimulus*, *protein binding* or *cell wall*). Without a convenient summary feature, the user may have difficulties to obtain the "big picture", i.e. the overview of the general, functional profile of the whole data set.

Hence, in addition to the default view based on the complete set of GO terms, GOAnalyzer provides the user with a high-level view on the data. The idea is to categorize the gene products on a basis of a relatively small

set of high-level GO terms to reduce the complexity of the ontology. The Gene Ontology Consortium offers several preconfigured "GO slims" on their web site which contain a subset of GO terms and their relations (`http://www.geneontology.org/GO.slims.shtml`). The GO slims are regularly downloaded and integrated into MEGAN's installation package.

Currently, four slims with different focuses are officially maintained as individual files at the GO website: Generic GO slim, GOA and whole proteome analysis, Plant GO slim and Yeast GO slim. Additionally, a prokaryotic subset of all GO terms is provided as a category in the GO ontology file. This subset contains only terms applicable to prokaryotes (e.g., terms like *nucleus* and *mitochondrion* are excluded). Table 4.2 gives an overview of the amount of contained terms of each GO slim.

Topics/Usage	**Terms**
GOA and whole proteome analysis	64
Yeast GO slim	89
Plant GO slim	105
Generic GO slim	131
Prokaryotic subset	8196

Table 4.2: GO slims and subsets maintained by the GO Consortium. All slims including the prokaryotic subset represent a minor subset of the terms contained in the official GO (28089 terms, as of August 20, 2009).

Analyzing the functional content of a metagenome, the user may switch between the mentioned GO slims/subset views and the complete, full view. In case a GO slim is selected, all computed GO terms of the full view are mapped to high-level GO terms of the GO slim. This is done by employing the parent-child relationship of the GO terms: Starting at a hit node v with assigned reads in the full view, the ontology DAG is traversed bottom-up until a node u is approached that is part of the GO slim/subset and ancestor of v. Then all read matches are assigned to this high-level term node u. As a result, the large DAG of the full view is replaced by a GO slim DAG showing only a minor fraction of all available terms. Given this data summary feature, the user is able to conduct a broad functional classification of environmental read sequences without having to consider the detail of the specific fine grained terms (see Figure 4.16 for an example).

An interesting idea would be to create special GO slims for metagenomic projects. For example, by selecting GO terms of interest according to different type of habitats (e.g. marine or terrestrial habitats), researcher could focus their search for gene products on characteristic GO terms.

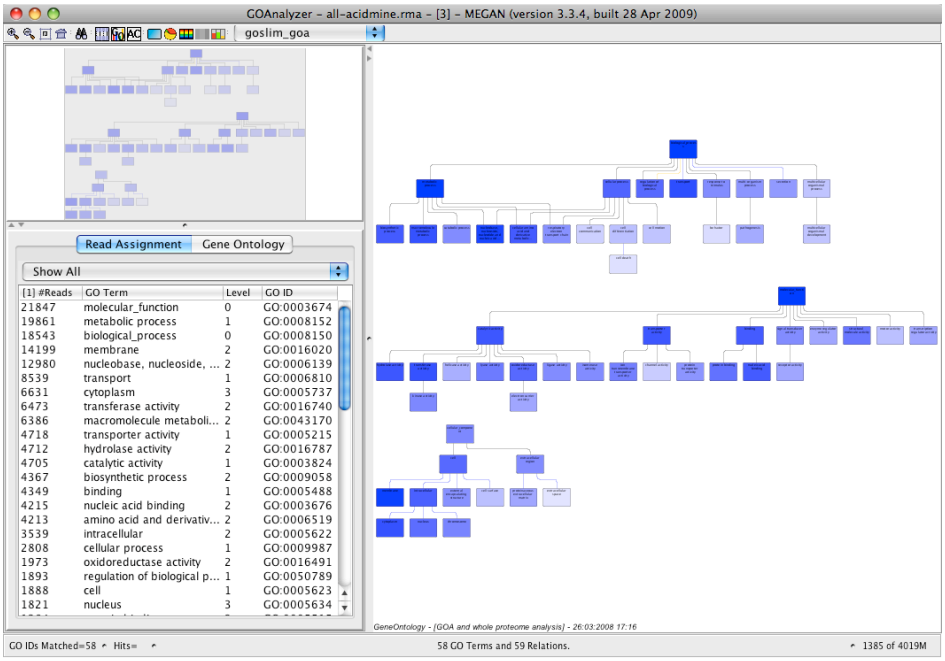


Figure 4.16: GOA GO slim. In this example, the same data set as in Figure 4.13 is loaded. Compared to the former 2327 nodes and 3601 edges of the full view, the GOA slim reduces the number of nodes and edges to 58 and 59, respectively.

Comparison Mode

Besides data analyses dealing with only one metagenome sample, studies comparing many different data sets simultaneously are of great interest. In the case that different samples are derived from the same habitat, one may, for instance, analyze differences in the species composition when extracting samples from various locations or depths of the habitat (e.g., in soil or marine environment). In contrast, analyses using samples from different environments are more focused on a general characterization of the habitats.

In Rusch et al. (2007), for example, researchers compared different marine samples at different locations by looking at the composition of marine planktonic microbiota which can be found in surface water samples. In contrast, another study (Dinsdale et al., 2008) aimed at a comprehensive comparison of metabolic profiles of differing environments, e.g., subterranean (mine), hypersaline ponds from solar salterns, marine, freshwater, coral-associated, terrestrial, animal-associated and many others.

Since more and more single metagenomes are described and published, the amount of studies comparing metagenomes are about to increase. To support the functional analysis of comparative metagenome studies, GOAnalyzer is able to process and visualize the distribution of read assignments derived from multiple data sets. According to MEGAN's taxonomical com-

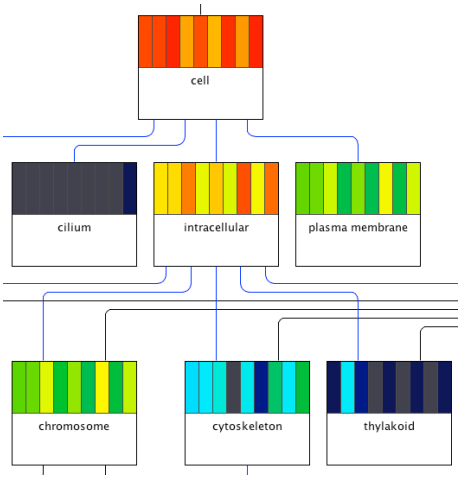


Figure 4.17: Heatmap node drawer

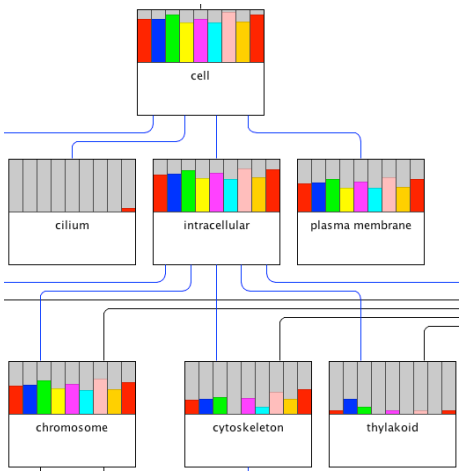
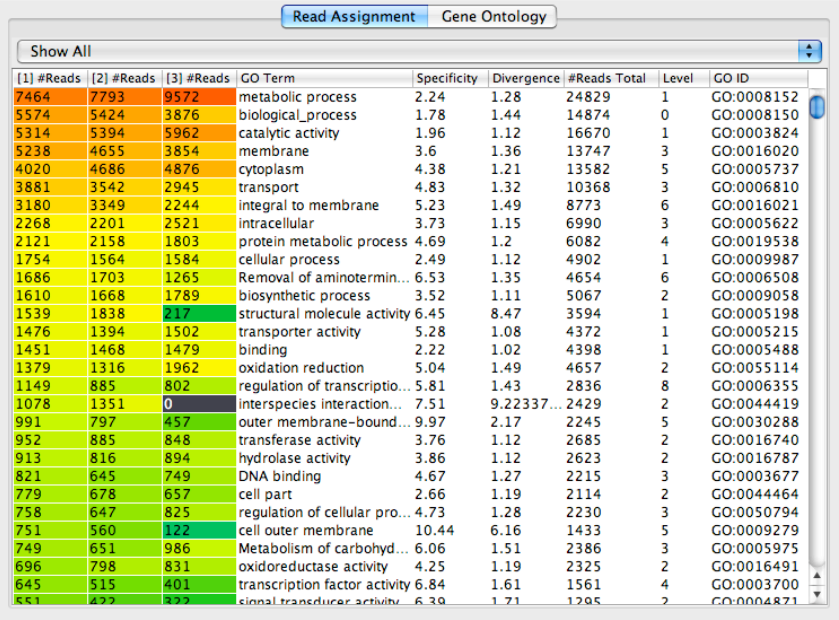


Figure 4.18: Meters node drawer

parison view, the GO term nodes can be drawn using different node drawers (as depicted in Figures 4.14, 4.17, and 4.18).

Further features enable the user to quickly find similarities or differences between data sets at a single glance. For instance, the tabular listing of the read assignment can be switched to a "heat map" mode. Such, the background colors of the cells in the table are set according to the number of reads. The color values are logarithmically scaled (black: no hits, red: many hits). Especially, when comparing a large number of data sets (> 4), the color coded visualization of the table helps to quickly determine differences between the data sets (see Figure 4.19). The table also includes columns listing the specificity score (as explained in Section 4.5.2) and, in case of multiple data sets, the divergence and sum of read assignments for each GO term. The divergence of a GO term V_n is represented by $div(V_n) = \frac{max(readAssignments(V_n))}{min(readAssignments(V_n))}$. It indicates significant differences between the maximum and minimum amount of assigned reads for all data sets of a single GO term.

Usually, the amount of sequenced reads derived from several metagenomic samples may vary to a certain extent. Even if the lab work was repeatedly conducted by the same person, or if the samples originated from the same habitat, this bias has to be considered when comparing multiple data sets. If the amount of reads differs significantly between data sets, an intermediate normalization step is usually preferable. If the user already has decided to normalize the number of reads in the taxonomical analysis in MEGAN, the data used by GOAnalyzer will be normalized as well: for each data set d_i and for all $i = 0, \ldots, k$ a factor $f \in \mathbf{R}$ is computed with $f_{d_i} = \frac{c}{\text{amount of reads in } d_i}$ whereas c is a constant number, e.g. 100,000. This factor is then used to scale all read amounts for each GO term. Such, con-



[1] #Reads	[2] #Reads	[3] #Reads	GO Term	Specificity	Divergence	#Reads Total	Level	GO ID
7464	7793	9572	metabolic process	2.24	1.28	24829	1	GO:0008152
5574	5424	3876	biological_process	1.78	1.44	14874	0	GO:0008150
5314	5394	5962	catalytic activity	1.96	1.12	16670	1	GO:0003824
5238	4655	3854	membrane	3.6	1.36	13747	3	GO:0016020
4020	4686	4876	cytoplasm	4.38	1.21	13582	5	GO:0005737
3881	3542	2945	transport	4.83	1.32	10368	3	GO:0006810
3180	3349	2244	integral to membrane	5.23	1.49	8773	6	GO:0016021
2268	2201	2521	intracellular	3.73	1.15	6990	3	GO:0005622
2121	2158	1803	protein metabolic process	4.69	1.2	6082	4	GO:0019538
1754	1564	1584	cellular process	2.49	1.12	4902	1	GO:0009987
1686	1703	1265	Removal of aminotermin...	6.53	1.35	4654	6	GO:0006508
1610	1668	1789	biosynthetic process	3.52	1.11	5067	2	GO:0009058
1539	1838	217	structural molecule activity	6.45	8.47	3594	1	GO:0005198
1476	1394	1502	transporter activity	5.28	1.08	4372	1	GO:0005215
1451	1468	1479	binding	2.22	1.02	4398	1	GO:0005488
1379	1316	1962	oxidation reduction	5.04	1.49	4657	2	GO:0055114
1149	885	802	regulation of transcriptio...	5.81	1.43	2836	8	GO:0006355
1078	1351	0	interspecies interaction...	7.51	9.22337...	2429	2	GO:0044419
991	797	457	outer membrane-bound...	9.97	2.17	2245	5	GO:0030288
952	885	848	transferase activity	3.76	1.12	2685	2	GO:0016740
913	816	894	hydrolase activity	3.86	1.12	2623	2	GO:0016787
821	645	749	DNA binding	4.67	1.27	2215	3	GO:0003677
779	678	657	cell part	2.66	1.19	2114	2	GO:0044464
758	647	825	regulation of cellular pro...	4.73	1.28	2230	3	GO:0050794
751	560	122	cell outer membrane	10.44	6.16	1433	5	GO:0009279
749	651	986	Metabolism of carbohyd...	6.06	1.51	2386	3	GO:0005975
696	798	831	oxidoreductase activity	4.25	1.19	2325	2	GO:0016491
645	515	401	transcription factor activity	6.84	1.61	1561	4	GO:0003700
551	423	332	signal transducer activity	6.39	1.71	1295	2	GO:0004871

Figure 4.19: Colored read assignment table. Screenshot showing the tabular read assingment. Cells are colored according to the amount of reads assigned to a GO term. Further information is listed for each term (specificity, divergence, DAG Level, ...) that support the user when searching for significant differences between multiple data sets.

tents of diverse data sets in terms of read amount are brought to the same scale.

Chart Tool

In addition to the DAG view or the tabular and tree listings of the read assignments, GOAnalyzer uses MEGAN's chart functionality to provide bar or pie charts for the GO analysis (Figure 4.20). Therefore, the user simply selects a set of designated GO terms in the graph view and clicks the GO chart button.

4.5.4 Discussion

In this subchapter, a new approach for the functional analysis of metagenome data has been described. The developed tool has been implemented as part of the MEGAN software. The Gene Ontology was used to characterize the functional gene content of a metagenomic sample. The classification of read sequences is based on a mapping of RefSeq identifiers found in a BLASTX result file (e.g. after a blasting against NCBI-*nr*) to GO terms. After applying a variant of MEGAN's LCA approach, each read is assigned to a single GO term in each of the three ontologies. A graphical user interface displays

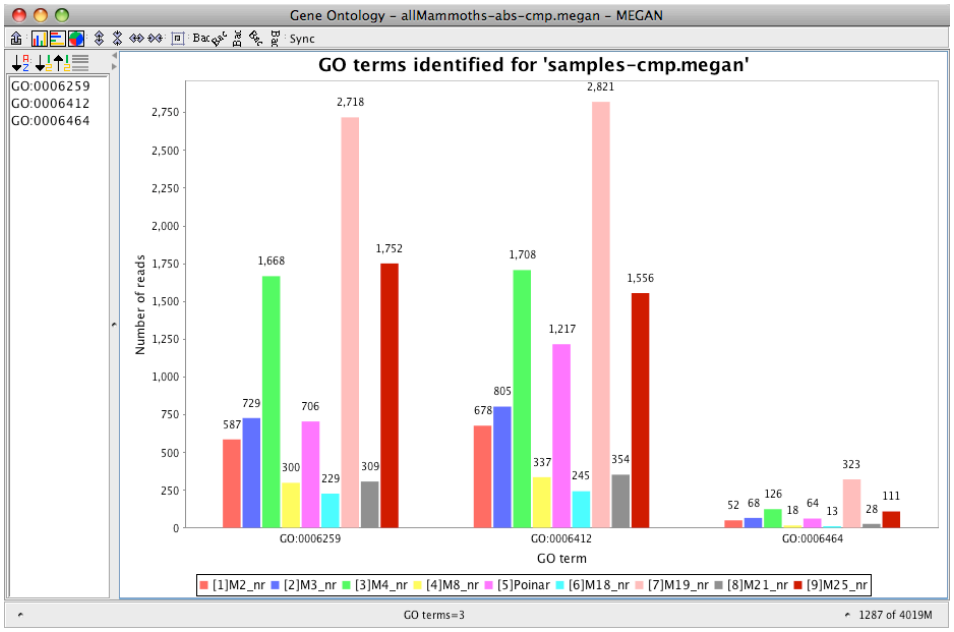


Figure 4.20: GO bar chart. Three GO terms have been selected for the bar chart view. Different colors represent different data sets.

the data in a DAG view and in a tree or tabular listing.

The main algorithmic concept of GOAnalyzer is based on the LCA approach. Because a BLAST run usually produces tens or even hundreds of BLAST matches for a single read that all, again, may be mapped to different GO terms, a single, representative GO term for each read and ontology is selected by the LCA algorithm.

What are the drawbacks and benefits of the LCA strategy?

Drawbacks First of all, a disadvantage may be the expected loss of accuracy. Due to the parent-child relationships of GO, the biological meaning of a LCA term, being a parent node of a set of GO terms, is always less specific. In the worst case, the LCA is the root node of the ontology. Specific gene products located near the leaves of the ontology are not considered any longer and replaced by a high-level term (*high-level terms* are defined to be GO terms located near to the root node). In this way, gene products of interest might be missed in the result. As a consequence, the composition of the GO term associated gene products might be rather general (e.g., *protein complex* (GO:0043234) instead of *proton-transporting V-type ATPase, V1 domain* (GO:0033180)). Such situations obviously complicate a distinctive characterization of a data set.

Another implication of the LCA algorithm is that reads having a large

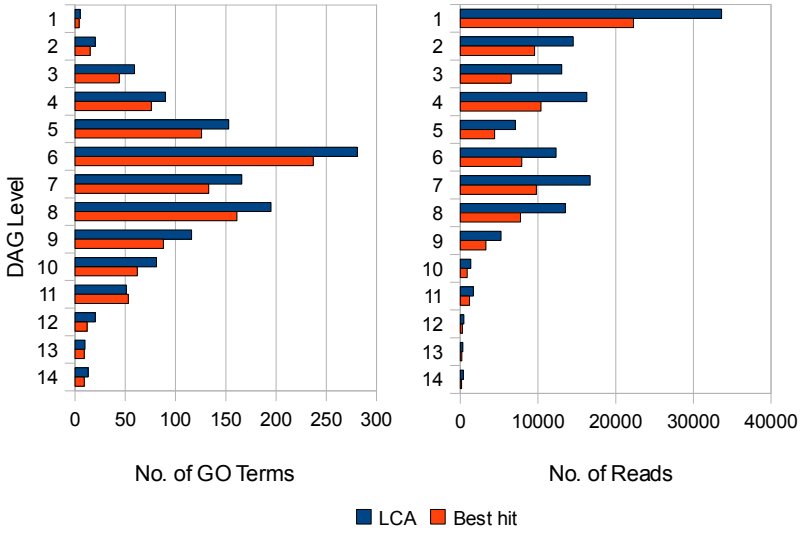


Figure 4.21: DAG level test: acid mine metagenome. The number of GO terms (left) and the number of assigned reads to GO terms (right) at different path levels are shown for the LCA approach and the best BLAST hit, respectively.

number of matches in the BLAST file (with a certain bit-score, prefiltered by MEGAN), are more likely to be assigned to high-level GO terms. Even if most matches are mapped to related gene products, one outlier (match mapped to functionally distant gene product) might generalize the result.

Facing these problems, we first tried to skip the complete LCA step, hoping to observe a significant impact on the specificity of the extracted GO terms, i.e. to find GO terms located nearer to leaf nodes. This test was performed by selecting only the best BLAST hit for a read instead of considering all matches that passed the quality filter of MEGAN. We used a data set of the acid mine drainage metagenome containing ≈321,000 reads (Tyson et al., 2004). Each GO term in the GO DAG was assigned with a *level index* (tree depth), such that the root node is assigned with level index 1. If a GO term could be assigned with various levels (i.e. the node is accessible using multiple paths), the maximal (deepest) level index is used.

The resulting diagrams are shown in Figure 4.21. First, the number of extracted GO terms and second, the amount of assigned reads per DAG level have been counted, respectively. In both cases, we expected to see a shift towards the lower level indices (towards the leaf nodes at level index 14) when selecting the best BLAST hit. But, remarkably, changes are only insignificant. The distribution remain roughly the same. Note that the best hit strategy yielded less GO terms and less reads at all. This is due to the fact, that the best hit is not always associated with a RefSeq identifier and,

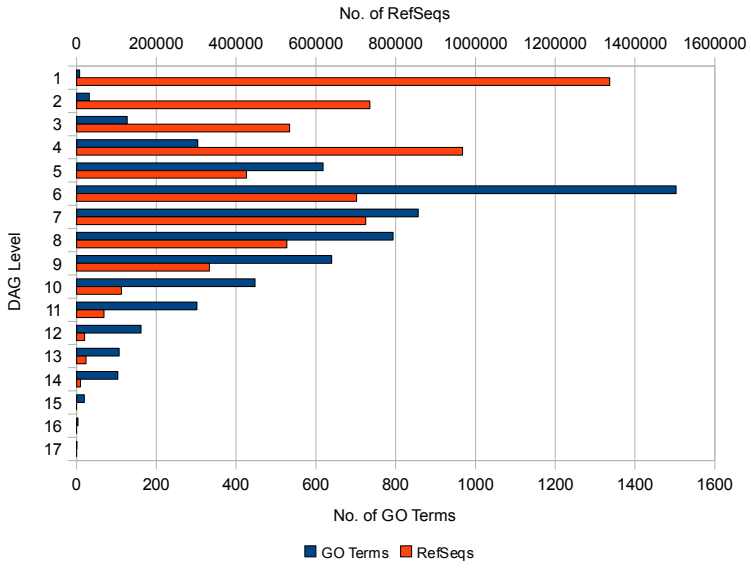


Figure 4.22: DAG level test: mapping file. Distribution of GO terms and RefSeq identifiers in correspondence to the DAG level of the maping file is shown.

therefore, could not be mapped to a GO term.

The left diagram indicates that most GO terms can be found at mid-levels 5 to 8. Only few GO terms at levels 11 to 13 have been assigned with reads. Interestingly, 24.6% and 26.3% of the reads for the LCA and best-hit approach, respectively, are located at the root node (level 1) (Figure 4.21, right diagram). This indicates that many reads could not be assigned to significant GO terms. However, it could be confirmed that the LCA approach does not necessarily produce worse results compared to the best-hit strategy.

To further evaluate the efficiency of the LCA approach, we looked more closely into the mapping file to find possible evidence for biased results. Therefore, we used the same approach as described earlier: All mappings of RefSeq identifiers to GO terms are used to analyze their DAG level distribution. The results are again visualized as a diagram shown in Figure 4.22. It can be seen that the distribution of the GO terms in the mapping file related to the DAG level (shown in in Figure 4.22, blue bars) is similar to the GO term distribution in the left diagram of Figure 4.21. This indicates that the mapping file itself actually determines the amount of GO terms per DAG level.

The red bars in Figure 4.22 represent the distribution of the RefSeq identifier which have been mapped to GO terms after applying the LCA

algorithm (note that a RefSeq identifier usually maps to a set of GO terms). Also here, the RefSeq distribution reveals that 20.5% of all RefSeqs are located directly at the root node (level 1). This proportion is similar to the amount of reads mapped to the same level in the acid mine data set (Figure 4.21, right diagram).

Both analyzes have shown that, first, omitting the LCA approach is not the better option. Taking only the best BLAST hit might miss a lot of GO terms and, at the same time, does not necessarily provide a more significant amount of specific GO terms. Secondly, results are strongly biased by the contents of the mapping file. The composition of the mapping file apparently influences the selection of GO terms, independent of choosing the LCA or best-hit approach. Since no other mapping file is available at this time, one has to keep this bias in mind when interpreting the results.

Another side-effect of our homology-based approach is that gene sequences in the sample showing no similar base or amino acid sequence will not likely be detected. For this purpose, *ab initio* gene prediction tools and pipelines like Glimmer (Delcher et al., 1999), fgenesb (`http://www.softberry.com`) or MetaGeneAnnotator (Noguchi et al., 2008) are more appropriate. And, as mentioned before, homology-based methods suffer from the database bias, i.e. the ever-changing and limited composition of biological databases (see Chapter 2.4.4, p. 42).

Benefits One of the advantages of the LCA approach is the reduction of complexity in terms of data manageability and memory requirements. By following the principle "each read is assigned to a single GO term" the total amount of GO terms is significantly decreased. This complexity reduction and, at the same time, the substitution of a set of specific GO terms with a single one, is justifiable, when looking at the composition of typical metagenomic data sets: Environmental sequencing projects usually produce between ≈50,000 and up to tens of millions of read sequences depending on the employed sequencing technology and read length. Structuring and classifying this huge amount of data is a challenging task.

Here is a numerical example: assuming a generic set of sequences containing 1 million reads. Running BLASTX against the NCBI-nr database typically produces about 100 matches per read. After the bit-score filtering and the mapping of these matches to GO terms, 50 matches per read may remain. Assuming, each of these 50 matches may have a mapping to at least 15 GO terms. As a result, 750,000,000 GO terms would be generated that have to be stored and visualized in a convenient manner. This conservative estimate reveals the complexity that one encounters when processing metagenomic data.

As GOAnalyzer is a GUI based stand-alone software which is supposed to be run on standard laptops, its memory usage is of particular impor-

tance. By restricting the number of GO terms per read to at most three, only three integer values (the GO identifier) have to be stored and hold in memory. Consequently, compared to the result of the calculation above, only 3 million GO terms (4%) need to be considered. The complexity reduction considerably facilitates the overview and the analysis of the functional content of a metagenome. As reads are assigned to single GO terms ($(n:1)$-relationship), the extraction and inspection of reads per term can easily be accomplished. For instance, the user may click on a term node in the DAG view to export the read sequences into a multi-FASTA file. Multiple assignments of the same read to different GO terms would weaken the strict classification and binning of reads.

Overall, it has to be pointed out that "loss of accuracy" as it has been priorly described, does not lead to a "loss of correctness". The LCA of a set of GO terms is never "wrong" in terms of the biological meaning. It is just a higher-ranked GO term summarizing its child terms with a more general description. By selecting a parent node in the DAG, false-positive read assignments can be avoided and, at the same time, biological correctness is still assured.

The usage of GO as classification structure for annotating reads, is still not widely common in published metagenomic studies. Other classifications used instead are, for example, COG, TIGRFAM, Pfam, or SEED (Tatusov et al., 1997; Haft et al., 2003; Finn et al., 2008; Overbeek et al., 2005). However, this diversity of methods obviously complicates the feasibility of comparative studies, since content, structure and focus differs between these classifications. GO takes an exceptional position among these classifying methods and databases. As explained earlier, the GO consortium tries to determine consistent descriptions of gene products in different databases. So, there is an obvious tendency to link different namespaces (databases) covering similar biological content. GO is the first step for a "unification of biology"(referring to Ashburner et al. (2000)). For this purpose, there already exist a lot of mappings to GO (`http://www.geneontology.org/GO.indices.shtml`) and more will certainly be added.

Concluding the enumeration of pluses and minuses of our strategy, the aim of this project was to provide a user-friendly, intuitive and biologically sound software tool for the functional analysis of metagenomic data. The main challenge was to create a software tool capable of dealing with the diversity and richness of typical metagenomic data sets. At same time, the software should be used on a standard laptop and should allow for convenient, visual analysis of data.

To balance these two points, MEGAN uses the LCA approach in both cases: in the process of the taxonomical and the functional binning of environmental read sequences. This strategy enables the user to get a first insight into the taxonomical and functional composition of a metagenomic sample in one step. Once the read sequences have been blasted on a com-

puter cluster, MEGAN and and its extension GOAnalyzer provide both, a bird's eye view and a close-up view down to the read sequences and even BLAST matches. These features make MEGAN the preferable tool for a comprehensive overview analysis during the first steps of a metagenomic analysis.

Chapter 5

MetaSim: A Sequencing Simulator for Genomics and Metagenomics

5.1 Introduction

The recent developments of next-generation DNA sequencing technologies opened the flood gates to an extensive amount of sequence data. Prior to any biological interpretation of the studied DNA and its features, the set of short sequencing reads has to be processed, aligned, assembled or classified depending on the type of biological analysis.

For example, for a typical whole genome sequencing project of a single organism, the reads have to be filtered by quality and then assembled to obtain the final genome sequence (see Section 2.3.1, p. 26). Once larger assembled fragments (contigs) are obtained, analyses like gene prediction or motif finding become reasonable. In case of re-sequencing projects (see Section 2.3.2, p. 31), the reads have to be mapped to a closely related reference genome by efficient short read mapping software (Trapnell and Salzberg, 2009). A different scenario of processing reads is the analysis of environmental DNA obtained from a community sequencing project (see Section 2.4, p. 33). In contrast to single genome studies, the new discipline of metagenomics focuses on the analysis of the microbial diversity found in various habitats, like e.g. ocean (Rusch et al., 2007), soil (Tringe et al., 2005), mines (Tyson et al., 2004; Edwards et al., 2006) or the human microbiome (Turnbaugh et al., 2007). Due to the high complexity of ecologic systems, the assembly of reads is very challenging, i.e. the assembly of reads into contigs belonging to only one species fails or is misleading (see Section 2.4.3, p. 37). To avoid such complications, the initial steps in metagenomic studies often comprise the characterization and classification of reads (binning) into separated sets. On the one hand, reads are classified taxonomically to get an overview about

the contained organisms (taxonomical analysis). On the other hand, the functional spectrum of the sample (independent of the species origin) is of main interest. Therefore, genes are predicted by using homology or *ab-initio* approaches (depending on the read lengths) (see Section 2.4.3, p. 40).

Further, the fast and cost-effective generation of sequencing data, enables researches to perform series of measurements to compare, for instance, the taxonomical composition of samples derived from the same location within several time points under varying environmental conditions (Gilbert et al., 2008). Such comparative studies again depend on powerful, statistical techniques and analysis tools which are able to deal with the highly variable data (Mavromatis et al., 2007; Mitra et al., 2009)

This overview of different types of read processing presents only an incomplete list of all common analysis strategies. But it should give an idea of how the progress of next-generation sequencing technologies (see Section 2.2.2) is spurring the field of bioinformatic software development. The amount of developed assembler software is a good measure for this: Between the year 2000 and July 2009 more than 18 different genome assemblers have been introduced (Scheibye-Alsing et al., 2009) and more are likely to be published or at least upgraded in the near future. Regarding metagenomic studies, the data size generated (measured in base pairs) occasionally exceeds common single genome sequencing projects. However, it is striking that the number of specialized software and algorithms for processing environmental sequences is surprisingly low. As a consequence, many studies use the classic methods, software or web services that originally were not intended for metagenomic data.

To sum it up, there is a great demand for improved and specialized software solutions in both research fields of genomics and metagenomics that keep up with the rapid developments and improvements of NGS technologies. The vast collection of current assembly and mapping software for genomics and the upcoming tools and methods for metagenomics need to be compared and benchmarked to evaluate their performance and applicability. Standardized test scenarios using simulated and verifiable data are useful for developers to analyze their programs and for users to select the software that optimally fits their needs. These considerations motivated us to develop MetaSim (Richter et al., 2008), a DNA sequencing simulator software for the generation of synthetic reads based on given genome sequences. A prior study (Mavromatis et al., 2007) provided three data sets with varying complexity by selecting original sequence reads from 113 isolated genomes. The authors anticipated that the community uses these precomputed data sets as standard test cases for software testing.

In contrast to this "static" approach, our software MetaSim allows researchers to create their own test data by choosing from a set of source genomes and error models derived from several sequencing technologies (Sanger (Sanger et al., 1977), Roche's 454 (Margulies et al., 2005) and Illu-

mina (Bentley, 2006)). These error models (or error probabilities per base position) are used to modify the original sequence at certain base positions to reflect real sequencing error patterns of each technology. Further, MetaSim offers the possibility to load individual error models based on empirical data.

MetaSim takes as input a set of known DNA sequences and an abundance profile. The profile determines the source DNA sequences and their relative abundances for the simulation read sequneces. The abundance values can be used to reflect the variable species composition of a metagenome. This general approach allows users to use MetaSim flexibly as either read simulator for single genomes or for metagenomes.

Former simulation software tools for Sanger reads were `celsim` and `GenFrac` reported in (Myers, 1999b; Engle and Burks, 1993; 1994). These tools could be used for the simulation of Sanger reads. Other studies (Chatterji et al., 2008; Krause et al., 2008b) used the functionality of *ReadSim* (*unpublished*), a pre-version of MetaSim to produce sequencing reads of different lengths and error characteristics. Some concepts of ReadSim have been adopted and further improved in MetaSim.

This chapter describes the main implementation details of MetaSim and summarizes results of a simulation study for benchmarking the MEGAN software (Huson et al., 2007).

5.2 Implementation

The main steps of MetaSim's simulation processing pipeline are the selection of source genome sequences, the configuration of an abundance profile, the sampling of sequence fragments and the subsequent generation of synthetic reads according to a chosen error model. MetaSim includes an internal database (based on `http://hsqldb.org`) to allow for a convenient selection of source genome sequences for the simulation. This database locally stores and maintains all imported genome sequences and can be explored within the program.

5.2.1 Generation of Species Profiles

As already mentioned, the abundance profile contains names or identifiers (e.g. NCBI's gi number) of all sequences selected for simulation. Additionally, the abundance value determines the relative amount of genome sequences simulated for a metagenome. The text-based profile file (`.mprf`) has a simple structure as shown in this example (The `#` symbol marks comment lines and `%` represents the wildcard character.):

```
### MetaSim taxon profile
# 100 Methanoculleus marisnigri JR1
# 50 Alcanivorax borkumensis SK1
```

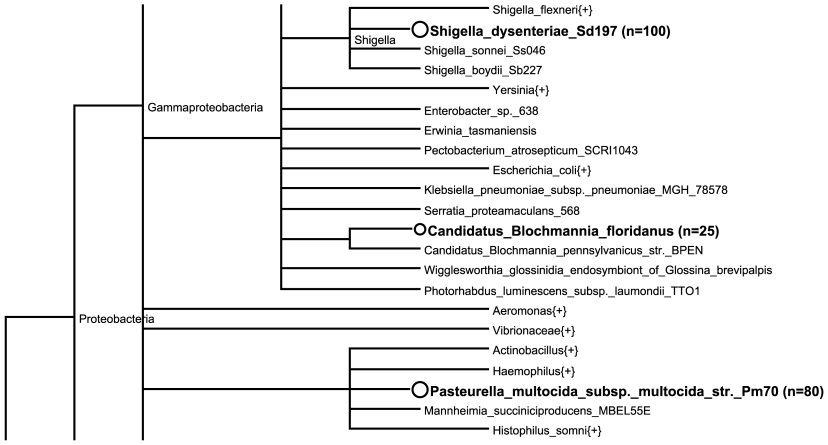


Figure 5.1: TaxEditor. This visual component enables the user to assign abundance values via right-clicking on nodes in the taxonomic tree. Different node sizes correspond to the assigned abundance values. Figure taken from Richter et al. (2008).

```
###
100 name "Methanoculleus marisnigri JR1"
50 name "Alcanivorax bork%"
```

Here, 100 genome copies of *Methanoculleus marisnigri JR1* and 50 genome copies of *Alcanivorax borkumensis SK1* are part of this (meta)genome dataset. This means that the genome sequence of *M. marisnigri JR1* occurs twice as much in the metagenome as the sequence of *A. borkumensis SK1*.

Alternatively, MetaSim provides a visual component that allows to set the abundance values for genome sequences directly in a taxonomic tree. Therefore, an "induced" tree viewer (TaxEditor) of the NCBI taxonomy (Wheeler et al., 2008b) is integrated that displays the genomes in the database as nodes in a rooted tree according to their taxonomical relationships (Figure 5.1). An interesting feature is the possibility to set abundance values not only to species (leaf) nodes (single organisms in the database) but also to inner nodes in the taxonomy (e.g. at genus level). Such, the abundance value of an inner node is split and applied to its descendant species which are available from the database.

5.2.2 Population Sampler

Typical environmental samples contain a vast variety of microbial species. Most of these organisms (or bacterial strains) are usually still unknown because they could not be cultured and isolated in the laboratory before. As a consequence, one can hardly estimate their occurrence and, in general, the genetic diversity of a sample. When it comes to simulating metagenomes, one has to keep that in mind. Thus, to mimic the complexity of real world

data sets, MetaSim provides a population sampler that generates evolved (mutated) offsprings of single source genome sequences. The calculation is based on a mathematical model of DNA evolution and a given evolutionary tree that determines how the offsprings descend from the source genome. By default, we use the Yule-Harding model to generate phylogenetic trees (Yule, 1925; Harding, 1971), but the user may load individual trees as well. For the model of DNA evolution, the widely known Jukes-Cantor model (Jukes and Cantor, 1969) has been implemented. It defines the probability of a change for each base pair, with an adjustable transition rate α (0.001 by default) and time t based on the edge weights of the tree. After applying the population sampler to a genome sequence, the desired number of evolved genomes are added to the internal database.

5.2.3 Read Sampling

MetaSim uses different statistical models to simulate the frequency of simulated reads, the distribution of the read lengths and the probability of occurring mate-pairs.

First, larger fragments called *clones* are extracted from the set of genomes with normally or uniformly distributed lengths. These clones are the basis for either the read or mate-pair sampling. If only a single genome sequence is included in a profile file, the clones are sampled randomly from this genome sequence (Figure 5.2 a)). In contrast, a metagenome consists of many genomes with different lengths and assigned abundances, and therefore, the clones have to be sampled from many sequences. So, each genome sequence s is assigned a weight

$$w_s = l_s \times c_s \times a_s \tag{5.1}$$

where l_s represents the length, c_s the copy number and a_s the abundance value of s. The copy number can be used, for instance, to model the abundance of plasmids versus the organism genomes. For each length of the clone length distribution, the weights of all sequences are summed up to obtain the summarized weight w_{sum} that is needed to compute a sequence probability $p_s = \frac{w_s}{w_{sum}}$. Considering the overall lengths distribution, a frequency value for each sequence is then obtained (Richter et al., 2008).

After sampling the clones, the actual read sequence is sampled from the clone ends (Figure 5.2 b). Again, the read lengths can be either normally or uniformly distributed. Note, that so far, the read sequence has not yet been modified to simulate sequencing errors. This final step, the application of error models on the read sequences, is described in the next paragraph (Figure 5.2 c).

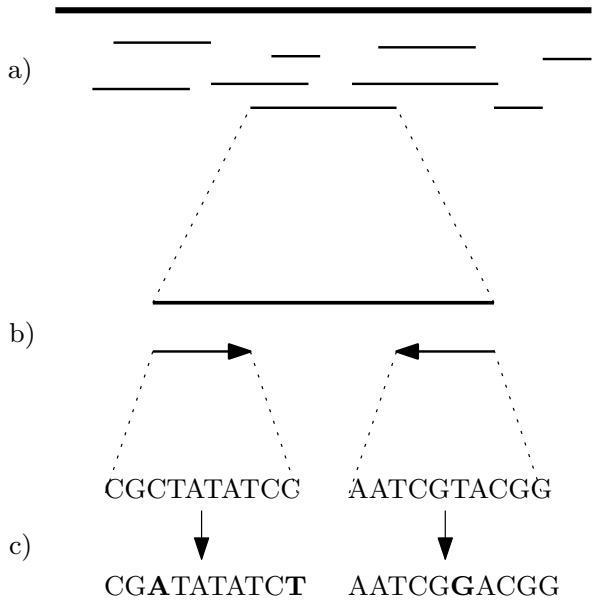


Figure 5.2: Clone and read sampling. a) Clones are randomly extracted from a single source genome sequence (thick black line). b) Clone ends are used to sample the read sequence or, optionally, the two reads for a mate-pair. c) The final step is the application of error models to the read sequences to obtain the modified, synthetic read sequence.

5.2.4 Read Sequence Modification

In the previous step, the raw read sequences were sampled from the clones. Next, these reads are modified according to the chosen error model associated to either the Sanger or Roche's 454 sequencing technology or to the empirical error model. The modification of bases reflects the fact, that, due to various technical reasons, sequencing machines do incorporate errors (indels, substitutions) into the final sequence. These sequencing errors often complicate or even mislead the subsequent processing (e.g. alignment, gene finding, etc.). Due to that reason, companies selling sequencers put lots of effort in improving the overall read quality.

The incorporation of sequencing errors is based on different statistical models depending on the chosen sequencing technology, as explained in the following subsections.

Simulation of Sanger Reads

The characteristics of Sanger sequencing have already been introduced in Section 2.2.1. MetaSim uses a similiar approach as reported in Myers (1999b). The empirical observation is that the base quality decreases towards the end of the read. The error rates for insertion, deletions and sub-

stitutions are fixed values whereas the general error rate per base position at the beginning of a read is lower than the rate at the read end. The read length is distributed either normally or uniformly. Optionally, mate-pairs can be simulated with a certain probability.

Simulation of 454 Reads

Read sequencing following the sequencing-by-synthesis approach has been introduced in Section 2.2.2 (p. 13). The main concepts of the 454 sequencing system are reported in Margulies et al. (2005). In short, the four nucleotides are periodically flowed over hundreds of thousands of beads each containing many copies of single stranded DNA fragments. Within each flow, light signals caused by the addition of nucleotides are recorded by a CCD camera. The light intensity directly corresponds to the incorporated number of nucleotides. Such, single bases and even homopolymers, i.e. consecutive stretches of equal nucleotides, can be detected. Due to chemical and technical issues, the signal is subject to fluctuations that may lead to sequencing errors. MetaSim's pyrosequencing read simulator is based on data published in 2005. To that time, an average error rate of ≈3% was reported (Margulies et al., 2005).

Our strategy to simulate 454 reads is to model the process of light emission and the detection of the observed base sequence in the base-calling procedure (for details see Richter et al. (2008)). Given a source read sequence, for each simulated flow of single nucleotides, all its homopolymers are extracted. In a second step, the homopolymer lengths are converted into virtual light emissions using a normal distribution. During the base-calling, the algorithm then calculates which probable length is to set for the observed homopolymer length.

The optional generation of 454 mate-pairs follows the protocol reported in Korbel et al. (2007): two read sequences are generated, a fixed linker sequence is concatenated connecting both reads. Finally, the error simulator produces a synthetic mate-pair.

Simulation of Reads Using Empirical Models

The characterization of empirical error rates of DNA sequencing technologies relies on the divergence between the observed and the expected nucleotide at specific base positions. In other words, the error rate probability is not directly determined by the specifications of the sequencer (e.g., light detection) but rather by the analysis of alignment experiments (empirical data). Such an experiment to detect possible base indels or substitution may be, for instance, a resequencing project: Reads are mapped to an already completed reference genome to detect sequence variations. This approach has already been applied to characterize typical errors of 454 data (Huse et al., 2007)

and is therefore an accepted method to reveal error patterns in sequencing data.

For the MetaSim software project, we received empirical error models for the Illumina sequencer generated at the Max-Planck-Institute (MPI) for Developmental Biology, Tübingen, Germany. The research group of Dr. Weigel conducted resequencing projects to study mutations in the plant *Arabidopsis thaliana*, an important model plant in molecular biology. Based on 36 bp reads, the error statistics revealed that the substitution errors occurs most frequently. Further, it was observed that the substitution rate directly depends on the base at the current and previous position. Another finding was that the error probability increases towards the read ends (from ≈0.0006 to 0.05 for a 36 bp read).

To enable the import of such statistics, a new file format was created (`.mconf`) that enables the definition of individual error probabilities for deletions, insertions or substitutions per base position adopting the general approach of the program `GenFrac` (Engle and Burks, 1994). An empirical error model is based on several mappings, each consisting of three parameters:

- the type of error,
- base at the position where the error occurs and
- base preceding the position where the error occurs.

Hence, in total, 48 mappings are possible to define an individual error model. Here is an example listing three types of error rates in a `.mconf` file:

```
# Set error rates for insertions for every A
INSERTION_ERROR (A)
 0.0187
 0.0083
 ...

# Set error rates for deletions at a C following a G
DELETION_ERROR G(C)
 0.00212
 0.00256
 ...

# All substitutions
SUBSTITUTION_ERROR
0.00699374780209504
0.00841141135948587
0.00830145870238677
0.00879477131238090
...
```

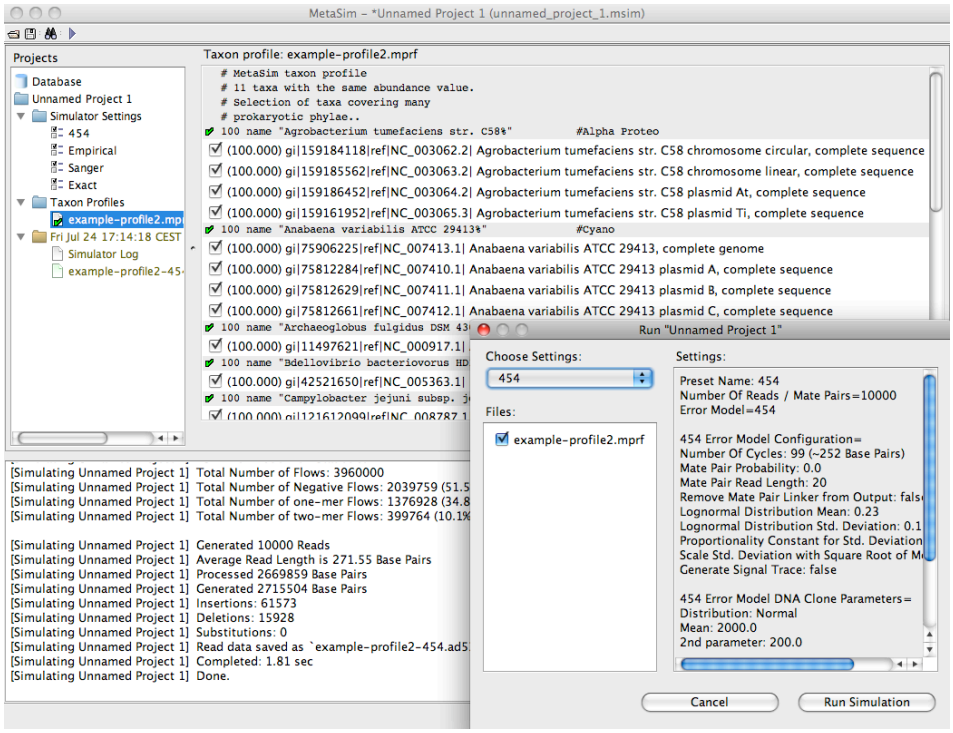


Figure 5.3: MetaSim screenshot. The GUI is divided into three parts: the project tree (top left), the database and profile view (top right) and the message panel (bottom). Within a configuration window, the user selects a taxon profile and simulator settings to finally run the simulation.

MetaSim includes an error model for 36 bp and 62 bp Illumina reads based on empirical statistics of the MPI, Tübingen. By adding or removing error mappings for specific base positions, users may create error models for other read lengths.

5.3 Results

MetaSim is a Java program and installers for different operating systems can be downloaded from: `http://www-ab.informatik.uni-tuebingen.de/software/metasim`. Besides the interactive graphical user interface (GUI) (see Figure 5.3), MetaSim can be controlled via command-line for automatic simulation runs.

To perform a simulation, the user initially creates an abundance profile file, and then chooses one of the four pre-configured simulator settings (Sanger, 454, Illumina, exact reads). The simulator settings are all adjustable (e.g. number of reads, length of clones, probability of mate-pairs,...). For each successful run, a new result folder is added to the current project

containing a log file and a preview of the final (optionally compressed) multiFASTA file. A typical simulation run which generates 100 Mbp of sequence (e.g. 400,000 454 reads of length ≈250 bp) takes less than 80 seconds on a single processor computer.

In the following section, a summary of a simulation study using MetaSim is presented.

5.3.1 Simulation Study

In Richter et al. (2008), we exemplary conducted several simulation runs to benchmark the performance of the MEGAN software (Huson et al., 2007). The goal was to measure the sensitivity and specificity of the taxonomical assignment of reads to nodes in the NCBI taxonomy as described in Section 4.3. For this analysis, three abundance profiles were generated named simLC, simMC and simHC representing low, medium and high complexity communities, respectively (in correspondence to Mavromatis et al. (2007)) (for the complete listing of organisms and abundances see supplemental Tables C.5, C.6 and C.7, p. 120 ff). The classification of community profiles into three complexity levels is based on observations of actual environmental samples. It has been observed that the microbial diversity of different ecological habitats significantly varies depending on environmental factors like, for example, energy and nutrient sources, temperature range, salinity or oxygen concentration. The ecologic difference becomes evident in the species diversity and abundance. For example, samples derived from terrestrial habitats are usually highly complex (many different organisms with similar abundance), whereas samples from extreme habitats like hydrothermal vents or bioreactors (Garcia Martin et al., 2006) represent low community complexity (few dominant organisms besides low abundance ones).

Each of the three abundance profiles have been used to generate three sets of reads differing in length, applied sequencing error models and other parameter settings (For a list of all parameter settings refer to Richter et al. (2008).). The resulting read data sets comprise ≈15 Mbp, respectively. Consequently, the number of reads differ accordingly depending on the simulation settings: for 454 reads with read length 100 bp (250 bp), 150,000 (60,000) read sequences were generated. Additionally, the Sanger read data set consists of 18,750 reads with ≈800 bp reads. The resulting read data sets were blasted against the NCBI-nr database (as of March 2008) and the BLAST output was imported and processed by MEGAN.

One of the findings was that the number of assigned reads correlates with the simulated read length. The longer a read sequence is, the more assignments to taxon nodes can be found and the less “No Hits” reads do appear (“No hits” reads could not be successfully aligned to any sequence in the database.). This is an obvious fact since longer sequences give rise to more BLAST high scoring pairs in the database than short ones. Table 5.1 shows

that there is a clear discrepancy between the assignment rate of short (100 bp) and longer read lengths: almost all sampled Sanger reads (800 bp) could be assigned to a taxon whereas many short 454 reads could not be aligned to the any database entries at all. A further result of this analysis (not presented here) is that the number of correct (true-positive) assignments to taxa increases with longer read lengths (supplementary information found in Richter et al. (2008)).

Simulation	Total Reads	% Assigned Reads	%Unassigned Reads	%No Hits
simLC-454-100	150000	83.14	0.46	16.40
simLC-454-250	60000	98.58	0.85	0.57
simLC-S-800	18750	99.45	0.55	0.00
simMC-454-100	150000	81.71	0.52	17.76
simMC-454-250	60000	98.08	1.02	0.91
simMC-S-800	18750	99.28	0.71	0.01
simHC-454-100	150000	81.68	0.51	17.81
simHC-454-250	60000	97.55	0.93	1.52
simHC-S-800	18750	99.08	0.87	0.05

Table 5.1: Summary of the read assignment rates. For each simulation run, the percentage of assigned, unassigned and "No Hits" reads are listed. "No Hits" reads are reads which did not match anything in the NCBI-nr database.

Besides the read length analysis, another question was whether the taxonomical classification of MEGAN reflects the abundance distribution of the taxon profiles. The simLC abundance profile (Table C.5) consisted only of two microbial species (*E. coli strain K12 substr. MG1655* and *M. marisnigri JR1*) that derive from two distinct superkingdoms of the taxonomy (Bacteria and Archaea). This is an interesting example because there already exist a lot of sequenced strains for the genus *Escherichia* in the databases that might complicate the assignment. By contrast, regarding *M. marisnigri JR1* only rather distantly related species had been sequenced so far. The MEGAN visualization of the phylogenetic classification of the simLC abundance profile is shown in Figure 5.4. Interestingly, the amount of assigned reads to *E. coli K12* is quite low (192) compared to the originally sampled read count (3,214). Obviously, the remaining bacterial reads have been assigned to related *E. coli* strains or other clades in the *Bacteria* subtree. this may be due to a high number of genetic functions shared among bacterial species or due to a too low bit-score quality filter. According to the expectations, most of the sampled reads (15,366 of 15,509) of *M. marsinigri JR1* have been assigned to the correct taxon. The assignment rate observed for the simLC experiment for all three simulation runs is illustrated in Fig-

Figure 5.4: MEGAN taxonomic analysis of simLC data set. Two arrows indicate the organisms whose genome sequences were used for the read sampling and simulation (Sanger technology, 800 bp). Figure taken from Richter et al. (2008).

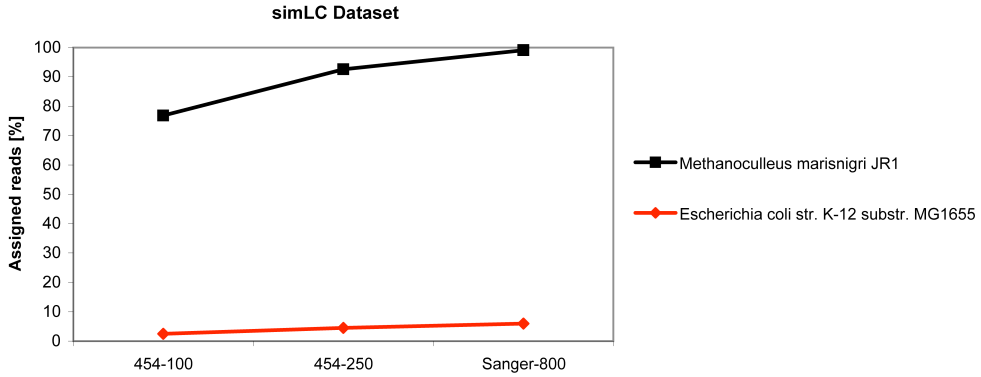


Figure 5.5: Read assignment rate (simLC). For all three simulation runs, the amount of assigned reads for *E. coli str. K12 substr. MG1655* is noticably low. Figure taken from Richter et al. (2008).

ure 5.5. This finding confirms that the assignment specificity of MEGAN's LCA algorithm is influenced by the conservation of the read sequence and the composition of the database (Huson et al., 2007). In other words, this means that the depth of the assignment in the taxonomic tree reflects the conservation of the input sequence.

The simulation experiments (and further analyses of the simMC and simHC data sets in Richter et al. (2008)) indicate that MEGAN is capable of binning most simulated reads correctly. A side effect of the LCA algorithm is that reads with many different BLAST matches may be placed to nodes higher in the taxonomy (nearer to the root). However, this approach prevents false-positive decisions that may occur if only the best BLAST hit is taken for the taxonomical placement of a read (Koski and Golding, 2001).

Eventually, at least three important issues could be tested with the nine simulated read data sets: 1) What is the assignment specificity of MEGAN when comparing different read lengths? 2) Does MEGAN's read assignment reflect the abundance profile determined by the profile files? and 3) Which sequencing technology is the most appropriate one for an efficient, taxonomical analysis of environmental samples?

In general, it becomes clear that simulation studies help to unveil strengths and potential weaknesses of computational tools.

5.4 Discussion

At the time of publication, the MetaSim software filled the gap of missing simulation software for applications in genomics (regarding sequence assembly) and metagenomics. New technical improvements and innovative,

pioneering studies in both research fields still spur software developers to implement new tools and algorithms. The testing of applications using simulated, and therefore verifiable data sets facilitate the comparison of different software tools.

The overall success of a software tool can be measured by its adaptability and applicability to new or changed conditions and specifications. Besides the built-in error models for the Sanger and Roche's 454 sequencing technologies, MetaSim provides functionality for simulations based on empirical data. A comprehensive set of error mappings allows for the individual design of error models independent of the sequencer technology or read lengths. Such, MetaSim is theoretically able to integrate any upcoming sequencing technology that will be available in the foreseeable future (e.g. as described in Eid et al. (2009) and Clarke et al. (2009)).

The situation of altering conditions appeared, for instance, when an improved version of the 454 sequencer GS20$^{\text{TM}}$, the GSFLX$^{\text{TM}}$(Droege and Hill, 2008) was introduced, three years after the initial launch of the sequencer in 2005 (Margulies et al., 2005). Longer reads and an improved read accuracy could be achieved by optimizing the chemistry and computer algorithms of the platform. For example, the read accuracy could be increased from 96% in 2005 to >99.5% in 2008. At the end of 2008/beginning of 2009, the next update, GS FLX Titanium$^{\text{TM}}$was announced promising even longer reads (≈400 bp reads). These tremendous advancements in such a short time normally would require a prompt update of read processing softwares like MetaSim. To be clearly here, this is sometimes hard to achieve. Thus, MetaSim's import functionality for empirical error models enables to keep up with recent innovations in the rush for cheaper and faster sequencer machines.

Future improvements of MetaSim should include the additional generation of quality values normally derived during the base-calling phase. These values are crucial, for instance, if reads need to be quality filtered or trimmed prior to the assembly step (see Section 2.3.1). A good idea would be the incorporation of the widely used Phred scoring scheme described in Ewing et al. (1998) and Ewing and Green (1998). Although Sanger, 454 and Illumina sequencer systems are based on different technologies, they do all provide these scores. A phred score of a base is

$$Q_{phred} = -10 \log_{10}(e) \tag{5.2}$$

whereas e is the estimated probability of a base being wrong. For example, if a base has a probability of 1/1000 of being incorrect, it is assigned a value of 30.

MetaSim has already been mentioned and applied in several studies (Hoff et al., 2009; Rokas and Abbot, 2009; Zagordi et al., 2009; Monzoorul Haque et al., 2009). Additionally, our department used MetaSim for a metagenomic

simulation study (Mitra et al., 2010) and for benchmarking an upcoming genome assembler, called LOCAS (Klein and Huson, 2009).

Chapter 6

Concluding Remarks

Central subject of this thesis was the processing and analysis of sequencing data subject to genomic and metagenomic studies. Several computational approaches were developed that deal with the complexity of highly fragmented DNA sequences occurring in genome assembly projects and in the analysis of environmental samples. The DNA sequencing landscape has been remarkably revolutionized in the last years by the second-generation sequencing technologies. The commercial launch of the first third-generation sequencing technologies is expected within the the next 2-5 years. This conclusion points out the achievements of this work in the light of the current and upcoming sequencing technologies.

The first software presented, OSLay, employs the synteny between related genome sequences to sort and order the contigs of an unfinished genome assembly with regard to a reference sequence. The problem of detecting a contig layout, which is needed for gap-closure at the end of an assembly, poses a serious challenge independent of the chosen sequencing technology. While Sanger sequencing suffers from the cloning bias and lower sequence coverages, next-generation sequencing technologies currently produce shorter reads which also leads to gapped assemblies. However, this situation may be improved when single-molecules technologies will be able to generate >1 Kbp fragments.

OSLay provides a user-friendly interface to examine the contig ordering result. Further the graphical output allows the user to detect possible misassemblies or recombinations between two genome sequences. As a precondition, the availability of a suitable reference genome is mandatory for the OSLay algorithm. Thus, OSLay might bring great potential for hybrid assembly projects that use different sequencing technologies for sequencing a single genome. Because the single assemblies likely differ from each other to a certain extent, OSLay will be able to sort the contigs of the target assemblies with regard to the reference assembly and vice versa. Since many new sequencing platforms now provide protocols for obtaining mate-pairs, an in-

corporation of this information into the OSLay algorithm would definitely improve the quality of the results. Further, the software will benefit from the increasing number of sequenced genomes that may serve as reference genomes.

The study of metagenomes is a young and vibrant research field that led to many different approaches and methods for the analysis of environmental samples. Today, metagenomic projects profit from next-generation sequencing platforms to gain exciting insights into various ecological niches. However, the short read technologies like Illumina's Genome Analyzer and ABI's SOLiD are still not frequently applied, possibly due to the limited read length. According to a simulation study (Mitra et al., 2010), this might change in the near future because the recently introduced mate-pair protocols facilitate the read classification.

The main computational challenges in metagenomics are the taxonomical and the functional analysis of environmental samples which are approached by assembly and homology- or composition-based methods (among others). In this work, a homology-based method was developed and implemented to functionally classify environmental reads. By applying a lowest-common-ancestor approach, GOAnalyzer assigns each read to its encoded biological function, molecular process and cellular component according to the hierarchically ordered terms of the Gene Ontology (Ashburner et al., 2000). Note that this read-based approach can hardly be applied to assembled contigs because they may code for more than one (partial) open-reading-frame which complicates the assignment process. In addition, since vendors of upcoming third-generation sequencing technology promise read lengths of 1 Kbp and longer, the algorithm has to be adapted in the future to deal with longer sequences. Overall, the traditional classification, gene prediction and annotation methods for metagenomic data will greatly benefit from longer read sequences derived from single-molecule sequencing.

The advancements of next-generation sequencing technologies have stimulated the field of bioinformatic software development, especially for genome assembly, genome resequencing, and metagenomics. Consequently, there is a pool of different tools that need to be evaluated by users to select the software that fits their individual needs. In this thesis, the tool MetaSim was presented that enables the simulation of synthetic reads based on given genome sequences. It allows the generation of reads considering specific error models derived from various sequencing platforms. By providing the opportunity to determine individual species abundances, MetaSim is also capable of generating simulated metagenomes that assist the benchmarking or testing of new methods and algorithms applied to environmental data.

The success of this software strongly depends on regular updates that consider the latest changes of the sequencing platforms, such as modifications of error rates or read lengths. To enable the simulation of reads derived from upcoming sequencing platforms, MetaSim provides function-

ality to read in empirical error models, thereby, allowing the generation of reads by setting error rates the lead to base modifications independent of the sequencing platform. Consequently, to ensure the future applicability of MetaSim for a broad range of users, the software should be equipped with multiple error models representing various sequencing technologies. This feature is important to keep up with the pace of the innovative developments of DNA sequencing.

To summarize, recent and upcoming sequencing technologies will continue to be the driving force to broaden the understanding of the biological diversity of our planet. On the one hand, the rapid technological developments pose many challenges with regard to biological analyses and efficient data handling and processing. One the other hand, they offer exciting prospects for the research fields of genomics and metagenomics to gain knowledge about still unknown organisms, their evolutionary history, and theirs interplays with other life forms within a community.

Besides 63 metagenomic projects, 979 microbial and 116 eukaryal genomes have been already completed (GOLD database, September 2009: `http://www.genomesonline.org`). More than hundreds of thousands of unknown genomes are still out there. Let's go for it!

Appendix A

Publications

A.1 Published Manuscripts

1. Daniel C. Richter, Stephan C. Schuster and Daniel H. Huson.
   OSLay: Optimal Syntenic Layout of Unfinished Assemblies.
   Bioinformatics. (2007), volume 23, number 13, pages 1573–1579.

   > Summary: The whole genome shotgun approach to genome sequencing results in a collection of contigs that must be ordered and oriented to facilitate efficient gap closure. We present a new tool OSLay that uses synteny between matching sequences in a target assembly and a reference assembly to layout the contigs (or scaffolds) in the target assembly. The underlying algorithm is based on maximum weight matching. The tool provides an interactive visualization of the computed layout and the result can be imported into the assembly editing tool Consed to support the design of primer pairs for gap closure.
   > Motivation: To enhance efficiency in the gap closure phase of a genome project it is crucial to know which contigs are adjacent in the target genome. Related genome sequences can be used to layout contigs in an assembly.
   > Availability: OSLay is freely available from: `http://www-ab.informatik.uni-tuebingen.de/software/oslay`

2. Daniel H. Huson, Daniel C. Richter, Christian Rausch, Tobias Dezulian, Markus Franz and Regula Rupp.
   Dendroscope: An interactive viewer for large phylogenetic trees.
   BMC Bioinformatics. (2007), volume 8, pages 460

   > Background: Research in evolution requires software for visualizing and editing phylogenetic trees, for increasingly very large datasets, such as arise in expression analysis or metagenomics, for example. It would be desirable to have a program that provides these services in an efficient and

user-friendly way, and that can be easily installed and run on all major operating systems. Although a large number of tree visualization tools are freely available, some as a part of more comprehensive analysis packages, all have drawbacks in one or more domains. They either lack some of the standard tree visualization techniques or basic graphics and editing features, or they are restricted to small trees containing only tens of thousands of taxa. Moreover, many programs are difficult to install or are not available for all common operating systems.
Results: We have developed a new program, Dendroscope, for the interactive visualization and navigation of phylogenetic trees. The program provides all standard tree visualizations and is optimized to run interactively on trees containing hundreds of thousands of taxa. The program provides tree editing and graphics export capabilities. To support the inspection of large trees, Dendroscope offers a magnification tool. The software is written in Java 1.4 and installers are provided for Linux/Unix, MacOS X and Windows XP.
Conclusion: Dendroscope is a user-friendly program for visualizing and navigating phylogenetic trees, for both small and large datasets.

3. Daniel C. Richter, Felix Ott, Alexander F. Auch, Ramona Schmid, Daniel H. Huson.
MetaSimA Sequencing Simulator for Genomics and Metagenomics.
PLoS ONE (2008), volume 3, number 10, pages 359–360.

Background: The new research field of metagenomics is providing exciting insights into various, previously unclassified ecological systems. Next-generation sequencing technologies are producing a rapid increase of environmental data in public databases. There is great need for specialized software solutions and statistical methods for dealing with complex metagenome data sets.
Methodology/Principal Findings: To facilitate the development and improvement of metagenomic tools and the planning of metagenomic projects, we introduce a sequencing simulator called MetaSim. Our software can be used to generate collections of synthetic reads that reflect the diverse taxonomical composition of typical metagenome data sets. Based on a database of given genomes, the program allows

the user to design a metagenome by specifying the number of genomes present at different levels of the NCBI taxonomy, and then to collect reads from the metagenome using a simulation of a number of different sequencing technologies. A population sampler optionally produces evolved sequences based on source genomes and a given evolutionary tree.
Conclusions/Significance: MetaSim allows the user to simulate individual read datasets that can be used as standardized test scenarios for planning sequencing projects or for benchmarking metagenomic software.

4. Daniel H. Huson, Daniel C. Richter, S. Mitra, Alexander F. Auch and Stephan C. Schuster.
Methods for comparative metagenomics.
BMC Bioinformatics (2009), volume 10 (Suppl 1):S12

Background: Metagenomics is a rapidly growing field of research that aims at studying uncultured organisms to understand the true diversity of microbes, their functions, cooperation and evolution, in environments such as soil, water, ancient remains of animals, or the digestive system of animals and humans. The recent development of ultra-high throughput sequencing technologies, which do not require cloning or PCR amplification, and can produce huge numbers of DNA reads at an affordable cost, has boosted the number and scope of metagenomic sequencing projects. Increasingly, there is a need for new ways of comparing multiple metagenomics datasets, and for fast and user-friendly implementations of such approaches.
Results: This paper introduces a number of new methods for interactively exploring, analyzing and comparing multiple metagenomic datasets, which will be made freely available in a new, comparative version 2.0 of the stand-alone metagenome analysis tool MEGAN.
Conclusion: There is a great need for powerful and user-friendly tools for comparative analysis of metagenomic data and MEGAN 2.0 will help to fill this gap.

A.2 Submitted Manuscripts

5. Fangqing Zhao, Ji Qi, Daniel C. Richter, Anne Buboltz, Daniel H. Huson and Stephan C. Schuster.
Metagenomic analysis of the microbiome associated with the hairs of extinct woolly mammoths.
PNAS (2009)

Despite the low temperature and low nutrients in the permafrost, a large diversity of microorganisms occupies this environmental niche. Here, we study the diversity and community structure of microbes isolated from unique permafrost samples, 13 hair and 1 bone specimen from woolly mammoths. After analyzing approximate 1 Gb of environmental sequences from these specimens, we determined that bone sample contains more soil bacteria suggesting that the taxonomic composition of bacterial assemblages varies greatly between these samples. A number of putative ancient bacteria were identified, as they had an elevated DNA damage rate compared to modern strains, such as strains from Pseudomonas, Acinetobacter, Polaromonas, Caulobacter and Stenotrophomonas. A small proportion of fungal sequences were found in the mammoth biomes, with six out of the ten most abundant species being plant pathogens. Psychrophilic Flavobacterium spp., Psychroflexus spp. and Psychrobacter spp. dominate the cold adapted bacteria. Comparisons between the mammoth biomes and modern microbial communities would shed light on both the ancient microorganisms associated with woolly mammoths but also those inhabit the permafrost.

Appendix B

Contribution

This thesis describes several algorithms and their implementations that resulted from my studies during my PhD. Here, I would like to separate the contribution of other colleagues or collaborators from my own work.

Chapter 3. OSLay: Syntenic Layout of Unfinished Assemblies

The first time I got in touch with the practical problem of ordering and sorting contigs for genome assembly was during my Diploma thesis at the department of Daniel H. Huson. To that time, my former colleague, Christian Rausch, gave a talk about that topic at the GCB 2004 (Friedrichs et al., 2004). Another former member of the group, Olaf D. Friedrichs, wrote a rudimentary script implementing the algorithm. After my basic JAVA implementation of this approach using the CGViz framework developed at our department (Friedrichs et al., 2003), I spent a lot of time in the first year of my PhD to further improve the algorithm. In 2006, during a three-month research stay in the group of Stephan C. Schuster (PennState University, PA, USA), I elaborated on the implementation benefiting from valuable contributions by Stephan C. Schuster and his group members. Eventually, in 2007, I wrote the manuscript (Richter et al., 2007) and interacted with the editor and reviewers, whereas Daniel H. Huson and Stephan C. Schuster contributed many useful comments.

Chapter 4. Metagenome Analysis using MEGAN

In 2006, Daniel H. Huson was a co-author of the pioneering "metagenomics to paleogenomics" publication (Poinar et al., 2006) that employed a metagenomic approach to sequence DNA of a woolly mammoth from Siberia. This study has sparked my interests, so I decided to set my research focus on the computational analysis of metagenomes. The MEGAN software had been already applied in the mentioned mammoth study to taxonomically classify the sequencing reads. The LCA algorithm was described in Huson et al. (2007).

Becoming aware that MEGAN lacked an alternative graphical presentation of taxonomical results, I first started to work on an universal charting tool for MEGAN. At the same time, I came across the list of "Microbial Genome Properties" of the NCBI (`http://www.ncbi.nlm.nih.gov/genomes/lproks.cgi`). I realized that no tool exists to analyze metagenomes regarding its organism properties, so I contributed code to provide a built-in microbial attribute browser within MEGAN.

In the last months of my PhD, I devoted all my energy into the implementation of a new module that enables MEGAN users to functionally classify metagenomic read sequences. Since MEGAN uses the BLAST output to infer the taxon assignments for reads, I conceived a homology-based approach to annotate reads based on BLASTX matches found in the NCBI-nr database. I decided to map the reads onto the terms of the Gene Ontology (Ashburner et al., 2000), whereas Daniel H. Huson supported me throughout the conception of the algorithm. Additionally, he helped me to integrate my code into the MEGAN sources. The visualization of the functional classification was implemented using the `yFiles` library as recommended by Michael (Miggi) Schröder, a former Diploma student in our group and current staff member of the company yWorks (`http://www.yworks.com`).

Chapter 5. MetaSim: A Sequencing Simulator for Genomics and Metagenomics

In 2006, Ramona Schmid, a Diploma student supervised by Daniel H. Huson, worked on a read simulation tool, called ReadSim (`http://www-ab.informatik.uni-tuebingen.de/software/readsim`). It already contained error models for base modifications for the Sanger and 454 sequencing technology. However, this work remained unpublished. Two years later, Felix Ott, a Diploma student supervised by my colleague Alexander Auch and me, was assigned the task to implement a new simulation tool that additionally simulates metagenomic data sets. Because some (theoretical) work was already accomplished by Ramona, Felix could build on the concepts of ReadSim. However, he did a great job in developing a comprehensive and intuitive software (MetaSim), whereas Alexander Auch, Daniel H. Huson and I contributed the conceptual ideas.

Then, Alexander and I conceived the simulation study and the design of the publication (Richter et al., 2008). Alexander performed the BLAST runs and we analyzed and interpreted the results. I wrote the manuscript, selected the journal and interacted with editors and reviewers.

Appendix C

Supplementary Material

C.1 Metagenome Analysis using MEGAN

Category	c	Properties
Gram Stain	✓	positive (positive stain observed) negative unknown
Endospores	✓	yes no unknown
Motility	✓	yes no (organism has not. yet. been found to be motile)
Shape		e.g. coccoid, rod-shaped, or spiral-shaped
Arrangement		e.g. single, pairs, tetrad, filaments, rosettes, or chains

Table C.1: Cellular features. General characteristics describing microbial organisms. Column **c** indicates controlled vocabulary.

Category	c	Properties
Salinity	✓	non-halophilic (0-2% NaCl) mesophilic (2-5% NaCl) moderate halophile (5-20% NaCl) extreme halophile (20-30% NaCl)
Oxygen Req.	✓	unknown aerobic microaerophilic facultative anaerobic
Habitat	✓	unknown host-associated aquatic terrestrial specialized multiple

Table C.2: Environmental features. Description of the environment the organism prefers to live at. Column **c** indicates controlled vocabulary.

Category	c	Properties
Opt. Temp.		Degree celsius the organism grows best at.
Range	✓	unknown cryophilic (-30 to -2°C) psychrophilic (-1 to +10°C) mesophilic (+11 to +45°C) thermophilic (+46 to +75°C) hyperthermophilic (above +75°C)
Habitat	✓	unknown host-associated aquatic terrestrial specialized multiple

Table C.3: Range of temperatures. Range of temperature the organisms prefers to grow at. Column **c** indicates controlled vocabulary.

Category	c	Properties
Pathogenic in		Organisms that this bacterium is pathogenic in
Disease		Name of disease caused by a pathogenic bacterium

Table C.4: Pathogenicity. Information about the pathogenicity of an organism. Column **c** indicates controlled vocabulary.

C.2 MetaSim: A Sequencing Simulator for Genomics and Metagenomics

Abdce	Species	Mbp	454-100	454-250	S-800
90	Methanoculleus marisnigri JR1	2.5	82.70	82.16	82.71
10	Escherichia coli str. K-12 substr. MG1655	4.6	17.30	17.39	17.29

Table C.5: Species abundance and percentage of sampled reads of the simLC data set. 454-100: 454 technology. 150.000 reads (length: 100). 454-250: 454 technology. 60.000 reads (length: 250). S-800: Sanger technology. 18.750 reads (length: 800)

Abdce	Species	Mbp	454-100	454-250	S-800
100	Pseudomonas fluorescens PfO-1	6.4	38.42	38.39	38.18
100	Shigella dysenteriae Sd197	4.6	27.07	27.47	27.25
80	Pasteurella multocida subsp. multocida str. Pm70	2.3	10.82	10.87	10.81
50	Buchnera aphidicola str. APS	6.6	1.97	1.94	1.78
50	Francisella tularensis subsp. tularensis Schu 4	1.9	5.69	5.6	5.56
25	Alcanivorax borkumensis SK2	3.1	4.68	4.57	4.6
25	Candidatus Blochmannia floridanus	7.1	1.03	1.08	1.21
25	Pseudomonas entomophila L48	5.9	8.89	8.65	9.26
5	Escherichia coli str. K12 substr. MG1655	4.6	1.43	1.43	1.40

Table C.6: Species abundance and percentage of sampled reads of the simMC data set. 454-100: 454 technology. 150.000 reads (length: 100). 454-250: 454 technology. 60.000 reads (length: 250). S-800: Sanger technology. 18.750 reads (length: 800)

Abdce	Species	Mbp	454-100	454-250	S-800
100	Agrobacterium tumefaciens str. C58	5.7	11.7	11.7	11.3
100	Anabaena variabilis ATCC 29413	7.1	14.7	14.9	14.6
100	Archaeoglobus fulgidus DSM 4304	2.2	4.54	4.41	4.55
100	Bdellovibrio bacteriovorus HD100	3.8	7.84	7.79	7.83
100	Campylobacter jejuni subsp. jejuni 81-176	1.7	3.52	3.6	3.57
100	Clostridium acetobutylicum ATCC 824	4.1	8.6	8.56	8.49
100	Lactococcus lactis subsp. cremoris SK11	2.6	5.38	5.32	5.54
100	Nitrosomonas europaea ATCC 19718	2.8	5.81	5.66	5.59
100	Pseudomonas aeruginosa PA7	6.6	13.6	13.6	14
100	Streptomyces coelicolor A3(2)	9.1	18.7	18.9	18.9
100	Sulfolobus tokodaii str. 7	2.7	5.64	5.59	5.6

Table C.7: Species abundance and percentage of sampled reads of the simHC data set. 454-100: 454 technology. 150.000 reads (length: 100). 454-250: 454 technology. 60.000 reads (length: 250). S-800: Sanger technology. 18.750 reads (length: 800)

Bibliography

Adessi C, Matton G, Ayala G, Turcatti G, Mermod JJ, Mayer P, Kawashima E, 2000. Solid phase DNA amplification: characterisation of primer attachment and amplification mechanisms. Nucleic Acids Res, 28(20):E87.

Alterovitz G, Xiang M, Mohan M, Ramoni MF, 2007. GO PaD: the Gene Ontology Partition Database. Nucleic Acids Res, 35(Database issue):D322–7.

Altschul SF, Gish W, Miller W, Myers EW, Lipman DJ, 1990. Basic local alignment search tool. 215:403–410.

Ashburner M, Ball CA, Blake JA, et al., 2000. Gene ontology: tool for the unification of biology. The Gene Ontology Consortium. Nat Genet, 25(1):25–29.

Bard JBL, Rhee SY, 2004. Ontologies in biology: design, applications and future challenges. Nat Rev Genet, 5(3):213–222.

Batzoglou S, Jaffe DB, Stanley K, Butler J, Gnerre S, Mauceli E, Berger B, Mesirov JP, Lander ES, 2002. ARACHNE: a whole-genome shotgun assembler. Genome Res, 12(1):177–89.

Bentley D, 2006. Whole-genome re-sequencing. Current Opinion in Genetics & Development, 16:545–552.

Bentley DR, Balasubramanian S, Swerdlow HP, et al., 2008. Accurate whole human genome sequencing using reversible terminator chemistry. Nature, 456(7218):53–9.

Berg RD, 1996. The indigenous gastrointestinal microflora. Trends Microbiol, 4(11):430–5.

Blow N, 2007. Genomics: the personal side of genomics. Nature, 449(7162):627–30.

Blow N, 2008a. DNA sequencing: generation next-next. Nature Methods, 5(308):267–274.

Blow N, 2008b. Metagenomics: exploring unseen communities. Nature, 453(7195):687–90.

Bohannon J, 2007. Metagenomics. Ocean study yields a tidal wave of microbial DNA. Science, 315(5818):1486–7.

Bohnebeck U, Lombardot T, Kottmann R, Glöckner FO, 2008. MetaMine–a tool to detect and analyse gene patterns in their environmental context. BMC Bioinformatics, 9:459.

Braslavsky I, Hebert B, Kartalov E, Quake SR, 2003. Sequence information can be obtained from single DNA molecules. Proc Natl Acad Sci U S A, 100(7):3960–4.

Breitbart M, Salamon P, Andresen B, Mahaffy JM, Segall AM, Mead D, Azam F, Rohwer F, 2002. Genomic analysis of uncultured marine viral communities. Proc Natl Acad Sci U S A, 99(22):14250–5.

Chan CKK, Hsu AL, Halgamuge SK, Tang SL, 2008. Binning sequences using very sparse labels within a metagenome. BMC Bioinformatics, 9:215.

Chatterji S, Yamazaki I, Bai Z, Eisen J, 2008. CompostBin: A DNA composition-based algorithm for binning environmental shotgun reads. In Vingron M, Wong L, editors, RECOMB. Springer, volume 4955 of *Lecture Notes in Computer Science*, pp. 17–28.

Check Hayden E, 2009. Genome sequencing: the third generation. Nature, 457(7231):768–9.

Clarke J, Wu HC, Jayasinghe L, Patel A, Reid S, Bayley H, 2009. Continuous base identification for single-molecule nanopore DNA sequencing. Nat Nanotechnol, 4(4):265–270.

Cox-Foster DL, Conlan S, Holmes EC, et al., 2007. A metagenomic survey of microbes in honey bee colony collapse disorder. Science, 318(5848):283–7.

Culley AI, Lang AS, Suttle CA, 2006. Metagenomic analysis of coastal RNA virus communities. Science, 312(5781):1795–8.

Dahm R, 2008. The First Discovery of DNA. American Scientist, 96(4):320.

Daniel R, 2005. The metagenomics of soil. Nat Rev Microbiol, 3(6):470–8.

Delcher AL, Harmon D, Kasif S, White O, Salzberg SL, 1999. Improved microbial gene identification with GLIMMER. Nucleic Acids Res, 27(23):4636–4641.

Diaz N, Krause L, Goesmann A, Niehaus K, Nattkemper T, 2009. TACOA -Taxonomic classification of environmental genomic fragments using a kernelized nearest neighbor approach. BMC Bioinformatics, 10(1):56.

Dinsdale EA, Edwards RA, Hall D, et al., 2008. Functional metagenomic profiling of nine biomes. Nature, 452(7187):629–632.

Dressman D, Yan H, Traverso G, Kinzler KW, Vogelstein B, 2003. Transforming single DNA molecules into fluorescent magnetic particles for detection and enumeration of genetic variations. Proc Natl Acad Sci U S A, 100(15):8817–22.

Droege M, Hill B, 2008. The Genome Sequencer FLX System–longer reads, more applications, straight forward bioinformatics and more complete data sets. J Biotechnol, 136(1-2):3–10.

Edwards RA, Rodriguez-Brito B, Wegley L, Haynes M, Breitbart M, Peterson DM, Saar MO, Alexander S, Alexander ECJ, Rohwer F, 2006. Using pyrosequencing to shed light on deep mine microbial ecology. BMC Genomics, 7:57.

Eid J, Fehr A, Gray J, et al., 2009. Real-time DNA sequencing from single polymerase molecules. Science, 323(5910):133–138.

Eisen JA, 2007. Environmental shotgun sequencing: its potential and challenges for studying the hidden world of microbes. PLoS Biol, 5(3):e82.

Engle ML, Burks C, 1993. Artificially generated data sets for testing DNA sequence assembly algorithms. Genomics, 16(1):286–288.

Engle ML, Burks C, 1994. GenFrag 2.1: new features for more robust fragment assembly benchmarks. Comput Appl Biosci, 10(5):567–568.

Ewing B, Green P, 1998. Base-calling of automated sequencer traces using phred. II. Error probabilities. Genome Res, 8(3):186–194.

Ewing B, Hillier L, Wendl MC, Green P, 1998. Base-calling of automated sequencer traces using phred. I. Accuracy assessment. Genome Res, 8(3):175–185.

Farrer RA, Kemen E, Jones JDG, Studholme DJ, 2009. De novo assembly of the Pseudomonas syringae pv. syringae B728a genome using Illumina/Solexa short sequence reads. FEMS Microbiol Lett, 291(1):103–11.

Fedurco M, Romieu A, Williams S, Lawrence I, Turcatti G, 2006. BTA, a novel reagent for DNA attachment on glass and efficient generation of solid-phase amplified DNA colonies. Nucleic Acids Res, 34(3):e22.

Field D, Garrity G, Gray T, et al., 2008. The minimum information about a genome sequence (MIGS) specification. Nat Biotechnol, 26(5):541–7.

Finn RD, Tate J, Mistry J, et al., 2008. The Pfam protein families database. Nucleic Acids Res, 36(Database issue):D281–D288.

Fleischmann RD, Adams MD, White O, Clayton RA, Kirkness EF, Kerlavage AR, Bult CJ, Tomb JF, Dougherty BA, Merrick JM, 1995. Whole-genome random sequencing and assembly of Haemophilus influenzae Rd. Science, 269(5223):496–512.

Franklin RE, Gosling RG, 1953. Molecular configuration in sodium thymonucleate. Nature, 171(4356):740–1.

Fraser CM, Gocayne JD, White O, et al., 1995. The minimal gene complement of Mycoplasma genitalium. Science, 270(5235):397–403.

Frias-Lopez J, Shi Y, Tyson GW, Coleman ML, Schuster SC, Chisholm SW, Delong EF, 2008. Microbial community gene expression in ocean surface waters. Proc Natl Acad Sci U S A, 105(10):3805–3810.

Friedrichs OD, Dezulian T, Huson D, 2003. A meta-viewer for biomolecular data. GI Jahrestagung, 1:375–380.

Friedrichs OD, Halpern AL, Lippert R, Rausch C, Schuster SC, Huson DH, 2004. Syntenic Layout of Two Assemblies of Related Genomes. In Giegerich R, Stoye J, editors, German Conference on Bioinformatics. GI, volume 53, pp. 3–12.

Gabow HN, 1976. An efficient implementation of Edmonds' algorithm for maximum matching on graphs. Journal of the ACM, 23:221–234.

Garcia Martin H, Ivanova N, Kunin V, et al., 2006. Metagenomic analysis of two enhanced biological phosphorus removal (EBPR) sludge communities. Nat Biotechnol, 24(10):1263–1269.

Gilbert JA, Field D, Huang Y, Edwards R, Li W, Gilna P, Joint I, 2008. Detection of large numbers of novel sequences in the metatranscriptomes of complex marine microbial communities. PLoS One, 3(8):e3042.

Gilbert MTP, Tomsho LP, Rendulic S, et al., 2007. Whole-genome shotgun sequencing of mitochondria from ancient hair shafts. Science, 317(5846):1927–30.

Gill SR, Pop M, Deboy RT, Eckburg PB, Turnbaugh PJ, Samuel BS, Gordon JI, Relman DA, Fraser-Liggett CM, Nelson KE, 2006. Metagenomic analysis of the human distal gut microbiome. Science, 312(5778):1355–1359.

Goldberg SMD, Johnson J, Busam D, et al., 2006. A Sanger/pyrosequencing hybrid approach for the generation of high-quality draft assemblies of marine microbial genomes. Proc Natl Acad Sci U S A, 103(30):11240–11245.

Gordon D, Abajian C, Green P, 1998. Consed: a graphical tool for sequence finishing. Genome Res, 8(3):195–202.

Green RE, Krause J, Ptak SE, et al., 2006. Analysis of one million base pairs of Neanderthal DNA. Nature, 444(7117):330–6.

Green RE, Malaspinas AS, Krause J, et al., 2008. A complete Neandertal mitochondrial genome sequence determined by high-throughput sequencing. Cell, 134(3):416–26.

Haeckel E, 1866. Generelle Morphologie der Organismen. Reimer, Berlin.

Haft DH, Selengut JD, White O, 2003. The TIGRFAMs database of protein families. Nucleic Acids Res, 31(1):371–373.

Handelsman J, Rondon MR, Brady SF, Clardy J, Goodman RM, 1998. Molecular biological access to the chemistry of unknown soil microbes: a new frontier for natural products. Chem Biol, 5(10):R245–R249.

Harding EF, 1971. The Probabilities of Rooted Tree-Shapes Generated by Random Bifurcation. Advances in Applied Probability, 3(1):44–77.

Harrington ED, Singh AH, Doerks T, Letunic I, von Mering C, Jensen LJ, Raes J, Bork P, 2007. Quantitative assessment of protein function prediction from metagenomics shotgun sequences. Proc Natl Acad Sci U S A, 104(35):13913–8.

Hirschman L, Clark C, Cohen KB, et al., 2008. Habitat-Lite: a GSC case study based on free text terms for environmental metadata. OMICS, 12(2):129–136.

Hoff KJ, Lingner T, Meinicke P, Tech M, 2009. Orphelia: predicting genes in metagenomic sequencing reads. Nucleic Acids Res, 37(Web Server issue):W101–5.

Hofreiter M, 2008. Paleogenomics. C R Palevol, 7(113-124).

Hugenholtz P, 2002. Exploring prokaryotic diversity in the genomic era. Genome Biol, 3(2):REVIEWS0003.

Huse SM, Huber JA, Morrison HG, Sogin ML, Welch DM, 2007. Accuracy and quality of massively parallel DNA pyrosequencing. Genome Biol, 8(7):R143.

Huson DH, Auch AF, Qi J, Schuster SC, 2007. MEGAN analysis of metagenomic data. Genome Res, 17(3):377–386.

Huson DH, Richter DC, Mitra S, Auch AF, Schuster SC, 2009. Methods for comparative metagenomics. BMC Bioinformatics, 10 (Suppl 1):S12.

Hutchison CAr, 2007. DNA sequencing: bench to bedside and beyond. Nucleic Acids Res, 35(18):6227–6237.

Jeong H, Kim JF, 2008. An Optimized Strategy for Genome Assembly of Sanger/pyrosequencing Hybrid Data using Available Software. Genomics and Informatics, 6(2):87–90.

Jothi R, Cuddapah S, Barski A, Cui K, Zhao K, 2008. Genome-wide identification of in vivo protein-DNA binding sites from ChIP-Seq data. Nucleic Acids Res, 36(16):5221–31.

Jukes T, Cantor C, 1969. Evolution of Protein Molecules. In Munro HN, editor, Mammalian Protein Metabolism, New York, NY: Academic Press, pp. 21–132.

Kanehisa M, Goto S, 2000. KEGG: kyoto encyclopedia of genes and genomes. Nucleic Acids Res, 28(1):27–30.

Klein JD, Huson DH, 2009. LOCAS - A low coverage assembly tool for resequencing projects with short reads. unpublished.

Korbel JO, Urban AE, Affourtit JP, et al., 2007. Paired-end mapping reveals extensive structural variation in the human genome. Science, 318(5849):420–426.

Koski LB, Golding GB, 2001. The closest BLAST hit is often not the nearest neighbor. J Mol Evol, 52(6):540–542.

Kottmann R, Gray T, Murphy S, Kagan L, Kravitz S, Lombardot T, Field D, Glockner FO, 2008. A standard MIGS/MIMS compliant XML Schema: toward the development of the Genomic Contextual Data Markup Language (GCDML). OMICS, 12(2):115–121.

Krause L, Diaz NN, Goesmann A, Kelley S, Nattkemper TW, Rohwer F, Edwards RA, Stoye J, 2008a. Phylogenetic classification of short environmental DNA fragments. Nucleic Acids Res, 36(7):2230–9.

Krause L, Diaz NN, Goesmann A, Kelley S, Nattkemper TW, Rohwer F, Edwards RA, Stoye J, 2008b. Phylogenetic classification of short environmental DNA fragments. Nucleic Acids Res, 36(7):2230–2239.

Kunin V, Copeland A, Lapidus A, Mavromatis K, Hugenholtz P, 2008. A bioinformatician's guide to metagenomics. Microbiol Mol Biol Rev, 72(4):557–78, Table of Contents.

Kurtz S, Phillippy A, Delcher AL, Smoot M, Shumway M, Antonescu C, Salzberg SL, 2004. Versatile and open software for comparing large genomes. Genome Biol, 5(2):R12.

Kyrpides NC, 2009. Fifteen years of microbial genomics: meeting the challenges and fulfilling the dream. Nat Biotechnol, 27(7):627–32.

Lander ES, Linton LM, Birren B, et al., 2001. Initial sequencing and analysis of the human genome. Nature, 409(6822):860–921.

Langmead B, Trapnell C, Pop M, Salzberg SL, 2009. Ultrafast and memory-efficient alignment of short DNA sequences to the human genome. Genome Biol, 10(3):R25.

Levene P, 1919. The structure of yeast nucleic acid. J Biol Chem, 40(2):415–24.

Levy S, Sutton G, Ng PC, et al., 2007. The diploid genome sequence of an individual human. PLoS Biol, 5(10):e254.

Ley TJ, Mardis ER, Ding L, et al., 2008. DNA sequencing of a cytogenetically normal acute myeloid leukaemia genome. Nature, 456(7218):66–72.

Li H, Ruan J, Durbin R, 2008. Mapping short DNA sequencing reads and calling variants using mapping quality scores. Genome Res, 18(11):1851–8.

MacLean D, Jones JDG, Studholme DJ, 2009. Application of 'next-generation' sequencing technologies to microbial genetics. Nat Rev Microbiol, 7(4):287–96.

Marguerat S, Wilhelm BT, Bahler J, 2008. Next-generation sequencing: applications beyond genomes. Biochem Soc Trans, 36(Pt 5):1091–1096.

Margulies M, Egholm M, Altman W, et al., 2005. Genome sequencing in microfabricated high-density picolitre reactors. Nature, 437(7057):376–380.

Markowitz VM, Ivanova NN, Szeto E, et al., 2008. IMG/M: a data management and analysis system for metagenomes. Nucleic Acids Res, 36(Database issue):D534–8.

Mavromatis K, Ivanova N, Barry K, et al., 2007. Use of simulated data sets to evaluate the fidelity of metagenomic processing methods. Nat Methods, 4(6):495–500.

Maxam A, Gilbert W, 1977. A new method for sequencing DNA. Proc. Natl. Acad. Sci., 74:560–564.

McHardy AC, Mart'n HG, Tsirigos A, Hugenholtz P, Rigoutsos I, 2007. Accurate phylogenetic classification of variable-length DNA fragments. Nat Methods, 4(1):63–72.

Meyer F, Paarmann D, D'Souza M, et al., 2008. The metagenomics RAST server - a public resource for the automatic phylogenetic and functional analysis of metagenomes. BMC Bioinformatics, 9:386.

Miller W, Drautz DI, Janecka JE, et al., 2009a. The mitochondrial genome sequence of the Tasmanian tiger (Thylacinus cynocephalus). Genome Res, 19(2):213–20.

Miller W, Drautz DI, Janecka JE, et al., 2009b. The mitochondrial genome sequence of the Tasmanian tiger (Thylacinus cynocephalus). Genome Res, 19(2):213–220.

Mitra S, Klar B, Huson DH, 2009. Visual and statistical comparison of metagenomes. Bioinformatics, 25(15):1849–1855.

Mitra S, Schubach M, Huson DH, 2010. Short clones or long clones? A simulation study on the use of paired-reads in metagenomics. submitted to APBC.

Monzoorul Haque M, Ghosh TS, Komanduri D, Mande SS, 2009. SOrt-ITEMS: Sequence orthology based approach for improved taxonomic estimation of metagenomic sequences. Bioinformatics, 25(14):1722–30.

Myers EW, 1999a. Whole-Genome DNA Sequencing. Computing in Science and Engineering, IEEE.

Myers EW, Sutton GG, Delcher AL, et al., 2000. A whole-genome assembly of Drosophila. Science, 287(5461):2196–204.

Myers G, 1999b. A dataset generator for whole genome shotgun sequencing. In Proc Int Conf Intell Syst Mol Biol. pp. 202–10.

Noguchi H, Park J, Takagi T, 2006. MetaGene: prokaryotic gene finding from environmental genome shotgun sequences. Nucleic Acids Res, 34(19):5623–30.

Noguchi H, Taniguchi T, Itoh T, 2008. MetaGeneAnnotator: detecting species-specific patterns of ribosomal binding site for precise gene prediction in anonymous prokaryotic and phage genomes. DNA Res, 15(6):387–396.

Overbeek R, Begley T, Butler RM, et al., 2005. The subsystems approach to genome annotation and its use in the project to annotate 1000 genomes. Nucleic Acids Res, 33(17):5691–5702.

Pennisi E, 2006. Genomics. On your mark. Get set. Sequence! Science, 314(5797):232.

Perkel JM, 2009. Sanger who? Sequencing the next generation. Science, DOI: 10.1126/science.opms.p0800033.

Pevzner PA, Tang H, Waterman MS, 2001. An Eulerian path approach to DNA fragment assembly. Proc Natl Acad Sci U S A, 98(17):9748–53.

Pignatelli M, Aparicio G, Blanquer I, Hernández V, Moya A, Tamames J, 2008. Metagenomics reveals our incomplete knowledge of global diversity. Bioinformatics, 24(18):2124–2125.

Poinar HN, Schwarz C, Qi J, et al., 2006. Metagenomics to paleogenomics: large-scale sequencing of mammoth DNA. Science, 311(5759):392–394.

Pop M, 2009. Genome assembly reborn: recent computational challenges. Brief Bioinform, 10(4):354–66.

Pop M, Salzberg S, Shumway M, 2002. Genome sequence assembly: algorithms and issues. IEEE Computer Society, 35(7):47–54.

Poretsky R, Hewson I, Sun S, Allen A, Zehr J, Moran M, 2009. Comparative day/night metatranscriptomic analysis of microbial communities in the North Pacific subtropical gyre. Environ Microbiol.

Pruitt KD, Tatusova T, Klimke W, Maglott DR, 2009. NCBI Reference Sequences: current status, policy and new initiatives. Nucleic Acids Res, 37(Database issue):D32–6.

Pushkarev D, Neff NF, Quake SR, 2009. Single-molecule sequencing of an individual human genome. Nat Biotechnol.

Qi W, Nong G, Preston JF, Ben-Ami F, Ebert D, 2009. Comparative metagenomics of Daphnia symbionts. BMC Genomics, 10:172.

Raes J, Foerstner KU, Bork P, 2007. Get the most out of your metagenome: computational analysis of environmental sequence data. Curr Opin Microbiol, 10(5):490–8.

Ramírez O, Gigli E, Bover P, Alcover JA, Bertranpetit J, Castresana J, Lalueza-Fox C, 2009. Paleogenomics in a temperate environment: shotgun sequencing from an extinct Mediterranean caprine. PLoS One, 4(5):e5670.

Reference Genome Group of the Gene Ontology C, 2009. The Gene Ontology's Reference Genome Project: a unified framework for functional annotation across species. PLoS Comput Biol, 5(7):e1000431.

Reinhardt JA, Baltrus DA, Nishimura MT, Jeck WR, Jones CD, Dangl JL, 2009. De novo assembly using low-coverage short read sequence data from the rice pathogen Pseudomonas syringae pv. oryzae. Genome Res, 19(2):294–305.

Rendulic S, Jagtap P, Rosinus A, et al., 2004. A Predator unmasked: The life cycle of *Bdellovibrio bacteriovorus* from a genomic perspective. Science, 303:689–692.

Rhee SY, Wood V, Dolinski K, Draghici S, 2008. Use and misuse of the gene ontology annotations. Nat Rev Genet, 9(7):509–515.

Richter DC, Ott F, Auch AF, Schmid R, Huson DH, 2008. MetaSim: a sequencing simulator for genomics and metagenomics. PLoS ONE, 3(10):e3373.

Richter DC, Schuster SC, Huson DH, 2007. OSLay: optimal syntenic layout of unfinished assemblies. Bioinformatics, 23(13):1573–1579.

Rokas A, Abbot P, 2009. Harnessing genomics for evolutionary insights. Trends Ecol Evol, 24(4):192–200.

Rothberg JM, Leamon JH, 2008. The development and impact of 454 sequencing. Nat Biotechnol, 26(10):1117–1124.

Rusch DB, Halpern AL, Sutton G, et al., 2007. The Sorcerer II Global Ocean Sampling expedition: northwest Atlantic through eastern tropical Pacific. PLoS Biol, 5(3):e77.

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

Samad A, Huff EF, Cai W, Schwartz DC, 1995. Optical mapping: a novel, single-molecule approach to genomic analysis. Genome Res, 5(1):1–4.

Sanger F, Coulson AR, 1975. A rapid method for determining sequences in DNA by primed synthesis with DNA polymerase. J Mol Biol, 94(3):441–448.

Sanger F, Coulson AR, Hong GF, Hill DF, Petersen GB, 1982. Nucleotide sequence of bacteriophage lambda DNA. J Mol Biol, 162(4):729–773.

Sanger F, Nicklen S, Coulson AR, 1977. DNA sequencing with chain-terminating inhibitors. Proc Natl Acad Sci U S A, 74(12):5463–5467.

Scheibye-Alsing K, Hoffmann S, Frankel A, et al., 2009. Sequence assembly. Computational Biology and Chemistry, 33(2):121–136.

Schuster SC, 2008. Next-generation sequencing transforms today's biology. Nat Methods, 5(1):16–18.

Selengut JD, Haft DH, Davidsen T, Ganapathy A, Gwinn-Giglio M, Nelson WC, Richter AR, White O, 2007. TIGRFAMs and Genome Properties: tools for the assignment of molecular function and biological process in prokaryotic genomes. Nucleic Acids Res, 35(Database issue):D260–4.

Seshadri R, Kravitz SA, Smarr L, Gilna P, Frazier M, 2007. CAMERA: a community resource for metagenomics. PLoS Biol, 5(3):e75.

Shendure J, Ji H, 2008. Next-generation DNA sequencing. Nat Biotechnol, 26(10):1135–1145.

Shendure J, Porreca GJ, Reppas NB, Lin X, McCutcheon JP, Rosenbaum AM, Wang MD, Zhang K, Mitra RD, Church GM, 2005. Accurate multiplex polony sequencing of an evolved bacterial genome. Science, 309(5741):1728–32.

Smit A, Hubley R, Green P, 1996-2004. RepeatMasker Open-3.0.

Smith HO, Wilcox KW, 1970. A restriction enzyme from Hemophilus influenzae. I. Purification and general properties. J Mol Biol, 51(2):379–391.

Smith LM, Sanders JZ, Kaiser RJ, Hughes P, Dodd C, Connell CR, Heiner C, Kent SB, Hood LE, 1986. Fluorescence detection in automated DNA sequence analysis. Nature, 321(6071):674–679.

Smith TF, Waterman MS, 1981. Identification of common molecular subsequences. J Mol Biol, 147(1):195–7.

Snyder LAS, Loman N, Pallen MJ, Penn CW, 2009. Next-generation sequencing–the promise and perils of charting the great microbial unknown. Microb Ecol, 57(1):1–3.

Sorek R, Zhu Y, Creevey CJ, Francino MP, Bork P, Rubin EM, 2007. Genome-wide experimental determination of barriers to horizontal gene transfer. Science, 318(5855):1449–52.

Szczepanowski R, Bekel T, Goesmann A, Krause L, Kromeke H, Kaiser O, Eichler W, Puhler A, Schluter A, 2008. Insight into the plasmid metagenome of wastewater treatment plant bacteria showing reduced susceptibility to antimicrobial drugs analysed by the 454-pyrosequencing technology. J Biotechnol, 136(1-2):54–64.

Tatusov RL, Fedorova ND, Jackson JD, et al., 2003. The COG database: an updated version includes eukaryotes. BMC Bioinformatics, 4:41.

Tatusov RL, Koonin EV, Lipman DJ, 1997. A genomic perspective on protein families. Science, 278(5338):631–637.

Teeling H, Waldmann J, Lombardot T, Bauer M, Glšckner FO, 2004. TETRA: a web-service and a stand-alone program for the analysis and comparison of tetranucleotide usage patterns in DNA sequences. BMC Bioinformatics, 5:163.

Trapnell C, Salzberg SL, 2009. How to map billions of short reads onto genomes. Nat Biotech, 27(5):455–457.

Tringe SG, von Mering C, Kobayashi A, et al., 2005. Comparative Metagenomics of Microbial Communities. Science, 308:554–557.

Tringe SG, Zhang T, Liu X, et al., 2008. The airborne metagenome in an indoor urban environment. PLoS ONE, 3(4):e1862.

Turcatti G, Romieu A, Fedurco M, Tairi AP, 2008. A new class of cleavable fluorescent nucleotides: synthesis and optimization as reversible terminators for DNA sequencing by synthesis. Nucleic Acids Res, 36(4):e25.

Turnbaugh PJ, Ley RE, Hamady M, Fraser-Liggett CM, Knight R, Gordon JI, 2007. The human microbiome project. Nature, 449(7164):804–810.

Turnbaugh PJ, Ley RE, Mahowald MA, Magrini V, Mardis ER, Gordon JI, 2006. An obesity-associated gut microbiome with increased capacity for energy harvest. Nature, 444(7122):1027–1031.

Tyson GW, Chapman J, Hugenholtz P, Allen EE, Ram RJ, Richardson PM, Solovyev VV, Rubin EM, Rokhsar DS, Banfield JF, 2004. Community structure and metabolism through reconstruction of microbial genomes from the environment. Nature, 428(6978):37–43.

Urich T, Lanzen A, Qi J, Huson DH, Schleper C, Schuster SC, 2008. Simultaneous assessment of soil microbial community structure and function through analysis of the meta-transcriptome. PLoS ONE, 3(6):e2527.

Valdivia-Granda W, 2008. The next meta-challenge for Bioinformatics. Bioinformation, 2(8):358–362.

Venter JC, Adams MD, Myers EW, et al., 2001. The sequence of the human genome. Science, 291(5507):1304–1351.

Venter JC, Remington K, Heidelberg JF, et al., 2004. Environmental Genome Shotgun Sequencing of the Sargasso Sea. Science, 304(5667):66–74.

von Mering C, Jensen LJ, Snel B, Hooper SD, Krupp M, Foglierini M, Jouffre N, Huynen MA, Bork P, 2005. STRING: known and predicted protein-protein associations, integrated and transferred across organisms. Nucleic Acids Res, 33(Database issue):D433–D437.

Wang J, Wang W, Li R, et al., 2008. The diploid genome sequence of an Asian individual. Nature, 456(7218):60–5.

Warnecke F, Luginbühl P, Ivanova N, et al., 2007. Metagenomic and functional analysis of hindgut microbiota of a wood-feeding higher termite. Nature, 450(7169):560–5.

Watson JD, Crick FH, 1953. Molecular structure of nucleic acids; a structure for deoxyribose nucleic acid. Nature, 171(4356):737–738.

Wheeler DA, Srinivasan M, Egholm M, et al., 2008a. The complete genome of an individual by massively parallel DNA sequencing. Nature, 452(7189):872–876.

Wheeler DL, Barrett T, Benson DA, et al., 2008b. Database resources of the National Center for Biotechnology Information. Nucleic Acids Res, 36(Database issue):D13–D21.

Whitman WB, Coleman DC, Wiebe WJ, 1998. Prokaryotes: the unseen majority. Proc Natl Acad Sci U S A, 95(12):6578–83.

Williamson SJ, Rusch DB, Yooseph S, et al., 2008. The Sorcerer II Global Ocean Sampling Expedition: metagenomic characterization of viruses within aquatic microbial samples. PLoS One, 3(1):e1456.

Woese CR, 1987. Bacterial evolution. Microbiol Rev, 51(2):221–71.

Wu R, Kaiser AD, 1968. Structure and base sequence in the cohesive ends of bacteriophage lambda DNA. J Mol Biol, 35(3):523–537.

Wu R, Taylor E, 1971. Nucleotide sequence analysis of DNA. II. Complete nucleotide sequence of the cohesive ends of bacteriophage lambda DNA. J Mol Biol, 57(3):491–511.

Ye Y, Doak TG, 2009. A parsimony approach to biological pathway reconstruction/inference for genomes and metagenomes. PLoS Comput Biol, 5(8):e1000465.

Yooseph S, Sutton G, Rusch DB, et al., 2007. The Sorcerer II Global Ocean Sampling expedition: expanding the universe of protein families. PLoS Biol, 5(3):e16.

Yule GU, 1925. A mathematical theory of evolution, based on the conclusions of Dr. J.C. Willis. Philosophical Transactions of the Royal Society of London Ser. B, Biol. Sci., 213:21–87.

Zagordi O, Geyrhofer L, Roth V, Beerenwinkel N, 2009. Deep Sequencing of a Genetically Heterogeneous Sample: Local Haplotype Reconstruction and Read Error Correction, Springer Berlin / Heidelberg, volume 5541 of *Lecture Notes in Computer Science*, pp. 271–284.

Zhao F, Qi J, Richter D, Buboltz A, Huson D, Schuster S, 2009. Metagenomic analysis of the microbiome associated with the hairs of extinct woolly mammoths. submitted to PNAS.

Lebens- und Bildungsweg

Name:	Daniel Christian Richter
Geburtsdatum und -ort:	13.12.1978 in Mülheim an der Ruhr

1985 - 1989	Grundschule Hölterstrasse in Mülheim an der Ruhr
1989 - 1998	Gymnasiums An den Buchen in Mülheim an der Ruhr
06/1998	Abitur (Note: 1,9)
	Leistungskurse: Englisch und Biologie
07/1998 - 07/1999	Zivildienst: Dt. Rotes Kreuz, Mülheim an der Ruhr
10/1999 - 03/2005	Studium der Bioinformatik an der Eberhard-Karls-Universität Tübingen
03/1999 - 09/1999	Diplomarbeit (Betreuer: Prof. Dr. Huson) mit dem Titel *Optimal Syntenic Layout Assembly of Two Related Genomes.*
03/2005	Diplom in Bioinformatik
	(Note: Sehr gut)
seit 04/2005	Promotion an der Fakultät für Informatik, Universität Tübingen, Arbeitsbereich *Algorithmen der Bioinformatik* bei Prof. Dr. Huson