Naiara Rodríguez-Ezpeleta
Michael Hackenberg
Ana M. Aransay *Editors*

Bioinformatics for High Throughput Sequencing

Springer

Bioinformatics for High Throughput Sequencing

Naiara Rodríguez-Ezpeleta · Michael Hackenberg
Ana M. Aransay
Editors

Bioinformatics for High Throughput Sequencing

Springer

Editors
Naiara Rodríguez-Ezpeleta
Genome Analysis Platform
CIC bioGUNE
Derio, Bizkaia, Spain
nrodriguez@cicbiogune.es

Ana M. Aransay
Genome Analysis Platform
CIC bioGUNE
Derio, Bizkaia, Spain
amaransay@cicbiogune.es

Michael Hackenberg
Computational Genomics
and Bioinformatics Group
Genetics Department & Biomedical
Research Center (CIBM)
University of Granada, Spain
mlhack@gmail.com

ISBN 978-1-4614-0781-2 e-ISBN 978-1-4614-0782-9
DOI 10.1007/978-1-4614-0782-9
Springer New York Dordrecht Heidelberg London

Library of Congress Control Number: 2011937571

Printed on acid-free paper

Springer is part of Springer Science+Business Media (www.springer.com)

Preface

The purpose of this book is to collect in a single volume the essentials of high throughput sequencing data analysis. These new technologies allow performing, at an unprecedented low cost and high speed, a panoply of experiments spanning the sequencing of whole genomes or transcriptomes, the profiling of DNA methylation, and the detection of protein–DNA interaction sites, among others. In each experiment a massive amount of sequence information is generated, making data analysis the major challenge in high throughput sequencing-based projects. Hundreds of bioinformatics applications have been developed so far, most of them focusing on specific tasks. Indeed, numerous approaches have been proposed for each analysis step, while integrated analysis applications and protocols are generally missing. As a result, even experienced bioinformaticians struggle when they have to discern among countless possibilities to analyze their data. This, together with a lack of enough qualified personnel, reveals an urgent need to train bioinformaticians in existing approaches and to develop integrated, "from start to end" software applications to face present and future challenges in data analysis.

Given this scenario, our motivation was to assemble a book covering the aforementioned aspects. Following three fundamental introductory chapters, the core of the book focuses on the bioinformatics aspects, presenting a comprehensive review of the methods and programs existing to analyze the raw data obtained from each experiment type. In addition, the book is meant to provide insight into challenges and opportunities faced by both, biologists and bioinformaticians, during this new era of sequencing data analysis.

Given the vast range of high throughput sequencing applications, we set out to edit a book suitable for readers from different research areas, academic backgrounds and degrees of acquaintance with this new technology. At the same time, we expect the book to be equally useful to researchers involved in the different steps of a high throughput sequencing project.

The "newbies" eager to learn the basics of high throughput sequencing technologies and data analysis will find what they yearn for specially by reading the first introductory chapters, but also by obviating the details and getting the rudiments of the

core chapters. On the other hand, biologists that are familiar with the fundamentals of the technology and analysis steps, but that have little bioinformatic training will find in the core chapters an invaluable resource where to learn about the different existing approaches, file formats, software, parameters, etc. for data analysis. The book will also be useful to those scientists performing downstream analyses on the output of high throughput sequencing data, as a perfect understanding of how their initial data was generated is crucial for an accurate interpretation of further outcomes. Additionally, we expect the book to be appealing to computer scientists or biologists with a strong bioinformatics background, who will hopefully find in the problematic issues and challenges raised in each chapter motivation and inspiration for the improvement of existing and the development of new tools for high throughput data analysis.

Naiara Rodríguez-Ezpeleta
Michael Hackenberg
Ana M. Aransay

Contents

Contributors

Ana M. Aransay Genome Analysis Platform, CIC bioGUNE, Parque Tecnológico de Bizkaia, Derio, Spain

Douglas W. Bryant, Jr. Department of Botany and Plant Pathology, Center for Genome Research and Biocomputing, Oregon State University, Corvallis, OR, USA

Department of Electrical Engineering and Computer Science, Oregon State University, Corvallis, OR, USA

Jason Buenrostro Division of Oncology, Department of Medicine, Stanford Genome Technology Center, Stanford University School of Medicine, Stanford, CA, USA

Yaniv Erlich Whitehead Institute for Biomedical Research, Cambridge, MA, USA

Axel Fischer Cancer Genomics Group, Department of Vertebrate Genomics, Max Planck Institute for Molecular Genetics, Berlin, Germany

Elizabeth M. Glass Mathematics and Computer Science Division, Argonne National Laboratory, Argonne, IL, USA

Computation Institute, The University of Chicago, Chicago, IL, USA

Rodrigo Goya Canada's Michael Smith Genome Sciences Centre, BC Cancer Agency, Vancouver, BC, Canada

Centre for High-Throughput Biology, University of British Columbia, Vancouver, BC, Canada

Department of Computer Science, University of British Columbia, Vancouver, BC, Canada

Michael Hackenberg Computational Genomics and Bioinformatics Group, Genetics Department, University of Granada, Granada, Spain

Hanlee P. Ji Division of Oncology, Department of Medicine, Stanford Genome Technology Center,, Stanford University School of Medicine, Stanford, CA, USA

Martin Kerick Cancer Genomics Group, Department of Vertebrate Genomics, Max Planck Institute for Molecular Genetics, Berlin, Germany

Marco A. Marra Canada's Michael Smith Genome Sciences Centre, BC Cancer Agency, Vancouver, BC, Canada

Department of Medical Genetics, University of British Columbia, Vancouver, BC, Canada

Davis J. McCarthy Bioinformatics Division, Walter and Eliza Hall Institute, Melbourne, Australia

Folker Meyer Mathematics and Computer Science Division, Argonne National Laboratory, Argonne, IL, USA

Computation Institute, The University of Chicago, Chicago, IL, USA

Institute for Genomics and Systems Biology, The University of Chicago, Chicago, IL, USA

Irmtraud M. Meyer Centre for High-Throughput Biology, University of British Columbia, Vancouver, BC, Canada

Department of Computer Science, University of British Columbia, Vancouver, BC, Canada

Department of Medical Genetics, University of British Columbia, Vancouver, BC, Canada

Todd C. Mockler Department of Botany and Plant Pathology, Center for Genome Research and Biocomputing, Oregon State University, Corvallis, OR, USA

Samuel Myllykangas Division of Oncology, Department of Medicine, Stanford Genome Technology Center, Stanford University School of Medicine, Stanford, CA, USA

Alicia Oshlack Bioinformatics Division, Walter and Eliza Hall Institute, Melbourne, Australia

School of Physics, University of Melbourne, Melbourne, Australia

Murdoch Childrens Research Institute, Parkville, Australia

Konrad Paszkiewicz School of Biosciences, University of Exeter, Exeter, UK

Paolo Ribeca Centro Nacional de Análisis Genómico, Baldiri Reixac 4, Barcelona, Spain

Mark D. Robinson Bioinformatics Division, Walter and Eliza Hall Institute, Melbourne, Australia

Department of Medical Biology, University of Melbourne, Melbourne, Australia

Epigenetics Laboratory, Cancer Research Program, Garvan Institute of Medical Research, Darlinghurst, NSW, Australia

Naiara Rodríguez-Ezpeleta Genome Analysis Platform, CIC bioGUNE, Parque Tecnológico de Bizkaia, Derio, Spain

Michal-Ruth Schweiger Cancer Genomics Group, Department of Vertebrate Genomics, Max Planck Institute for Molecular Genetics, Berlin, Germany

David Sexton Center for Human Genetics Research, Vanderbilt University, Nashville, TN, USA

Mona A. Sheikh Whitehead Institute for Biomedical Research, Cambridge, MA, USA

Gordon K. Smyth Bioinformatics Division, Walter and Eliza Hall Institute, Melbourne, Australia

Department of Mathematics and Statistics, University of Melbourne, Melbourne, Australia

David J. Studholme School of Biosciences, University of Exeter, Exeter, UK

Geetu Tuteja Department of Developmental Biology, Stanford University, Stanford, CA, USA

Matthew J. Wakefield Bioinformatics Division, Walter and Eliza Hall Institute, Melbourne, Australia

Department of Zoology, University of Melbourne, Melbourne, Australia

Matthew D. Young Bioinformatics Division, Walter and Eliza Hall Institute, Melbourne, Australia

Michael Q. Zhang Department of Molecular and Cell Biology, Center for Systems Biology, The University of Texas at Dallas, Richardson, TX, USA

Bioinformatics Division, TNLIST, Tsinghua University, Beijing, China

Chapter 1
Introduction

Naiara Rodríguez-Ezpeleta and Ana M. Aransay

Abstract Thirty-five years have elapsed since the development of modern DNA sequencing till today's apogee of high-throughput sequencing. During that time, starting from the sequencing of the first small phage genome (5,386 bases length) and going towards the sequencing of 1,000 human genomes (three billion bases length each), massive amounts of data from thousands of species have been generated and are available in public repositories. This is mostly due to the development of a new generation of sequencing instruments a few years ago. With the advent of this data, new bioinformatics challenges arose and work needs to be done in order to teach biologist swimming in this ocean of sequences so they get safely into port.

1.1 History of Genome Sequencing Technologies

1.1.1 Sanger Sequencing and the Beginning of Bioinformatics

The history of modern genome sequencing technologies starts in 1977, when Sanger and collaborators introduced the "dideoxy method" (Sanger et al. 1977), whose underlying concept was to use nucleotide analogs to cause base-specific termination of primed DNA synthesis. When dideoxy reactions of each of the four nucleotides were electrophoresed in adjacent lanes, it was possible to visually decode the corresponding base at each position of the read. From the beginning, this method allowed to read sequences of about 100 bases length, which was latter increased to 400. By the late 1980s, the amount of sequence data obtained by a single person in a day went up to 30 kb (Hutchison 2007). Although seemingly ridiculous compared

N. Rodríguez-Ezpeleta (✉) • A.M. Aransay
Genome Analysis Platform, CIC bioGUNE, Parque Tecnológico de Bizkaia,
Building 502, Floor 0, 48160 Derio, Spain
e-mail: nrodriguez@cicbiogune.es; amaransay@cicbiogune.es

N. Rodríguez-Ezpeleta et al. (eds.), *Bioinformatics for High Throughput Sequencing*,
DOI 10.1007/978-1-4614-0782-9_1,

to the amount of sequence data we deal with today, already at this scale data analysis and processing represented an issue. Computer programs were needed in order to gather the small sequence chunks into a complete sequence, to allow editing of the assembled sequence, to search for restriction sites, or to translate sequences into all reading frames. It was during this "beginning of bioinformatics" that the first suite of computer programs applied to biology was developed by Roger Staden. With the Staden package (Staden 1977), still in use today (Staden et al. 2000; Bonfield and Whitwham 2010), a widely used file formats (Dear and Staden 1992) and ideas, such as the use of base quality scores to estimate accurate consensus sequences (Bonfield and Staden 1995), were already advanced.

As the amount of sequence data increased, the need for a data repository became evident. In 1982, GenBank was created by the National Institute of Health (NIH) to provide "timely, centralized, accessible repository for genetic sequences" (Bilofsky et al. 1986), and 1 year later, more than 2,000 sequences were already stored in this database. Rapidly, tools for comparing and aligning sequences were developed. Some spread fast and are still in use today, such as FASTA (Pearson and Lipman 1988) and BLAST (Altschul et al. 1990). Even during those early times, it became already clear that bioinformatics is central to the analysis of sequence data and to the generation of hypothesis and resolving of biological questions.

1.1.2 Automated Sequencing

In 1986, Applied Biosystems (ABI) introduced automatic DNA sequencing for which different fluorescently end-labelled primers were used in each of the four dideoxy sequencing reactions. When combined in a single electrophoresis gel, the sequence could be deduced by measuring the characteristic fluorescence spectrum of each of the four bases. Computer programs were developed that automatically converted fluorescence data into a sequence without needing to autoradiography the sequencing gel and manually decode the bands (Smith et al. 1986). Compared to manual sequencing, the automation allowed the integration of data analysis into the process so that problems at each step could be detected and corrected as they appeared (Hutchison 2007).

Very shortly after the introduction of automatic sequencing, the first sequencing facility with six automated sequencers was set up at the NIH by Craig Venter and colleagues, which was expanded to 30 sequencers in 1992 at The Institute for Genomic Research (TIGR). One year later, one of today's most important sequencing centres, the Wellcome Trust Sanger Institute, was established. Among the earliest achievements of automated sequencing was the reporting of 337 new and 48 homolog-bearing human genes via the expressed sequence tag (EST) approach (Adams et al. 1991), which allows to selectively sequence fragments of gene transcripts. Using this approach, fragments of more than 87,000 human transcripts were sequenced shortly after, and today over 70 million ESTs from over 2,200 different organisms are available in dbEST (Boguski et al. 1993). In 1996, DNA sequencing

became truly automated with the introduction of the first commercial DNA sequencer that used capillary electrophoresis (the ABI Prism 310), which replaced manual pouring and loading gels with automated reloading of the capillaries from 96-well plates.

1.1.3 From Single Genes to Complete Genomes: Assemblers as Critical Factors

It was not until 1995 that the first cellular genomes, the ones of *Haemophilus influenzae* (Fleischmann et al. 1995) and of *Mycoplasma genitalium* (Fraser et al. 1995), were sequenced at TIGR. This was made possible thanks to the previously introduced whole genome shotgun (WGS) method, in which genomic DNA is randomly sheared, cloned and sequenced. In order to produce a complete genome, results needed to be assembled by a computer program, revealing assemblers as critical factors in the application of shotgun sequencing to cellular genomes. Originally, most large-scale DNA sequencing centres developed their own software for assembling the sequences that they produced; for example, the TIGR assembler (Sutton et al. 1995) was used to assemble the aforementioned two genomes. However, this later changed as the software grew more complex and as the number of sequencing centres increased. Genome assembly is a very difficult computational problem, made even more difficult in most eukaryotic genomes because many of them contain large numbers of identical sequences, known as repeats. These repeats can be thousands of nucleotides long, and some occur at thousands of different positions, especially in the large genomes of plants and animals. Thus, when more complex genomes such as the ones of the yeast *Saccharomyces cerivisiae* (Goffeau et al. 1996), the nematode *Caenorhabditis elegans* (The *C. elegans*_Sequencing_Consortium 1998) or the fruit fly *Drosophila melanogaster* (Adams et al. 2000) were envisaged, the importance of computer programs that were able to assemble thousands of reads into contigs became, if possible, even more evident. Besides repeats, these assemblers needed to be able to handle thousands of sequence reads and to deal with errors generated by the sequencing instrument.

1.1.4 The Human Genome: The Culmination of Automated Sequencing

The establishment of sequencing centres with hundreds of sequencing instruments and fully equipped with laboratory-automated procedures had as one of its ultimate goal the deciphering of the human genome. The Human Genome sequencing project formally began in 1990 when $3 billion were awarded by the United States Department of Energy and the NIH for this aim. The publicly funded effort became

an international collaboration between a number of sequencing centres in the United States, United Kingdom, France, Germany, China, India and Japan, and the whole project was expected to take 15 years. Parallel and in direct competition, Celera Genomics (founded by Applera Corporation and Craig Venter in May 1998) started its own sequencing of the human genome using WGS. Due to widespread international cooperation and advances in the field of genomics (especially in sequence analysis) as well as major advances in computing technology, a "rough draft" of the genome was finished by 2000 and the Celera and the public human genomes were published the same week (Lander et al. 2001; Venter et al. 2001). The sequencing of the human genome made bioinformatics stepping up a notch because of the considerable investment needed in software development for assembly, annotation and visualization (Guigo et al. 2000; Huson et al. 2001; Kent et al. 2002). And not only that: the complete sequence of the human genome was just the beginning of a series of more in-depth comparative studies that also required specific computing infrastructures and software implementation.

1.2 Birth of a New Generation of Sequencing Technologies

The above-described landscape has drastically changed in the past few years with the advent of new high-throughput technologies, which have noticeably reduced the per-base sequencing cost, while at the same time significantly increasing the number of bases sequenced (Mardis 2008; Schuster 2008). In 2005, Roche introduced the 454 pyrosequencer, which could easily generate more data than 50 capillary sequencers at about one sixth of the cost (Margulies et al. 2005). This was followed by the release of the Solexa Genome Analyzer by Illumina in 2006, which used sequencing by synthesis to generate tens of millions of 32 bp reads, and of the SOLiD and Heliscope platforms by Applied Biosystems and Helicos, respectively, in 2007. Today, updated instruments with increased sequencing capacity are available from all platforms, and new companies have emerged that have introduced new sequencing technologies (Pennisi 2010). The output read length depends on the technology and the specific biological application, but generally ranges from 36 to 400 bp. A detailed review of the chemistries behind each of these methods is described in Chap. 2.

These new generation of high-throughput sequencers, which combine innovations in sequencing chemistry and in detecting strand synthesis via microscopic imaging in real time, raised the amount of data obtained by a single instrument on a single day raise to 40 Gb (Kahn 2011). This means that what was previously carried out in 10 years by big consortiums involving several sequencing centres bearing each tens of sequencing instruments can now be done in a few days by a single investigator: a total revolution for genomic science. Together with the throughput increase, these new technologies have also increased the spectrum of applications of DNA sequencing to span a wide variety of research areas such as epidemiology, experimental evolution, social evolution, palaeogenetics, population genetics, phylogenetics or biodiversity (Rokas and Abbot 2009). In some cases, sequencing has replaced traditional

approaches such as microarrays, furthermore offering finer outcomes. A review of each of the applications of high-throughput sequencing in the context of specific research areas is presented in Chap. 3.

This new hoping and visibly positive scenario does not come without drawbacks. Indeed, the new spectrum of applications together with the fact that this massive amount of data comes in the form of short reads appeals for a heavy investment in the development of computational methods that can analyse the resulting datasets to infer biological meaning and to make sense of it all. This book focuses, among others, on the new bioinformatic challenges that come together with the generation of this massive amount of sequence data.

1.3 High-Throughput Sequencing Brings New Bioinformatic Challenges

1.3.1 Specialized Requirements

Compared to previous eras in genome sequencing history in which data generation was the limiting factor, the challenge now is not the data generation, but the storage, handling and analysis of the information obtained, requiring specialized bioinformatics facilities and knowledge. Indeed, as numerous experts argue, data analysis, not sequencing, will now be the main expense hurdle to many sequencing projects (Pennisi 2011). The first thing to worry about is the infrastructure needed. Sequencing datasets can range from occupying a few to hundreds of gigabytes per sample, implying high requirement of disk storage, memory and computing power for the downstream analyses, and often needing supercomputing centres or cluster facilities. Another option, if one lacks proper infrastructure, is to use cloud computing (e.g. the Elastic Compute Cloud from Amazon), which allow scientists to virtually rent both, storage and processing power, by accessing servers as they need them. However, this requires moving data from researchers to "the cloud" back and forth, which, given file sizes, is not trivial (Baker 2010). Once the data obtained and the appropriate infrastructure set, there is still an important gap to be filled: that of the bioinformaticists that will do the analysis. As mentioned in some recent reviews, there is a worry that there won't be enough people to analyse the large amounts of data generated, and bioinformaticists seem to be in short supply everywhere (Pennisi 2011). These and other related issues are presented in more detail in Chap. 4.

1.3.2 New Applications, New Challenges

The usual concern when it comes to high-throughput data analysis is that there is not such "Swiss army knife"-type software that covers all possible biological questions and combinations of experiment designs and data types. Therefore, the users have to carefully document themselves about the analysis steps required for a given

application, which often involves choosing among tens of available software for each step. Moreover, most programs come with a particular and often extensive set of parameters whose adequate application strongly depends on factors such as the experiment design, data types and biological problem studied. To make things even more complex, for some (if not for all) applications new algorithms are continuously emerging. The goal of this book is to guide the readers in their high-throughput analysis process by explaining the principles behind existing applications, methods and programs so that they can extract the maximum information from their data.

1.4 High-Throughput Data Analysis: Basic Steps and Specific Pipelines

1.4.1 Pre-processing

A common step to every high-throughput data analysis is base calling, a process in which the raw signal of a sequencing instrument, i.e. intensity data extracted from images, is decoded into sequences and quality scores. Although often neglected because usually performed by vendor-supplied base callers, this step is crucial since the characterization of errors may strongly affect downstream analysis. More accurate base callers reduce the coverage required to reach a given precision, directly decreasing sequencing costs. Not in vain, alternative to vendors base calling strategies are being explored, whose benefits and drawbacks are described in Chap. 5. Once the sequences and quality scores obtained, the following elementary step of every analysis is either the de novo assembly of the sequences, if the reference is not known, or the alignment of the reads to a reference sequence. These issues are extensively addressed in Chaps. 6 and 7.

1.4.2 Detecting Modifications at the DNA Level

Apart from deciphering new genomes via de novo assembly, DNA re-sequencing offers the possibility to address numerous biological questions applied to a wide range of research areas. For example, if the DNA is previously immunoprecipitated or enriched for methylated regions prior to sequencing, protein binding or methylated sites can be detected. The specific methods and software required for the analysis of these and related datasets are discussed in Chaps. 8 and 9.

1.4.3 Understanding More About RNA by Sequencing DNA

High-throughput sequencing allows studying RNA at an unprecedented level. The wideliest used and most studied application is the detection of differential

expression between samples for which sequencing provides more accurate and complete results than the traditionally used microarrays. The underlying concept of this method is that the number of cDNA fragments sequenced is proportional to the expression level; thus, by applying mathematical models to the counts for each sample and region of interest, differential expression can be detected. This and other applications of transcriptome sequencing are extensively discussed in Chap. 10. MicroRNAs are now the target of many studies aiming to understand gene regulation. As discussed in Chap. 11, high-throughput sequencing allows not only to profile the expression of known microRNAs in a given organism, but also to discover new ones and to compare their expression levels. Finally, Chap. 12 discusses how, as it was possible for DNA, protein binding sites can also be identified at the RNA level by means of high-throughput sequencing.

1.4.4 Metagenomics

In studies where the aim is not to understand a single species, but to study the composition and operation of complex communities in environmental samples, high-throughput sequencing has also played an important part. Traditional analyses focussed on a single molecule such as the 16S ribosomal RNA to identify the organisms present in a community, but this, in spite of potentially missing some representatives, does not give any insights into the metabolic activities of the community. Metagenomics based on high-throughput sequencing allows for taxonomic, functional and comparative analyses, but not without posing important conceptual and computational challenges that require new bioinformatics tools and methods to address them (Mitra et al. 2011). Chapter 13 focuses on MG-RAST, a high-throughput system built to provide high-performance computing to researchers interested in analysing metagenomic data.

1.5 What is Next?

The increasing range of high-throughput sequencing applications together with the falling cost for generating vast amounts of data suggests that these technologies will generate new opportunities for software and algorithm development. What will be next then is the formation of multidisciplinary scientists with expertise in both, biological and computational sciences, and making scientists from diverse backgrounds understand each other and work as a whole. As an example, understanding the disease of a patient by using whole genome sequencing would require the assembly of a "dream team" of specialists including biologists and computer scientists, geneticists, pathologists, physicians, research nurses, genetic counsellors and IT and systems support specialists, Elaine Mardis predicts (Mardis 2010). Tackling these issues and many others dealing with the current and future states of high-throughput data

analysis, we find Chap. 14 an excellent way to conclude this book and leave the reader with the concern that there is still a long way to walk, but with the satisfaction of knowing that we are in the right track.

References

Adams, M. D., S. E. Celniker, R. A. Holt, C. A. Evans, J. Gocayne, P. Amanatides, S. E. Scherer, P. W. Li *et al.* 2000. The genome sequence of Drosophila melanogaster. Science **287**.

Adams, M. D., J. M. Kelley, J. D. Gocayne, M. Dubnick, M. H. Polymeropoulos, H. Xiao, C. R. Merril, A. Wu *et al.* 1991. Complementary DNA sequencing: expressed sequence tags and human genome project. Science **252**:1651–1656.

Altschul, S. F., W. Gish, W. Miller, E. W. Myers, and D. J. Lipman. 1990. Basic local alignment search tool. J Mol Biol **215**:403–410.

Baker, M. 2010. Next-generation sequencing: adjusting to data overload. Nature Methods **7**:495–499.

Bilofsky, H. S., C. Burks, J. W. Fickett, W. B. Goad, F. I. Lewitter, W. P. Rindone, C. D. Swindell, and C. S. Tung. 1986. The GenBank genetic sequence databank. Nucleic Acids Res **14**:1–4.

Boguski, M. S., T. M. Lowe, and C. M. Tolstoshev. 1993. dbEST – database for "expressed sequence tags". Nat Genet **4**:332–333.

Bonfield, J., and R. Staden. 1995. The application of numerical estimates of base calling accuracy to DNA sequencing projects. Nucleic Acids Res **23**:1406–1410.

Bonfield, J. K., and A. Whitwham. 2010. Gap5 – editing the billion fragment sequence assembly. Bioinformatics **26**:1699–1703.

Dear, S., and R. Staden. 1992. A standard file format for data from DNA sequencing instruments. DNA Seq **3**:107–110.

Fleischmann, R. D., M. D. Adams, O. White, R. A. Clayton, E. F. Kirkness, A. R. Kerlavage, C. J. Bult, J. F. Tomb *et al.* 1995. Whole-genome random sequencing and assembly of Haemophilus influenzae Rd. Science **269**:496–512.

Fraser, C. M., J. D. Gocayne, O. White, M. D. Adams, R. A. Clayton, R. D. Fleischmann, C. J. Bult, A. R. Kerlavage *et al.* 1995. The minimal gene complement of Mycoplasma genitalium. Science **270**:397–403.

Goffeau, A., B. G. Barrell, H. Bussey, R. W. Davis, B. Dujon, H. Feldmann, F. Galibert, J. D. Hoheisel *et al.* 1996. Life with 6000 genes. Science **274**:563–547.

Guigo, R., P. Agarwal, J. F. Abril, M. Burset, and J. W. Fickett. 2000. An assessment of gene prediction accuracy in large DNA sequences. Genome Res **10**:1631–1642.

Huson, D. H., K. Reinert, S. A. Kravitz, K. A. Remington, A. L. Delcher, I. M. Dew, M. Flanigan, A. L. Halpern *et al.* 2001. Design of a compartmentalized shotgun assembler for the human genome. Bioinformatics **17 Suppl 1**:S132–139.

Hutchison, C. I. 2007. DNA sequencing: bench to bedside and beyond. Nucleic Acids Res **35**:6227–6237.

Kahn, S. D. 2011. On the future of genomic data. Science **331**:728–729.

Kent, W. J., C. W. Sugnet, T. S. Furey, K. M. Roskin, T. H. Pringle, A. M. Zahler, and D. Haussler. 2002. The human genome browser at UCSC. Genome Res **12**:996–1006.

Lander, E. S., L. M. Linton, B. Birren, C. Nusbaum, M. C. Zody, J. Baldwin, K. Devon, K. Dewar *et al.* 2001. Initial sequencing and analysis of the human genome. Nature **409**:860–921.

Mardis, E. R. 2008. The impact of next-generation sequencing technology on genetics. Trends Genet **24**:133–141.

Mardis, E. R. 2010. The $1,000 genome, the $100,000 analysis? Genome Med **2**:84.

Margulies, M., M. Egholm, W. E. Altman, S. Attiya, J. S. Bader, L. A. Bemben, J. Berka, M. S. Braverman *et al.* 2005. Genome sequencing in microfabricated high-density picolitre reactors. Nature **437**:376–380.

Mitra, S., P. Rupek, D. C. Richter, T. Urich, J. A. Gilbert, F. Meyer, A. Wilke, and D. H. Huson. 2011. Functional analysis of metagenomes and metatranscriptomes using SEED and KEGG. BMC Bioinformatics **12 Suppl 1**:S21.

Pearson, W. R., and D. J. Lipman. 1988. Improved tools for biological sequence comparison. Proc Natl Acad Sci USA **85**:2444–2448.

Pennisi, E. 2010. Genomics. Semiconductors inspire new sequencing technologies. Science **327**: 1190.

Pennisi, E. 2011. Human genome 10th anniversary. Will computers crash genomics? Science **331**: 666–668.

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

Sanger, F., S. Nicklen, and A. R. Coulson. 1977. DNA sequencing with chain-terminating inhibitors. Proc Natl Acad Sci USA **74**:5463–5467.

Schuster, S. C. 2008. Next-generation sequencing transforms today's biology. Nat Methods **5**:16–18.

Smith, L. M., J. Z. Sanders, R. J. Kaiser, P. Hughes, C. Dodd, C. R. Connell, C. Heiner, S. B. Kent *et al.* 1986. Fluorescence detection in automated DNA sequence analysis. Nature **321**:674–679.

Staden, R. 1977. Sequence data handling by computer. Nucleic Acids Res **4**:4037–4051.

Staden, R., K. F. Beal, and J. K. Bonfield. 2000. The Staden package, 1998. Methods Mol Biol **132**:115–130.

Sutton, G., O. White, M. D. Adams, and A. R. Kerlavage. 1995. TIGR Assembler: A new tool for assembling large shotgun sequencing projects. Genome Science and Technology **1**:9–19.

The_C.elegans_Sequencing_Consortium. 1998. Genome sequence of the nematode *C. elegans*: a platform for investigating biology. Science **282**:2012–2018.

Venter, J. C., M. D. Adams, E. W. Myers, P. W. Li, R. J. Mural, G. G. Sutton, H. O. Smith, M. Yandell *et al.* 2001. The sequence of the human genome. Science **291**:1304–1351.

Chapter 2
Overview of Sequencing Technology Platforms

Samuel Myllykangas, Jason Buenrostro, and Hanlee P. Ji

Abstract The high-throughput DNA sequencing technologies are based on immobilization of the DNA samples onto a solid support, cyclic sequencing reactions using automated fluidics devices, and detection of molecular events by imaging. Featured sequencing technologies include: GS FLX by 454 Life Technologies/Roche, Genome Analyzer by Solexa/Illumina, SOLiD by Applied Biosystems, CGA Platform by Complete Genomics, and PacBio RS by Pacific Biosciences. In addition, emerging technologies are discussed.

2.1 Introduction

High-throughput sequencing has begun to revolutionize science and healthcare by allowing users to acquire genome-wide data using massively parallel sequencing approaches. During its short existence, the high-throughput sequencing field has witnessed the rise of many technologies capable of massive genomic analysis. Despite the technological dynamism, there are general principles employed in the construction of the high-throughput sequencing instruments.

Commercial high-throughput sequencing platforms share three critical steps: DNA sample preparation, immobilization, and sequencing (Fig. 2.1). Generally, preparation of a DNA sample for sequencing involves the addition of defined sequences, known as "adapters," to the ends of randomly fragmented DNA (Fig. 2.2). This DNA preparation with common or universal nucleic acid ends is commonly referred to as the "sequencing library." The addition of adapters is required to anchor the DNA fragments of the sequencing library to a solid surface and define the site in

S. Myllykangas • J. Buenrostro • H.P. Ji (✉)
Division of Oncology, Department of Medicine, Stanford Genome Technology Center, Stanford University School of Medicine, CCSR, 269 Campus Drive, 94305 Stanford, CA, USA
e-mail: smyllyka@stanford.edu; jdbuenrostro@gmail.com; genomics_ji@stanford.edu

N. Rodríguez-Ezpeleta et al. (eds.), *Bioinformatics for High Throughput Sequencing*,
DOI 10.1007/978-1-4614-0782-9_2,

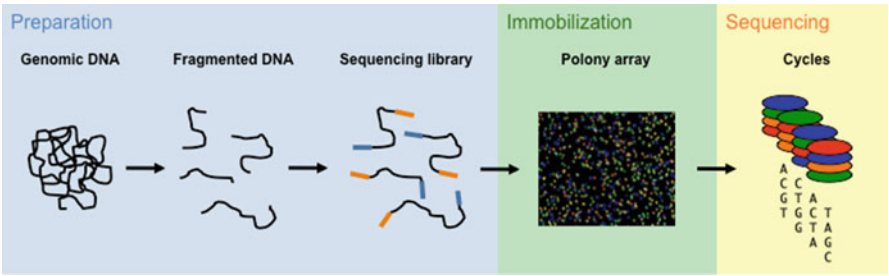

Fig. 2.1 High-throughput sequencing workflow. There are three main steps in high-throughput sequencing: preparation, immobilization, and sequencing. Preparation of the sample for high-throughput sequencing involves random fragmentation of the genomic DNA and addition of adapter sequences to the ends of the fragments. The prepared sequencing library fragments are then immobilized on a solid support to form detectable sequencing features. Finally, massively parallel cyclic sequencing reactions are performed to interrogate the nucleotide sequence

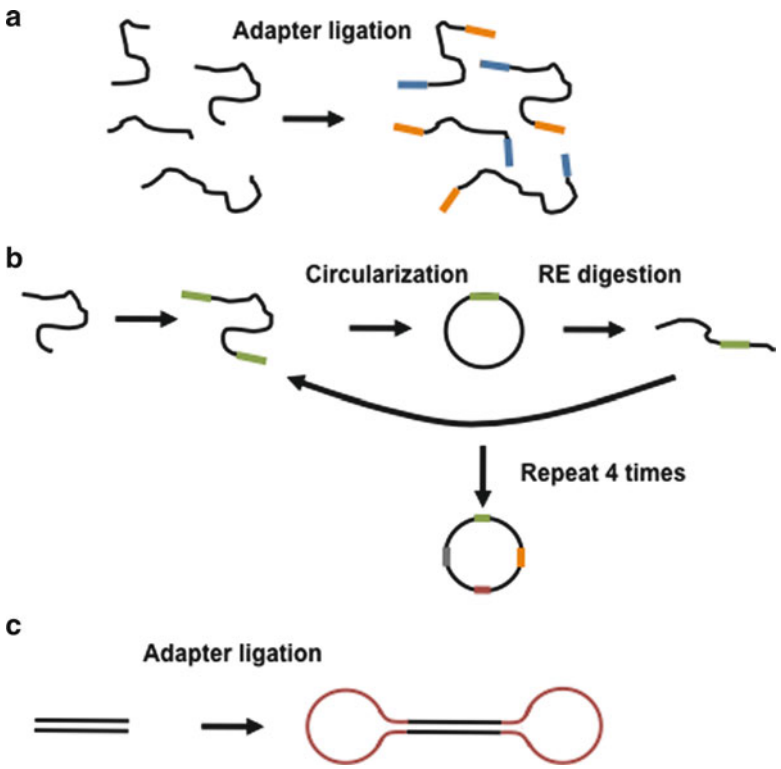

Fig. 2.2 Sequencing library preparation. There are three principal approaches for addition of adapter sequences and preparation of the sequencing library. (**a**) Linear adapters are applied in the GS FLX, Genome Analyzer, and SOLiD systems. Specific adaptor sequences are added to both ends of the genomic DNA fragments. (**b**) Circular adapters are applied in the CGA platform, where four distinct adaptor sequences are internalized into a circular template DNA. (**c**) Bubble adapters are used in the PacBio RS sequencing system. Hairpin forming bubble adapters are added to double-strand DNA fragments to generate a circular molecule

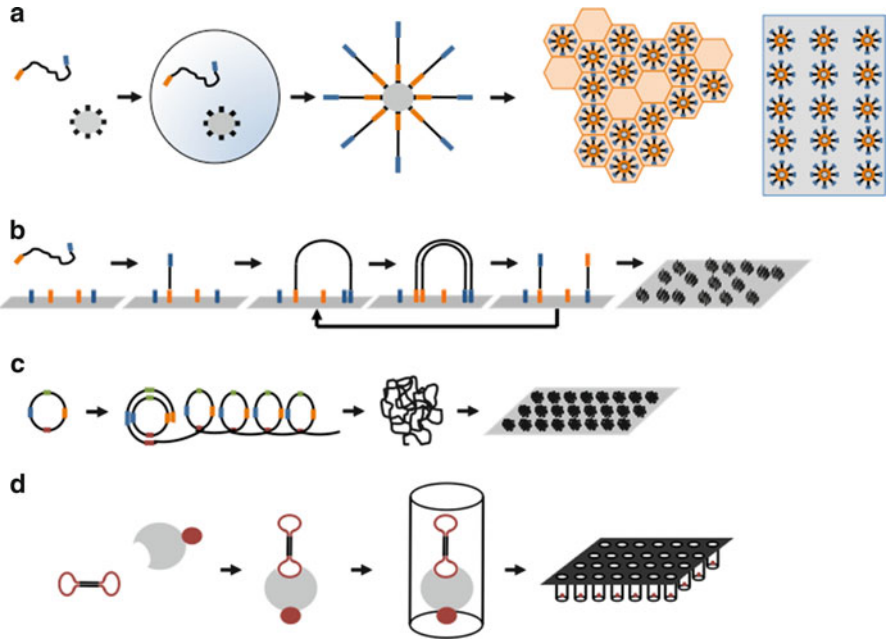

Fig. 2.3 Generation of sequencing features. High-throughput sequencing systems have taken different approaches in the generation of the detectable sequencing features. (**a**) Emulsion PCR is applied in the GS FLX and SOLiD systems. Single enrichment bead and sequencing library fragment are emulsified inside an aqueous reaction bubble. PCR is then applied to populate the surface of the bead by clonal copies of the template. Beads with immobilized clonal DNA collections are deposited onto a Picotiter plate (GS FLX) or on a glass slide (SOLiD). (**b**) Bridge-PCR is used to generate the in situ clusters of amplified sequencing library fragments on a solid support. Immobilized amplification primers are used in the process. (**c**) Rolling circle amplification is used to generate long stretches of DNA that fold into nanoballs that are arrayed in the CGA technology. (**d**) Biotinylated DNA polymerase binds to bubble adapted template in the PacBio RS system. Polymerase/template complex is immobilized on the bottom of a zero mode wave guide (ZMW)

which the sequencing reactions begin. These high-throughput sequencing systems, with the exception of PacBio RS, require amplification of the sequencing library DNA to form spatially distinct and detectable sequencing features (Fig. 2.3). Amplification can be performed in situ, in emulsion or in solution to generate clusters of clonal DNA copies. Sequencing is performed using either DNA polymerase synthesis for fluorescent nucleotides or the ligation of fluorescent oligonucleotides (Fig. 2.4).

The high-throughput sequencing platforms integrate a variety of fluidic and optic technologies to perform and monitor the molecular sequencing reactions. The fluidics systems that enable the parallelization of the sequencing reaction form the core of the high-throughput sequencing platform. Micro-liter scale fluidic devices support the DNA immobilization and sequencing using automated liquid dispensing mechanisms. These instruments enable the automated flow of reagents onto the immobilized

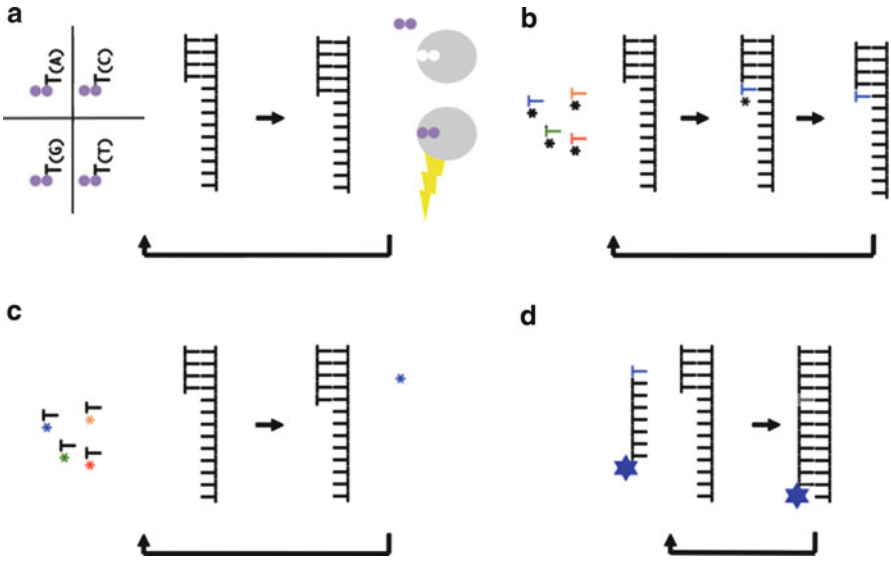

Fig. 2.4 Cyclic sequencing reactions. (**a**) Pyrosequencing is based on recording light bursts during nucleotide incorporation events. Each nucleotide is interrogated individually. Pyrosequencing is a technique used in GS FLX sequencing. (**b**) Reversible terminator nucleotides are used in the Genome Analyzer system. Each nucleotide has a specific fluorescent label and a termination moiety that prevents addition of other reporter nucleotides to the synthesized strand. All four nucleotides are analyzed in parallel and one position is sequenced at each cycle. (**c**) Nucleotides with cleavable fluorophores are used n the PacBio RS system. Each nucleotide has a specific fluorophore, which gets cleaved during the incorporation event. (**d**) Sequencing by ligation is applied in the SOLiD and CGA platforms. Although they have different approaches, the general principle is the same. Both systems apply fluorophore-labeled degenerate oligonucleotides that correspond to a specific base in the molecule

DNA samples for cyclic interrogation of the nucleotide sequence. Massive parallel sequencing systems apply high-throughput optical systems to capture information about the molecular events, which define the sequencing reaction and the sequence of the immobilized sequencing library. Each sequencing cycle consists of incorporating a detectable nucleic acid substrate to the immobilized template, washes, and imaging the molecular event. Incorporation–washing–imaging cycles are repeated to build the DNA sequence read. PacBio RS is based on monitoring DNA polymerization reactions in parallel by recording the light pulses emitted during each incorporation event in real time.

High-throughput DNA sequencing has been commercialized by a number of companies (Table 2.1). The GS FLX sequencing system (Margulies et al. 2005), originally developed by 454 Life Sciences and later acquired by Roche (Basel, Switzerland), was the first commercially available high-throughput sequencing platform. The first short read sequencing technology, Genome Analyzer, was developed by Solexa, which was later acquired by Illumina Inc. (San Diego, CA) (Bentley et al. 2008; Bentley 2006). The SOLiD sequencing system by Applied Biosystems

Table 2.1 High-throughput sequencing platforms

Platform	Company	Sequencing library	Support	Feature generation	Sequencing reaction	Detection method
GS FLX	454 Life Sciences, Roche	Linear adapters	Picotiter plate	Emulsion PCR	Synthesis	Pyrosequencing
Genome analyzer	Solexa, Illumina	Linear adapters	Flow cell	Bridge PCR	Synthesis	Fluorophore labeled reversible terminator nucleotides
SOLiD	Applied Biosystems	Linear adapters	Flow cell	Emulsion PCR	Ligation	Fluorophore labeled oligonucleotide probes
CGA platform	Complete genomics	Circular adapters	DNA nanoball arrays	Rolling circle amplification	Ligation	Fluorophore labeled oligonucleotide probes
PacBio RS	Pacific biosciences	Bubble adapters	Zero mode waveguide	Single molecule	Synthesis, real-time	Phospholinked fluorophore labeled nucleotides

(Foster City, CA) applies fluorophore labeled oligonucleotide panel and ligation chemistry for sequencing (Smith et al. 2010; Valouev et al. 2008). Complete Genomics (Mountain View, CA) has developed a sequencing technology called CGA that is based on preparing a semiordered array of DNA nanoballs on a solid surface (Drmanac et al. 2010). Pacific Biosciences (Menlo Park, CA) has developed PacBio RS sequencing technology, which uses the polymerase enzyme, fluorescent nucleotides, and high-content imaging to detect single-molecule DNA synthesis events in real time (Eid et al. 2009).

2.2 Genome Sequencer GS FLX

The Roche GS FLX sequencing process consists of preparing an end-modified DNA fragment library, sample immobilization on streptavidin beads, and pyrosequencing.

2.2.1 Preparation of the Sequencing Library

Sample preparation of the GS FLX sequencing system begins with random fragmentation of DNA into 300–800 base-pair (bp) fragments (Margulies et al. 2005). After shearing, fragmented double-stranded DNA is repaired with an end-repair enzyme cocktail and adenine bases are added to the 3′ ends of fragments. Common adapters, named "A" and "B," are then nick-ligated to the fragments ends. Nicks present in the adapter-to-fragment junctions are filled in using a strand-displacing *Bst* DNA polymerase. Adapter "B" carries a biotin group, which facilitates the purification of homo-adapted fragments (A/A or B/B). The biotin labeled sequencing library is captured on streptavidin beads. Fragments containing the biotin labeled B adapter are bound to the streptavidin beads while homozygous, nonbiotinylated A/A adapters are washed away. The immobilized fragments are denatured after which both strands of the B/B adapted fragments remain immobilized by the streptavidin–biotin bond and single-strand template of the A/B fragments are freed and used in sequencing.

2.2.2 Emulsion PCR and Immobilization to Picotiter Plate

In GS FLX sequencing, the single-strand sequencing library fragment is immobilized onto a specific DNA capture bead (Fig. 2.3a). GS FLX sequencing relies on capturing one DNA fragment onto a single bead. One-to-one ratio of beads and fragments is achieved by limiting dilutions. The bead-bound library is then amplified using a specific form of PCR. In emulsion PCR, parallel amplification of bead captured library fragments takes place in a mixture of oil and water. Aqueous bubbles, immersed in oil, form microscopic reaction entities for each individual capture bead. Hundreds of thousands of amplified DNA fragments can be immobilized on the surface of each bead.

In the GS FLX sequencing platform, beads covered with amplified DNA can be immobilized on a solid support (Fig. 2.3a). The GS FLX sequencing platform uses a "Picotiter plate," a solid phase support containing over a million picoliter volume wells (Margulies et al. 2005). The dimensions of the wells are such that only one bead is able to enter each position on the plate. Sequencing chemistry flows through the plate and insular sequencing reactions take place inside the wells. The Picotiter plate can be compartmentalized up to 16 separate reaction entities using different gaskets.

2.2.3 Pyrosequencing

The GS FLX sequencing reaction utilizes a process called pyrosequencing (Fig. 2.4a) to detect the base incorporation events during sequencing (Margulies et al. 2005). In pyrosequencing, Picotiter plates are flushed with nucleotides and the activity of DNA polymerase and the incorporation of a nucleotide lead to the release of a pyrophosphate. ATP sulfurylase and luciferase enzymes convert the pyrophosphate into a visible burst of light, which is detected by a CCD imaging system. Each nucleotide species (i.e., dATP, dCTP, dGTP, and dTTP) is washed over the Picotiter plate and interrogated separately for each sequencing cycle. The GS FLX technology relies on asynchronous extension chemistry, as there is no termination moiety that would prevent addition of multiple bases during one sequencing cycle. As a result, multiple nucleotides can be incorporated to the extending DNA strand and accurate sequencing through homopolymer stretches (i.e., AAA) represents a challenging technical issue for GS FLX. However, a number of improvements have been made to improve the sequencing performance of homopolymers (Smith et al. 2010).

2.3 Genome Analyzer

The Genome Analyzer system is based on immobilizing linear sequencing library fragments using solid support amplification. DNA sequencing is enabled using fluorescent reversible terminator nucleotides.

2.3.1 Sequencing Library Preparation

Sample preparation for the Illumina Inc. Genome Analyzer involves adding specific adapter sequences to the ends of DNA molecules (Fig. 2.2a) (Bentley et al. 2008; Bentley 2006). The production of a sequencing library initiates with fragmentation of the DNA sample, which defines the molecular entry points for the sequencing reads. Then, an enzyme cocktail repairs the staggered ends, after which, adenines (A) are added to the 3′ ends of the DNA fragments. A-tailed DNA is applied as a template to ligate double strand, partially complementary adapters to the DNA fragments.

Adapted DNA library is size selected and amplified to improve the quality of sequence reads. Amplification introduces end-specific PCR primers that bring in the portion of the adapter required for sample processing on the Illumina Inc. system.

2.3.2 Solid Support Amplification

Illumina Inc. flow cells are planar, fluidic devices that can be flushed with sequencing reagents. The inner surface of the flow cell is functionalized with two oligonucleotides, which creates an ultra-dense primer field. The sequencing library is immobilized on the surface of a flow cell (Fig. 2.3b). The immobilized primers on the flow cell surface have sequences that correspond to the DNA adapters present in the sequencing library. DNA molecules in the sequencing library hybridize to the immobilized primers and function as templates in strand extension reactions that generate immobilized copies of the original molecules.

In the Illumina Inc. Genome Analyzer system, the preparation of the flow cell requires amplification of individual DNA molecules of a sequencing library and formation of spatially condensed, microscopically detectable clusters of molecular copies (Fig. 2.3b). The primer functionalized flow cell surface serves as a support for amplification of the immobilized sequencing library by a process also known as "Bridge-PCR."

Generally, PCR is performed in solution and relies on repeated thermal cycles of denaturation, annealing, and extension to exponentially amplify DNA molecules. In the Illumina Inc. Genome Analyzer Bridge-PCR system, amplification is performed on a solid support using immobilized primers and in isothermal conditions using reagent flush cycles of denaturation, annealing, extension, and wash. Bridge-PCR initiates by hybridization of the immobilized sequencing library fragment and a primer to form a surface-supported molecular bridge structure. Arched molecule is a template for a DNA polymerase-based extension reaction. The resulting bridged double-strand DNA is freed using a denaturing reagent. Repeated reagent flush cycles generate groups of thousands of DNA molecules, also known as "clusters," on each flow cell lane. DNA clusters are finalized for sequencing by unbinding the complementary DNA strand to retain a single molecular species in each cluster, in a reaction called "linearization," followed by blocking the free 3′ ends of the clusters and hybridizing a sequencing primer.

2.3.3 Sequencing Using Fluorophore Labeled Reversible Terminator Nucleotides

The prepared flow cell is connected to a high-throughput imaging system, which consists of microscopic imaging, excitation lasers, and fluorescence filters. Molecularly, Illumina Inc.'s sequencing-by-synthesis method employs four distinct fluorophores and reversibly terminated nucleotides (Fig. 2.4b). The sequencing reaction initiates by

DNA polymerase synthesis of a fluorescent reversible terminator nucleotide from the hybridized sequencing primer. The extended base contains a fluorophore specific to the extended base and a reversible terminator moiety, which inhibits the incorporation of additional nucleotides.

After each incorporation reaction, the immobilized nucleotide fluorophores, corresponding to each cluster, are imaged in parallel. *X*–*Y* position of imaged nucleotide fluorophore defines the first base of a sequence read. Before proceeding to next cycle, reversible-terminator moieties and fluorophores are detached using a cleavage reagent, enabling subsequent addition of nucleotides. The synchronous extension of the sequencing strand by one nucleotide per cycle ensures that homopolymer stretches (consecutive nucleotides of the same kind, i.e., AAAA) can be accurately sequenced. However, failure to incorporate a nucleotide during a sequencing cycle results in off-phasing effect – some molecules are lagging in extension and the generalized signal derived from the cluster deteriorates over cycles. Therefore, Illumina Inc. sequencing accuracy declines as the read length increases, which limits this technology to short sequence reads.

2.4 SOLiD

The Applied Biosystems SOLiD sequencer, featured in Valouev et al. (2008) and Smith et al. (2010), is based on the Polonator technology (Shendure et al. 2005), an open source sequencer that utilizes emulsion PCR to immobilize the DNA library onto a solid support and cyclic sequencing-by-ligation chemistry.

2.4.1 Sequencing Library Preparation and Immobilization

The in vitro sequencing library preparation for SOLiD involves fragmentation of the DNA sample to an appropriate size range (400–850 bp), end repair and ligation of "P1" and "P2" DNA adapters to the ends of the library fragments (Valouev et al. 2008). Emulsion PCR is applied to immobilize the sequencing library DNA onto "P1" coated paramagnetic beads. High-density, semi-ordered polony arrays are generated by functionalizing the 3′ ends of the templates and immobilizing the modified beads to a glass slide. The glass slides can be segmented up to eight chambers to facilitate up scaling of the number of analyzed samples.

2.4.2 Sequencing by Ligation

The SOLiD sequencing chemistry is based on ligation (Fig. 2.4d). A sequencing primer is hybridized to the "P1" adapter in the immobilized beads. A pool of uniquely labeled oligonucleotides contains all possible variations of the complementary

bases for the template sequence. SOLiD technology applies partially degenerate, fluorescently labeled, DNA octamers with dinucleotide complement sequence recognition core. These detection oligonucleotides are hybridized to the template and perfectly annealing sequences are ligated to the primer. After imaging, unextended strands are capped and fluorophores are cleaved. A new cycle begins 5 bases upstream from the priming site. After the seven sequencing cycles first sequencing primer is peeled off and second primer, starting at n-1 site, is hybridized to the template. In all, 5 sequencing primers (n, n-1, n-2, n-3, and n-4) are utilized for the sequencing. As a result, the 35-base insert is sequenced twice to improve the sequencing accuracy.

Since the ligation-based method in the SOLiD system requires complex panel of labeled oligonucleotides and sequencing proceeds by off-set steps, the interpretation of the raw data requires a complicated algorithm (Valouev et al. 2008). However, the SOLiD system achieves a slightly better performance in terms of sequencing accuracy due to the redundant sequencing of each base twice by a dinucleotide detection core structure of the octamer sequencing oligonucleotides.

2.5 CGA Platform

The CGA Platform (Complete Genomics) represents the first high-throughput platform only available to the public as a service (Table 2.1). The CGA technology is based on preparation of circular DNA libraries (Fig. 2.2c) and rolling circle amplification (RCA) to generate DNA nanoballs that are arrayed on a solid support (Fig. 2.3c) (Drmanac et al. 2010).

2.5.1 Sequencing Library Preparation

DNA is randomly fragmented and 400–500 bp fragments are collected. The fragment ends are enzymatically end-repaired and dephosphorylated. Common adapters are ligated to the DNA fragments using nick translation. These adapter libraries are enriched and Uracils are incorporated in the products using PCR and uracil containing primers. Uracils are removed from the final product to create overhangs. The products are digested and methylated with *Acu*I and circularized using T4 DNA ligase in a presence of a splint oligonucleotide. The circularized products are purified using an exonuclease, which degrades residual linear DNA molecules. Linearization, adapter ligation, PCR amplification, restriction enzyme digestion, and circularization process are repeated until four unique adapters are incorporated into the circular sequencing library molecules. Prior to the final circularization step, a single-strand template is purified using strand separation by bead capture and exonuclease treatment. The final product contains two 13 base genomic DNA inserts and two 26 base genomic DNA inserts adjacent to the adapter sequences.

2.5.2 DNA Nanoball Array

To prepare the immobilized sequencing features for Complete Genomics sequencing, circular, single-strand DNA library is amplified using RCA and a highly processive and strand displacing Phi29 polymerase. RCA creates long DNA strands from the circular DNA library templates that contain short palindrome sequences. The palindrome sequences within the long linear products promote intramolecular coiling of the molecule and formation of the DNA nanoballs (DNBs). A nanoball is a long strand of repetitive fragments of amplified DNA, which forms a detectable, three-dimensional, condensed, and spherical sequencing object.

The hexamethyldisilazane (HDMS) covered surface of the CGA Platform's fluidic chamber is spotted by aminosilane using photolitography techniques. Three-hundred nm aminosilane spots cover over 95% of the CGA surface. While HDMS inhibits DNA binding, the positively charged aminosilane binds the negatively charged DNBs. Randomly organized but regionally ordered high-density array has 350 million immobilized DNBs within a distance of 1.29 μm between the centers of the spots.

2.5.3 Sequencing by Ligation Using Combinatorial Probe Anchors

Complete genomics' CGA Platform uses a novel strategy called combinatorial probe anchor ligation (cPAL) for sequencing. The process begins by hybridization between an anchor molecule and one of the unique adapters. Four degenerate 9-mer oligonucleotides are labeled with specific fluorophores that correspond to a specific nucleotide (A, C, G, or T) in the first position of the probe. Sequence determination occurs in a reaction where the correct matching probe is hybridized to a template and ligated to the anchor using T4 DNA ligase. After imaging of the ligated products, the ligated anchor-probe molecules are denatured. The process of hybridization, ligation, imaging, and denaturing is repeated five times using new sets of fluorescently labeled 9-mer probes that contain known bases at the n+1, n+2, n+3, and n+4 positions.

After five cycles, the fidelity of the ligation reaction decreases and sequencing continues by resetting the reaction using an anchor with degenerate region of 5 bases. Another five cycles of sequencing by ligation are performed using the fluorescently labeled, degenerate 9-mer probes. The cyclic sequencing of 10 bases can be repeated up to eight times, starting at each of the unique anchors, and resulting in 62–70 base long reads from each DNB.

Unlike other high-throughput sequencing platforms that involve additive detection chemistries, the cPAL technology is unchained as sequenced nucleotides are not physically linked. The anchor and probe constructs are removed after each sequencing cycle and the next cycle is initiated completely independent of the molecular

events of the previous cycle. A disadvantage of this system is that read lengths are limited by the sample preparation, even if, longer reads up to 120 bases can be achieved by adding more restriction enzyme sites.

2.6 PacBio RS

PacBio RS is a single-molecule real-time (SMRT) sequencing system developed by Pacific Biosciences (Eid et al. 2009).

2.6.1 Preparation of the Sequencing Library

SMRTbell is the default method for preparing sequencing libraries for PacBio RS in order to get high accuracy variant detection (Travers et al. 2010) (Fig. 2.3d). For genome sequencing, DNA is randomly fragmented and then end-repaired. Then, 3′ adenine is added to the fragmented genomic DNA, which facilitates ligation of an adapter with a T overhang. Single DNA oligonucleotide, which forms an intramolecular hairpin structure, is used as the adapter. The SMRTbell DNA template is structurally a linear molecule but the bubble adapters create a topologically circular molecule.

2.6.2 The SMRT Cell

The SMRT cell houses a patterned array of zero-mode waveguides (ZMWs) (Korlach et al. 2008b; Levene et al. 2003). ZMWs are nanofabricated on a glass surface. The volume of the nanometer-sized aluminum layer wells is in zeptoliter scale. The SMRT cell is prepared for polymerase immobilization by coating the surface with streptavidin. The preparation of the sequencing reaction requires incubating a biotinylated Phi29 DNA polymerase with primed SMRTbell DNA templates. The coupled products are then immobilized to the SMRT cell using a biotin–streptavidin interaction.

2.6.3 Processive DNA Sequencing by Synthesis

When the sequencing reaction begins, the tethered polymerase incorporates nucleotides with individually phospholinked fluorophores, each fluorophore corresponding to a specific base, to the growing DNA chain (Korlach et al. 2008a). During the initiation of a base incorporation event, the fluorescent nucleotide is brought into

the polymerase's active site and into proximity of the ZMW glass surface. At the bottom of the ZMV, high-resolution camera records the fluorescence of the nucleotide being incorporated. During the incorporation reaction a phosphate-coupled fluorophore is released from the nucleotide and that dissociation diminishes the fluorescent signal. While the polymerase synthesizes a copy of the template strand, incorporation events of successive nucleotides are recorded in a movie-like format.

The tethered Phi29 polymerase is a highly processive strand-displacing enzyme capable of performing RCA. Using SMRTbell libraries with small insert sizes, it is possible to sequence the template using a scheme called circular consensus sequencing. The same insert is read on the sense and antisense strands multiple times and the redundancy is dependent on insert size. This highly redundant sequencing approach improves the accuracy of the base calls overcoming the high error rates associated with real-time sequencing and allowing accurate variant detection. For low accuracy and long read lengths, larger insert sizes can be used. The unique method of detecting nucleotide incorporation events in real time allows the development of novel applications, such as the detection of methylated cytosines based on differential polymerase kinetics (Flusberg et al. 2010).

2.7 Emerging Technologies

The phenomenal success of high-throughput DNA sequencing systems has fueled the development of novel instruments that are anticipated to be faster than the current high-throughput technologies and will lower the cost of genome sequencing. These future generations of DNA sequencing are based on technologies that enable more efficient detection of sequencing events. Instruments for detection of ion release during incorporation of label-free natural nucleotides and nanopore technologies are emerging. The pace of technological development in the field of genome sequencing is overwhelming and new technological breakthroughs are probable in the near future.

2.7.1 Semiconductor Sequencing

Life Technology and Ion Torrent are developing the Ion Personal Genome Machine, which represents an affordable and rapid bench top system designed for small projects. The IPG system harbors an array of semiconductor chips capable of sensing minor changes in pH and detecting nucleotide incorporation events by the release of a hydrogen ion from natural nucleotides. The Ion Torrent system does not require any special enzymes or labeled nucleotides and takes advantage of the advances made in the semiconductor technology and component miniaturization.

2.7.2 Nanopore Sequencing

Nanopore sequencing is based on a theory that recording the current modulation of nucleic acids passing through a pore could be used to discern the sequence of individual bases within the DNA chain. Nanopore sequencing is expected to offer solutions to limitations of short read sequencing technologies and enable sequencing of large DNA molecules in minutes without having to modify or prepare samples. Despite the technology's potential many technical hurdles remain.

Exonuclease DNA sequencing from Oxford Nanopores represents a possible solution to some of the technical hurdles found in nanopore sequencing. The system seeks to couple an exonuclease to a biological alpha hemolysin pore and plant that construct onto a lipid bilayer. When the exonuclease encounters a single-strand DNA molecule, it cleaves a base and passes it through the pore. Each base creates a unique signature of current modulation as it crosses through the lipid bilayer, which can be detected using sensitive electrical methods.

2.8 Conclusions

Although high-throughput sequencing is in its infancy, it has already begun to reshape the ways in which biology is portrayed. In principle, massive parallel sequencing systems are powerful technology rigs that integrate basic molecular biology, automated fluidics devices, high-throughput microscopic imaging, and information technologies. By default, to be able to use these systems requires comprehensive understanding of the complex underlying molecular biology and biochemistry. The ultra-high-throughput instruments are essentially high-tech machines and understanding the engineering principles gives the user the ability to command and troubleshoot the massive parallel sequencing systems. The complexity and size of the experimental results is rescaling the boundaries of biological inquiry. With the advent of these technologies, it is required that users acquire computational skills and develop systematic data analysis pipelines. High-throughput sequencing has presented an introduction to an exciting new era of multidisciplinary science.

References

Bentley, D. R. 2006. Whole-genome re-sequencing. *Curr Opin Genet Dev* 16 (6):545–552. doi:S0959-437X(06)00208-5 [pii] 10.1016/j.gde.2006.10.009.

Bentley, DR, S Balasubramanian, HP Swerdlow, GP Smith, J Milton, CG Brown, KP Hall et al. 2008. Accurate whole human genome sequencing using reversible terminator chemistry. *Nature* 456:53–59.

Drmanac, R., A. B. Sparks, M. J. Callow, A. L. Halpern, N. L. Burns, B. G. Kermani, P. Carnevali et al. 2010. Human genome sequencing using unchained base reads on self-assembling DNA nanoarrays. *Science* 327 (5961):78–81. doi:1181498 [pii] 10.1126/science.1181498.

Eid, J, A Fehr, J Gray, K Luong, J Lyle, G Otto, P Peluso et al. 2009. Real-time DNA sequencing from single polymerase molecules. *Science* 323:133–138.

Flusberg, B. A., D. R. Webster, J. H. Lee, K. J. Travers, E. C. Olivares, T. A. Clark, J. Korlach, and S. W. Turner. 2010. Direct detection of DNA methylation during single-molecule, real-time sequencing. *Nat Methods* 7 (6):461–465. doi:nmeth.1459 [pii] 10.1038/nmeth.1459.

Korlach, J, A Bibillo, J Wegener, P Peluso, TT Pham, I Park, S Clark, GA Otto, and SW Turner. 2008. Long, processive enzymatic DNA synthesis using 100% dye-labeled terminal phosphate-linked nucleotides. *Nucleosides Nucleotides Nucleic Acids* 27:1072–1083.

Korlach, J., P. J. Marks, R. L. Cicero, J. J. Gray, D. L. Murphy, D. B. Roitman, T. T. Pham, G. A. Otto, M. Foquet, and S. W. Turner. 2008. Selective aluminum passivation for targeted immobilization of single DNA polymerase molecules in zero-mode waveguide nanostructures. *Proc Natl Acad Sci USA* 105 (4):1176–1181.

Levene, M. J., J. Korlach, S. W. Turner, M. Foquet, H. G. Craighead, and W. W. Webb. 2003. Zero-mode waveguides for single-molecule analysis at high concentrations. *Science* 299 (5607):682–686.

Margulies, M., M. Egholm, W. E. Altman, S. Attiya, J. S. Bader, L. A. Bemben, J. Berka et al. 2005. Genome sequencing in microfabricated high-density picolitre reactors. *Nature* 437 (7057): 376–380. doi:nature03959 [pii] 10.1038/nature03959.

Shendure, J., G. J. Porreca, N. B. Reppas, X. Lin, J. P. McCutcheon, A. M. Rosenbaum, M. D. Wang, K. Zhang, R. D. Mitra, and G. M. Church. 2005. Accurate multiplex polony sequencing of an evolved bacterial genome. *Science* 309 (5741):1728–1732. doi:1117389 [pii] 10.1126/science.1117389.

Smith, A. M., L. E. Heisler, R. P. St Onge, E. Farias-Hesson, I. M. Wallace, J. Bodeau, A. N. Harris et al. 2010. Highly-multiplexed barcode sequencing: an efficient method for parallel analysis of pooled samples. *Nucleic Acids Res* 38 (13):e142.

Travers, K. J., C. S. Chin, D. R. Rank, J. S. Eid, and S. W. Turner. 2010. A flexible and efficient template format for circular consensus sequencing and SNP detection. *Nucleic Acids Res* 38 (15):e159. doi:gkq543 [pii] 10.1093/nar/gkq543.

Valouev, A., J. Ichikawa, T. Tonthat, J. Stuart, S. Ranade, H. Peckham, K. Zeng et al. 2008. A high-resolution, nucleosome position map of C. elegans reveals a lack of universal sequence-dictated positioning. *Genome Res* 18 (7):1051–1063. doi:gr.076463.108 [pii] 10.1101/gr.076463.108.

Chapter 3
Applications of High-Throughput Sequencing

Rodrigo Goya, Irmtraud M. Meyer, and Marco A. Marra

Abstract Although different instruments for massively parallel sequencing exist, each with their own chemistry, resolution, error types, error frequencies, throughput and costs; the principle behind them is similar: to deduce an original sequence of bases by sampling many templates. The wide array of applications derives from the biological sources and methods used to manufacture the sequencing libraries and the analytic routines employed. By using DNA as source material, a whole genome can be sequenced or, through amplification methods, a more detailed reconstruction of a specific *locus* can be obtained. Transcriptomes can also be studied by capturing and sequencing different types of RNA. Other capture methods such as cross-linking followed by immunoprecipitation can be used to study DNA–protein interactions. We will explore these applications and others in the following sections and explain the different analysis strategies that are used to analyze each data type.

R. Goya
Canada's Michael Smith Genome Sciences Centre, BC Cancer Agency, Vancouver, BC, Canada

Centre for High-Throughput Biology, University of British Columbia, Vancouver, BC, Canada

Department of Computer Science, University of British Columbia, Vancouver, BC, Canada
e-mail: rgoya@bcgsc.ca

I.M. Meyer
Centre for High-Throughput Biology, University of British Columbia, Vancouver, BC, Canada

Department of Computer Science, University of British Columbia, Vancouver, BC, Canada

Department of Medical Genetics, University of British Columbia, Vancouver, BC, Canada
e-mail: irmtraud@cs.ubc.ca

M.A. Marra (✉)
Canada's Michael Smith Genome Sciences Centre, BC Cancer Agency, Vancouver, BC, Canada

Department of Medical Genetics, University of British Columbia, Vancouver, BC, Canada
e-mail: mmarra@bcgsc.ca

N. Rodríguez-Ezpeleta et al. (eds.), *Bioinformatics for High Throughput Sequencing*,
DOI 10.1007/978-1-4614-0782-9_3,

3.1 The Evolution of DNA Sequencing

For the last 30 years, DNA sequencing has been central to the study of molecular biology, having become a valuable tool in the efforts to understand the basic building blocks of living organisms. The availability of genome sequences provides researchers with the data required to map the genomic location and structure of functional elements (e.g., protein coding genes) and to enable the study of the regulatory sequences that play roles in transcriptional regulation. Large international collaborations have for some time undertaken the decoding of genome sequences for a diversity of organisms, including (but not limited to) the bacteria *Haemophilus influenzae Rd*, with a genome of 1.8 megabases (Fleischmann et al. 1995); the yeast *Saccharomyces cereviseae*, with a 12-megabase genome (Goffeau et al. 1996); the nematode *C. elegans*, with a 97-megabase genome (The *C. elegans* Sequencing Consortium 1998) and more recently the human genome, with ~3 gigabases of genomic data (Lander et al. 2001; Venter et al. 2001). Such projects have yielded data that has been used to develop molecular "parts lists" that reveal not only organismal gene content, but inform on the evolutionary relationships and pressures that have acted to shape genomes. The technology historically employed for such reference genome sequencing projects was based on Sanger chain termination sequencing. For whole genomes, the strategy included the cloning of DNA fragments, often in bacterial artificial chromosomes (BAC) or other large-insert-containing vectors for large (e.g., mammalian-sized) genomes, amplification of the templates in bacterial cells, "mapping" a redundant set of large insert clones to select an overlapping tiling set of clones for sequencing (Marra et al. 1997), preparation of sequencing libraries from individual large insert clones in the tiling set, and then Sanger sequencing and assembly of the short sequence reads into longer sequence "contigs" (Staden 1979). Although critical in the successful completion of numerous sequencing efforts, and still considered a gold standard for certain applications, Sanger sequencing's relatively low throughput and high cost can become limiting factors when designing large experiments where massively parallel data collection is required. The high-throughput capabilities of massive parallel sequencing have taken sequencing efforts in new directions not previously feasible, enabling both the analysis of new genomes and also facilitating genome comparisons across individuals from the same species, thereby identifying intraspecific variants in a high resolution genome-wide fashion.

3.1.1 Whole Genome Shotgun Sequencing

Whole genome shotgun sequencing uses genomic DNA as the source material for preparation of DNA sequencing "libraries." A library is a collection of DNA fragments, obtained from the source material and rendered suitable for sequence analysis through a process of library construction, which involves shearing of the DNA sample by chemical (e.g., restriction enzymes) or more random and, therefore,

preferable mechanical means (e.g., sonication). The aim of fragmentation is to reduce the physical size of the DNA template molecules to the optimal fragment length for the assay type and the instrument system being used, while endeavoring to maintain an unbiased representation of the starting DNA material. The resulting fragments are then subjected to gel-based electrophoretic separation, and the desired size range of DNA fragments is then recovered from the gel matrix. A uniform size distribution is especially useful when analyzing paired-end sequences, in which sequences are collected from both ends of linear template molecules. As will be explained later, paired-end information can enable certain types of bioinformatic analysis. Common goals of whole genome shotgun sequencing are alternatively (1) re-sequencing multiple individuals, for example to study intraspecific variation and the association of such variation with health and disease states, or (2) decoding a previously unsequenced genome to examine gene content and genome structure.

3.1.2 Whole Genome Re-sequencing

The term "re-sequencing" refers to the act of sequencing multiple individuals from the same species, where a reference genome has been generated and is used to assist in the interpretation of the data collected using next generation sequencing approaches. For example, re-sequencing of human genomes has been used to discover both mutations (Mardis et al. 2009; Shah et al. 2009b) and polymorphisms (The 1000 Genomes Project Consortium 2010). The existence of reference genome sequences has driven this application, which was the first one employed using Roche/454, Illumina/Genome-Analyzer, and Applied Biosystems/SOLiD technologies. Alongside the obvious scientific impetus for re-sequencing species of significance in medical research, an initial reason for the emergence of re-sequencing was largely technical – software for whole genome assembly did not exist, and so, in the absence of a reference genome to aid alignment, high-throughput sequencing was capable of little more than producing large collections of sequence reads, as opposed to extensive contigs of sequence data such as those produced using assembly of the much longer (and less numerous) Sanger sequencing reads used to produce reference genome sequences for the human (Lander et al. 2001; Venter et al. 2001), mouse (MGSC 2002), rat (Gibbs et al. 2004), and other genomes.

An early challenge in re-sequencing was the production of sequencing reads of sufficient length to align ("map") uniquely to the human genome. Using simulated data, it was estimated that reads of at least 25 nucleotides in length would be needed to uniquely cover 80% of the human genome, and reads of at least 43 bp would be required to cover 90% of the human genome (Whiteford et al. 2005). With the exception of the Roche/454 instrument, early achievement of such read lengths entailed both instrumentation and chemistry challenges. The Roche instrument was used to illustrate the potential of next generation re-sequencing when it was used to analyze Dr. James D. Watson's genome (Wheeler et al. 2008). Within a time span of 2 months, 24.5 gigabases of raw sequence data were generated for the Watson

genome, providing, after processing, 7.4-fold average base pair coverage of the genome. The sequence data provided sufficient resolution for the detection of known polymorphisms, novel mutations, insertions, deletions, and even copy number changes.

Since that landmark study, whole genome re-sequencing continues to be used actively in various projects, including for example the 1000 Genomes Project (http://www.1000genomes.org/), which aims to discover common sequence variants in healthy human populations, and also in various cancer studies (e.g., Mardis et al. 2009; Shah et al. 2009b) including those conducted under the auspices of the large TCGA (http://tcga.cancer.gov/) and ICGC (http://www.icgc.org/, 2010) consortia. Applications for whole genome re-sequencing continue to emerge, and the steady decrease in cost per base and the increased throughputs associated with the latest technology advances will hopefully make this mode of data collection as appealing financially as it is scientifically.

3.1.3 Capture for Targeted Re-sequencing

A solution to the costs associated with whole genome re-sequencing has emerged in the form of "capture" technologies. These technologies target a portion of the genome, thereby reducing the number of reads, compared to whole genome sequencing, which are required to achieve useful levels of redundancy of sequencing coverage. Thus, for the same cost, the reduction in sequencing costs associated with the reduction in the number of reads allows the analysis of larger sample cohorts than whole genome re-sequencing.

An early example of "capturing" specific regions for next generation re-sequencing was the work of Thomas et al. (2006), in which 5 exons of the gene EGFR were targeted through PCR amplification of 11 regions of 100 bp each, followed by sequence analysis using a 454 sequencing instrument. By applying this strategy to 22 lung cancer samples, of which 9 were known to be EGFR mutant and 13 wild type, and generating between 8,000 and 12,000 reads for each one, they were able to correctly detect the known mutations, as well as discover mutations in two samples deemed to be wild type by Sanger sequencing. These mutations were found with low representation (e.g., a deletion in 9% of 4,488 reads) and had appeared in Sanger traces at a level indistinguishable from noise. The depth of sequencing coverage attained, along with the sensitivity of the 454 sequencing approach, resulted in one of the first studies to demonstrate the detection of cancer mutations present in only a portion of the cells in heterogeneous tumor tissue.

Although this type of approach, and PCR more generally, can be used to target regions for sequencing, they become cumbersome when the number of targeted regions becomes large. To address this limitation, several methods have been developed. One such method uses oligonucleotide constructs, named selectors (Dahl et al. 2005), that consist of two target-specific probes linked by a generic sequence.

In this strategy, the selectors hybridize to their targets and generate circular DNA complexes that are amplified and subsequently sequenced. As a downside, the selector method depends on the presence of restriction enzyme recognition sequences for probe design and DNA processing, which limit the target space. Another method, based on DNA circularization (Porreca et al. 2007), used probes created using microarray technology that were designed to bind directly upstream and downstream of the target regions without relying on enzymatic digestion. In this study, 55,000 exons were targeted of which ~10,000 were successfully captured and amplified in a single reaction; the low percentage of captured exons was attributed to shorter target lengths and regions with extreme G+C content. A different take on enrichment involved the use of high-density microarrays to capture DNA fragments (Albert et al. 2007) via hybridization. In this work, a ~385,000 probe microarray was designed to target 6,726 regions corresponding to 660 genes; captured DNA was then sequenced using a Roche/454 instrument. Of the resulting reads ~90% could be mapped to the reference genome, with ~70% mapping to targeted regions and, more importantly, a high successful capture rate was obtained with ~95% of targets being covered by at least one read.

Solution-phase hybridization approaches, such as those pioneered by Gnirke et al. (2009) are now commonly used for targeted re-sequencing. Foremost among these are "exome" reagents that are commercially available from several vendors. These reagents, such as those available from Nimblegen and Agilent, typically contain oligonucleotide probes that are designed to recover, through hybridization in solution, 10s of megabases representing exons and other regions of high biological interest. Hybridized fragments can be recovered from solution and sequenced using several next generation sequencing approaches. This approach is being employed by many including large consortia such as TCGA (http://tcga.cancer.gov/) and has been used to discover mutations implicated in different diseases such as renal carcinoma (Varela et al. 2011), Miller syndrome (Ng et al. 2010b), and Kabuki syndrome (Ng et al. 2010a).

As sequencing throughput continues to rise and costs continue to fall, the opportunity for multiplexing (simultaneously sequencing more than one sample in a single reaction) has emerged. One approach to multiplexing is to directly pool captured sequences and then sequence them simultaneously. This approach may be of utility in cases where the relationship between a sample and its "genotype" is not of interest (e.g., Van Tassell et al. 2008). A different strategy based on tagging templates before sequencing permits the identification of the source sample for each template. Several sequencing technologies provide a step during library construction where a sample-specific "barcode" or "index" sequence is appended to each fragment of the samples to be sequenced. In this way, each library can be indexed with a specific sequence, and libraries can be pooled and sequenced together in the same run. During analysis, the "barcodes" can be used to associate the individual reads, each identified with a library-specific sequence tag, back to the original starting sample, thereby preserving the relationship between the sequences and the starting material.

3.1.4 De Novo Sequencing

De novo sequencing, in contrast to "re-sequencing," is the term frequently used to describe the act of sequencing an organism's genetic material without the requirement for alignment of the sequence reads to a reference genome. This was typically accomplished using Sanger sequencing of large insert bacterial clones, but with the creation of appropriate assembly tools, high-throughput sequencing (HTS) approaches are now being used to produce reference genome sequences.

An early example of such work was that of Margulies et al. (2005). In this work, pyrosequencing was applied to generate the 500-kb genome sequence of *Mycoplasma genitalium*. Margulies et al. successfully assembled the short genome using 306,178 short reads of different lengths, varying between 80 and 120 bp. They demonstrated that the oversampling obtained using their method effectively dealt with sequencing errors and could be used to generate a high quality assembly. Hybrid approaches combining Sanger and pyrosequencing have also been proposed (Goldberg et al. 2006). In this work, a combined approach was deployed to sequence the genomes of six marine microbes. Goldberg et al. concluded that pyrosequencing can be useful in the completion of genome sequences that contain unclonable regions or regions with high secondary structure, both of which can render Sanger sequencing difficult if not impossible. Hierarchical shotgun sequencing approaches have also been proposed as a solution for de novo sequencing of larger genomes; the SHort Read Assembly Protocol, or SHARP (Sundquist et al. 2007), for example, includes a step where overlapping 150 kb fragments are cloned and subsequently sequenced using short reads. A key distinction with respect to the protocol used for the Human Genome Project is that the overlap between clones is deduced through analysis of the sequences rather than through clone-based physical mapping approaches such as clone fingerprinting (Marra et al. 1997).

Despite the shortcomings implicit in the short read lengths typical of several next generation sequencing technologies, and the repeat rich structure of many large genomes, next generation sequencing has been used to successfully obtain de novo sequences from a variety of organisms ranging from bacteria (Reinhardt et al. 2009), to plants like the wild soybean (Kim et al. 2010), to mammals, including the giant panda (Li et al. 2010a), human (Li et al. 2010b), and even extinct species such as the Neanderthal (Green et al. 2010) and the mammoth (Miller et al. 2008).

Sequencing does not need to be limited to one species at a time. The relative recent field of Metagenomics focuses on studying the genetic content of whole microbial communities by analyzing environmental samples. This technique allows the study of microbes that resist laboratory cultivation, which greatly amplifies the range of organisms available for analysis; additionally, by sampling communities as they exist in nature, it is possible to study how different species co-exist, as well as highlight features related to their adaptability to the environment. First approaches to explore microbial diversity in heterogeneous samples focused on the phylogenetic analysis of ribosomal RNA. Schmidt et al. (1991) used this methodology to identify 15 unique bacteria and 1 eukaryote from a sample of marine picoplankton.

Later studies showed that instead of rRNA, using protein-coding genes as markers provides a better picture of the microbial community and permits quantitative analysis on the abundance of each species (von Mering et al. 2007). Another approach focuses on the discovery of new genes, or variations of known genes, which convey a desired phenotype. Healy et al. (1995) showed that by transforming an *E. coli* strain with DNA containing genes obtained from a metagenomic sample, screening for the desired phenotype and sequencing those who presented it, novel variants of functional proteins could be discovered. Shotgun sequencing of DNA has been shown to be a useful technique for sampling the gene content of metagenomes, and was used to sample the Sargasso Sea (Venter et al. 2004) where 148 novel bacterial phylotypes and 1.2 million novel genes were identified. The ability to recover full genomes from shotgun sequencing has also been demonstrated for metagenomes represented by few species (Tyson et al. 2004).

HTS has increased the breadth of metagenomic projects. For example, Dinsdale et al. (2008) used DNA pyrosequencing to obtain gene samples from nine microbial communities of distinct environments. By comparing the prevalence of different types of genes between the environments, they showed it was possible to determine different metabolic requirements characteristic to each habitat. A problem when working with large communities is the number of diverse organisms involved in a metagenomic sample, as it is often difficult to get enough coverage on genes of interest; Iwai et al. (2009) proposed a protocol termed Gene-Targeted (GT)-Metagenomics to counteract this. The method includes DNA amplification of the gene of interest followed by pyrosequencing; by focusing sequencing capabilities to one gene it is then possible to analyze its genetic variation across microbes in a specific community. On a different take on metagenomic applications, Warren et al. (2009) proposed the use of HTS to profile T-cell receptors in peripheral blood as a way of exploring an individual's immune system state. Using simulated data they showed that, theoretically, short reads generated by HTS could be used to characterize T-cell receptors (TCRs) with a 61% sensitivity for rare (1 parts per million) and 99% for more abundant clonotypes (6 ppm or more). The same group then applied this technique to profile the T-cell repertoire in pooled blood samples from 550 individuals (Freeman et al. 2009); through the analysis of TCR mRNA expressed by T lymphocytes, the group was able to identify 33,664 TCR clonotypes at different abundances, greatly increasing the number of known receptors from the 3,187 known at the time. Using longer reads (100–150 bp) and exhaustive sequencing of two blood samples from a healthy donor, Warren et al. (2011) recently established a directly measured individual T cell repertoire size of at least 1 M distinct TCRs. Comparative studies between whole metagenomes (Tringe et al. 2005) have also been successfully applied to elucidate functional roles of microbiomes in host organisms. For example, Turnbaugh et al. (2006) compared the gut microbiomes of obese and lean mice, demonstrating that some forms of obesity can be caused by microbes in the gut being more efficient at energy extraction.

The capabilities of metagenomic analysis coupled with HTS has prompted the creation of new international efforts such as The Human Microbiome Project (Turnbaugh et al. 2007), which aims to extend the study of microbes in the human body.

As part of this project the genomes of 900 human microbes are being sequenced from different body sites (gastrointestinal tract, oral cavity, urogenital/vaginal tract, skin, and respiratory tract); a recent report by The Human Microbiome Jumpstart Reference Strains Consortium (2010) included the complete sequence and annotation of 178 of such genomes. Similarly, the Metagenomics of the Human Intestinal Track project (MetaHit) aims to study the human gut microbiome with a focus on two diseases: irritable bowel syndrome (IBD) and obesity. In a recent report, Qin et al. (2010) presented a catalogue of 3.3 million genes belonging to approximately 1,000 species of bacteria obtained through Illumina HTS of DNA derived from fecal samples of 124 individuals. These catalogues aim to facilitate downstream metagenomic studies on human samples.

3.1.5 Analysis Strategies

Bioinformatic analysis of sequencing data can be divided into several stages. The first step is technology dependent and deals with processing the data provided by the sequencing instrument. Downstream analysis is then done ad hoc to the type of experiment. When sequencing new genomes, de novo assemblies are required, which are possibly followed up with genome annotations. Re-sequencing projects use the short reads for aligning (or *mapping assembly*) against a reference sequence of the source organism; these alignments are then analyzed to detect events relevant to the experiment being conducted (e.g., mutation discovery, detection of structural variants, copy number analysis).

The first step of bioinformatic analysis starts during sequencing and involves signal analysis to transform the sequencing instruments fluorescent measurements into a sequence of characters representing the nucleotide bases. As sequencers image surfaces densely packed with the DNA sequencing templates and sequencing products, image processing techniques are required for detection of the nascent sequences and conversion of this detected signal into nucleotide bases. Most technologies assign a base quality to each of the nucleotides, which is usually a value representing the confidence of the called bases. Although each vendor has methods specific to their technology to evaluate base quality, most provide the user with a Phred (Ewing et al. 1998)-like Score value: a quality measurement based on a logarithmic scale encoding the probability of error in the corresponding base call.

To achieve contiguous stretches of overlapping sequence (contigs) in de novo sequencing projects, software that can detect sequence overlaps among large numbers of relatively short sequence reads is required. The process of correctly ordering the sequence reads, called *assembly*, is complicated by the short read length; the presence of sequencing errors; repeat structures that may reside within the genome; and the sheer volume of data that must be manipulated to detect the sequence overlaps.

To address such complications, hybrid methods involving complimentary technologies have been successful. For example, by mixing 200 bp 454 sequence reads

with Sanger sequences, Goldberg et al. (2006) successfully sequenced the genomes of several marine organisms. A different approach eliminated the need for Sanger sequencing by mixing two distinct next generation sequencing technologies (Reinhardt et al. 2009). By taking advantage of 454's longer reads (250 bp) with short Illumina reads (36 bp), Reinhardt et al. were able to de novo sequence a 6.5 Mb bacterial genome. These studies provided practical examples of how the strengths of different technologies can be used to alleviate their respective shortcomings.

Homology with previously sequenced organisms can help when sequencing new genomes. The use of this strategy was demonstrated during sequencing of the mouse genome (Gregory et al. 2002); by taking advantage of the conserved regions between mouse and human, Gregory et al. were able to build a physical map of mouse clones, establishing a framework for further sequencing. A similar approach can be used to produce better assemblies with next generation sequencing. For example, to sequence the genome of the fungus *Sordaria macrospora* (Nowrousian et al. 2010) short reads from 454 and Illumina instruments were first assembled using Velvet (Zerbino and Birney 2008) and the resulting contigs were then compared to draft sequences of related fungi (*Neurospora crassa*, *N. discreta,* and *N. tetrasperma*). This process helped produce a better assembly by reducing the number of contigs from 5,097 to 4,629, while increasing the N50 (the contig length N for which 50% of the genome is contained in contigs of length N or larger) from 117 kb to 498 kb.

More recently, new algorithms have been developed which can assemble genomes using only short reads. Most of these methods are based on *de Bruijn* graphs. Briefly, the logic involves decomposing short reads into shorter fragments of length *k* (*k-mers*). The graph is built by creating a node for each *k-mer* and drawing a link, or "edge," between two nodes when they overlap by *k*-1 bp. These edges specify a graph in which overlapping sequences are linked. Sequence features can increase the resulting graph's complexity. The graph can, for example, contain loops due to highly similar sequences (e.g., gene family members or repetitive regions), and so-called *bubbles* can be created when single base differences (e.g., due to polymorphisms or sequencing errors) result in the creation of nonunique edges in the graph which yield not one but two possible paths around the sites of the sequence differences. Graph complexity and size increase for large genomes and, given that the graph needs to be available in memory for efficient analysis, not all implementations can handle human size genomes. Some publicly available implementations, such as Velvet (Zerbino and Birney 2008) and Euler-SR (Chaisson and Pevzner 2008), have been successfully used to assemble bacterial genomes. Another implementation, ABySS (Simpson et al. 2009), makes use of parallel computing through the Message Passing Interface (MPI) to distribute the graph between many nodes in a computing cluster. In this way, ABySS can efficiently scale up for the assembly of human size genomes using a collection of inexpensive computers. Two newer assemblers SOAPdenovo (Li et al. 2010c) and ALLPATHS-LG (Gnerre et al. 2010) are able to assemble human-sized genomes using large memory multi-cpu servers, requiring 150 Gb and 512 Gb RAM, respectively.

For re-sequencing experiments, high-throughput aligners are required to map reads to the reference genome. Many applications have long been available for

sequence alignments; however, the amount and size of the short reads created by next generation sequencing technologies required the development of more efficient algorithms. Some methods use "hashing" approaches, such is the case of Maq (Li et al. 2008) in which the reads are reduced in complexity to unique identifier keys ("hashed"). These can then be used to scan a table made from a similarly "hashed" representation of the reference genome to identify putative read alignments to the reference. Other methods, based on Burrows-Wheeler transformation, have become popular for read alignment. These include BWA (Li and Durbin 2009), Bowtie (Langmead et al. 2009), and Soap (Li et al. 2009b). Although these algorithms are relatively fast compared to Maq (Li et al. 2008), they are somewhat limited when it comes to splitting a read to achieve gapped alignments, which can occasionally be required due to insertion/deletion sequence differences ("indels") between sequence data and the reference. The Mosaik aligner (Hillier et al. 2008) attempts to approach this by using a Smith and Waterman (1981) algorithm to align the short reads.

3.1.5.1 Mutation Discovery

Identification of single nucleotide variants (SNVs), point mutations, and small indels are central to the study of interspecific variation. Such sequence variants are heavily studied as some have been linked to specific diseases (Shah et al. 2009a; Morin et al. 2010; Ng et al. 2010b). Before HTS technologies were available, the detection of mutations in disease states did not typically involve sequence-intensive analysis on a genome-wide scale. Instead, candidate gene approaches and genome wide association studies (GWAS) were frequently used. In contrast, HTS provides the means for mutation discovery at single base resolution over entire exomes, transcriptomes, or genomes. An additional advantage of HTS is its enhanced sensitivity compared to typical Sanger approaches. For example, using Illumina sequencing, it was seen that HTS could be used to detect mutations that traditional Sanger sequencing could not detect due to low representation of the mutated allele (Thomas et al. 2006). In a different study, the high resolution of HTS was used to simultaneously detect single nucleotide polymorphisms (SNP) and estimate minor allele frequency (MAF) (Van Tassell et al. 2008). By sequencing three pooled samples of enzyme-digested DNA from 66 individuals representing three cattle populations (Holstein lineage, Angus bulls and a group of mixed beef breeds), Van Tassell et al. were able to identify 60,042 putative SNPs and estimate MAFs by analyzing the ratio of nonreference to reference reads across all 66 cattle. Whole genome re-sequencing can also be used for SNP discovery, as is exemplified by projects like the 1000 Genomes Project (http://www.1000genomes.org/) which takes a HTS approach to detection of variants in human populations instead of the SNP microarrays previously used in the latter phases of the HapMap Project (The International HapMap Consortium 2003).

As cancer is a genetic disease (Hanahan and Weinberg 2000) and mutations are known to drive cancers (Stratton et al. 2009), HTS approaches towards mutation

discovery in human cancers have become popular. Most studies seek to identify somatic mutations, which are sequence changes that occur in the tumor DNA but are absent in the normal, or "germline-derived," DNA from the same individual. Such studies are often composed of two phases: one where sequencing is done on a smaller set of samples to detect candidate mutations and a second phase where mutated genes are sequenced in a larger "extension" cohort to determine their frequency in a larger population. Prior to the ready availability of HTS machines, heroic efforts were often required to sample even a fraction of human genomes in the search for cancer mutations. For example, Sjöblom et al. (2006) used Sanger sequencing to sequence more than three million PCR products. These products included 13,023 genes from each of 11 breast and 11 colorectal cancer samples. The massive amount of work represented in the study for generation of the PCR products is now eliminated through the library construction procedures of HTS platforms.

The large number of *loci* in which candidate mutations can occur makes HTS an ideal platform for relatively unbiased mutation discovery. There are an increasing number of examples in which HTS approaches have been used to characterize cancers, and many more are expected over the near to medium time frame. For example, sequencing bone marrow and skin samples from an acute myeloid leukemia (AML) patient permitted the identification of somatic mutations, which in turn led to the identification of genes that were recurrently mutated in other patients (Mardis et al. 2009; Ley et al. 2010). In another study, sequencing a primary and a metastatic breast tumor, sampled 9 years apart from the same patient, showed that mutations that are dominant at a later stage may be present at low representation earlier during tumor progression, indicating that through selective pressure imposed by treatment, subpopulations of cells may become more prevalent over time (Shah et al. 2009b). These results and others like them demonstrate the utility and impact that HTS can have when analyzing the mutational landscape of cancer patients.

Bioinformatic analysis for mutation detection using HTS re-sequencing starts with the alignment of reads to reference genomes. False positive nucleotide mismatches are expected due to read mapping errors, sequencing errors, and to the existence of actual variants, and it is a current challenge to distinguish between these alternative possibilities. One conceptual approach to distinguishing technical errors from *bona fide* variants takes advantage of the considerable redundancy of sequence coverage that HTS produces. For example, when multiple reads cover the same position in the reference assembly, and these consistently indicate the presence of a sequence variant, confidence in the robustness of the variant call is increased. To infer whether a mutation is present, the ratio between the sequence coverage of the reference allele and the nonreference allele needs to be evaluated. This can be done with hard thresholds on the ratio of supporting reads or with statistical methods that allow for some flexibility by assigning a score or probability of the mismatch being a mutation (Li et al. 2008, 2009b; Goya et al. 2010). Indel detection can be more challenging, especially in cases where the pairwise alignment, meaning the base-by-base matching between the query and the target, is unambiguous over the same mapped location. Methods for calling indels often include local realignment of the reads at the site of

the candidate mutation (Li et al. 2009a). Mutations can also be called using the output from assemblies by analyzing coverage supporting the bubbles created in the *de Bruijn* graph (e.g., Simpson et al. 2009), as alluded to previously.

Once discovered, sequence variants can be evaluated for novelty and for proposed effect on the gene product. This is readily achieved through comparison of variants to databases containing previously observed variation, such as dbSNP (Sherry et al. 2001). Similarly, databases exist which contain genes and mutations that have been previously linked to diseases. Examples include the Online Mendelian Inheritance in Man (OMIM; http://www.ncbi.nlm.nih.gov/omim) database and the COSMIC (http://www.sanger.ac.uk/genetics/CGP/cosmic/) and Cancer Census (http://www.sanger.ac.uk/genetics/CGP/Census/) databases. If discovery of somatic mutations is the study aim, then comparison of sequence data from matched tumor and normal samples from the same individuals can be used to distinguish somatic changes from those resident in the germ line (Mardis et al. 2009; Shah et al. 2009b; Morin et al. 2010; Ley et al. 2010).

Additional bioinformatics approaches can be employed to assess the possible effect of the sequence variant on the gene product (Ng and Henikoff 2003). For example, in the case of protein coding genes, variants and mutations may affect a codon such that a different amino acid is encoded (*missense mutation*) or an early stop codon is created (*nonsense mutation*). Both types of variant can change the structure or function of the resulting protein, which may in turn be a driving factor in disease. Of similar interest is the distinction of driver mutations (those that are directly involved in the disease) versus passenger mutations (those that get propagated by association with other mutations but do not have a functional role). Differentiating between these types of mutations has been used as a way of identifying tumor genes. One method involves the comparison between the frequency of synonymous and nonsynonymous mutations over genes (Greenman et al. 2007). Assuming that silent mutations do not confer any advantage to a tumor, these can be used to model a mutational profile under the hypothesis of no-selection, any gene that presents nonsynonymous mutations at frequencies that deviate from this profile can then be inferred to be under selective pressure and thus may be directly involved in the disease (Greenman et al. 2007).

Although HTS is useful for the discovery of candidate variants including mutations, other experimental methods such as Sanger sequencing still need to be used to validate them. An overly sensitive mutation caller may generate a large amount of putative mutations which, given the large number of bases covered, may be impossible to validate using Sanger sequencing. On the opposite side of the scale, an overly specific algorithm may miss many important variants. It is important to try to find a balance between the False Positives (calling a mutation where there is none) and the False Negatives (missing a mutation). Different methods can be employed, such as requiring specific types of supporting evidence (e.g., minimum number of reads, coverage on both strands, SNV not near an indel) or varying parameters in order to maximize concordance to known databases (e.g., dbSNP) or to approximate a transition/transversion ratio estimated for the organism in question.

3.1.5.2 Genomic Rearrangements

The availability of paired-end sequencing on HTS platforms has improved the detection of genome rearrangements to single base resolution (Chen et al. 2008). During paired-end sequencing library construction, sheared DNA fragments of a desired size are selected and prepared for sequencing. Paired-end sequencing involves sequencing both ends of the DNA fragments in the sequencing library. The resulting reads are thus matched in pairs, which are expected to align to the reference genome a certain distance apart, following the fragment size distribution specified during library construction, and in a certain orientation with respect to each other. The paired end read alignments can be assessed for inconsistencies in the orientation or distance between the paired ends. If these are detected, the presence of a genome rearrangement is inferred. For example, if the expected insert size (number of nucleotides between the sequenced reads from individual fragments) is 200 bp, but the paired reads map back to the reference genome 50,000 bp apart, one can infer that 50,000 bp have been lost, or deleted, in the source genome, relative to the reference genome. Using similar logic, if reads map to different chromosomes, a translocation may have been detected. Given that the library fragments are sequenced to yield tail to tail read orientations (in the case of Illumina paired-end sequences), one expects the reads to align to opposite strands in the reference. If this tail to tail read orientation is not preserved in the alignments, an inversion may be inferred.

There are several examples of the use of paired-end sequences to detect genome rearrangements. For example, Korbel et al. (2007) used paired-end sequences to observe that structural variants in the human genome were more widespread than initially thought. Campbell et al. (2008) used paired-end sequences to identify somatic rearrangements in cancer, and subsequently compared the pattern of re-arrangements in primary and metastatic tumors to infer the clonal evolution of cancer (Campbell et al. 2010). Ding et al. (2010) analyzed related samples from a primary tumor, brain metastasis, and xenograft, and compared their mutational profiles; they observed that in both the metastasis and xenograft tumor the mutational representation was a subset of the primary tumor, indicating that cells from different tumor subpopulations gave rise to the metastasis and the xenograft. In a more recent study, Stephens et al. (2011) scanned ten chronic lymphocytic leukemia patients for rearrangements and stumbled upon one which presented 42 somatic rearrangements involving a single arm of chromosome 4, further studies validated similar events on additional samples. This massive chromosomal remodeling, involving 10s–100s rearrangements, is estimated to happen in 2–3% of all cancers. Stephens et al. named this phenomenon "chromothripsis," denoting a process in which a single catastrophic event shatters one or more chromosomes.

Another approach to detect genome rearrangements involves the use of de novo assembly methods such as ABySS (Simpson et al. 2009). Here, the approach involves assembling contigs and then aligning the contigs (as opposed to individual reads) back to the reference genome. The advantage to this approach over a read-based alignment approach is that by analyzing the reads in a de novo fashion, the assembler

may be able to create contigs that represent the exact breakpoint of rearrangement events. For large genomes with many rearrangements (e.g., a human tumor sample), de novo assemblies will not be able to reconstruct the original genome; however, a broken up assembly may contain sufficiently large contigs which can then be compared to the reference genome to determine putative structural differences.

3.2 Transcriptomics

HTS can also be applied to characterize various types of RNA transcripts, including mRNAs, small RNAs (including micro RNAs), noncoding RNAs, and antisense RNAs, collectively known as the *transcriptome*. Genes encoded in the genome are activated via transcription, depending on their nature some will be further processed into protein, others will remain in RNA form, and others may be degraded. Different species of RNA can be captured using specific protocols and characterized using HTS, thus obtaining a snapshot of the corresponding RNA content in the cells. Messenger RNA (mRNA), for example, can be captured by targeting its poly(A) + tail, in this way a representation of the expression of mostly protein coding genes can be obtained. Many advantages of HTS in genome sequencing also apply in transcriptome sequencing: base pair resolution enables mutation discovery, pair-end reads enable detection of fusion genes, and coverage is proportional to concentration in the source material allowing for quantitative analysis. Transcriptome sequencing allows researchers to look at expression profiles with unprecedented detail.

3.2.1 RNA-Seq

Early efforts to explore transcriptomes used expressed sequence tags (EST). The EST technique involved the creation of cloned cDNA molecules from mRNA templates and sequencing 3′ or 5′ ends using Sanger sequencing. This approach helped catalyze gene discovery in many species, including human and mouse (Adams et al. 1993; Hillier et al. 1996; Marra et al. 1999). Sequencing of full-length cDNA clones eventually became feasible, fueled by the clear advantages of understanding transcript isoform structure at the sequence level (Strausberg et al. 1999; Gerhard et al. 2004; Ota et al. 2004).

Although ESTs provided rapid access to expressed genes, they were not optimal for gene expression profiling, largely due to their significant cost. The development of short (14–21 bp) serial analysis of gene expression (SAGE, Velculescu et al. 1995) tags and derivative technologies addressed cost issues by making it possible to detect expression of 30 or more transcripts in a single pass sequencing read, as opposed to a single transcript as in the EST technique. The increased number of transcripts detected using SAGE made this technology useful for gene expression profiling (Yamamoto et al. 2001; Polyak and Riggins 2001), with tag counts reflecting transcript abundance. Bias of SAGE tags to the 3′ ends of transcripts was addressed

through the development of the CAGE (cap analysis of gene expression, Kodzius et al. 2006) technique, which allows analysis of tag sequences adjacent to 5′ transcript cap structures.

Shortly after HTS approaches became available, transcript analysis technologies were adapted for use on ultra-high-throughput sequencers. For example, a version of SAGE called "DeepSAGE" was developed (Nielsen et al. 2006) on the 454 instrument, allowing an approximately sixfold enhancement in tag counts, from 50,000 to 300,000. An important advantage of certain tag sequencing approaches is that they allow one to determine whether the transcript originated from the forward or the reverse strand in the genome. Tag-Seq (Morrissy et al. 2009), derived from SAGE, can be used to measure expression values of genes with strand specificity. Recent developments have enabled the construction of strand-specific transcriptome libraries (Levin et al. 2010) for sampling entire cDNAs. These and other strand-specific approaches are enabling studies of the relationship between sense and antisense gene expression (Yassour et al. 2010).

Expressed sequences, longer than short tags, have also been analyzed using next generation sequencing. By capturing poly(A)+mRNA molecules and using a shotgun style approach akin to that previously defined for the genome, the entire mRNA content of a sample can be sequenced. This approach is known as whole transcriptome shotgun sequencing (WTSS) or RNA-Seq. In one study (Bainbridge et al. 2006), the transcriptome of a human prostate cancer cell line was explored using pyrosequencing. Analysis of the 181,279 reads of 102 bp average length obtained revealed the expression of 10,117 genes. A subsequent study explored the transcriptome of the HeLa S3 cell line (Morin et al. 2008), where random priming and sonication were used to produce coverage of entire cDNAs, and gene expression data were collected alongside exon-level expression and information on SNVs.

HTS of complete mRNA species can be used to detect mutations and help find "cancer genes." Mutations can be discovered in expressed transcripts by analyzing repeated coverage of nonreference alleles. For example, through the analysis of 15 transcriptome libraries of ovarian cancer samples, Shah et al. (2009a) were able to detect a mutation in a transcription factor gene (*FOXL2*) present only in a specific subtype known as granulosa-cell tumors (GCT). This mutation was then validated in matched DNA and was later found to be present in a larger cohort of GCT samples (86 out of 89) and not present in 149 epithelial ovarian tumors. In a similar setting, by sequencing the transcriptome of 31 diffuse large B-cell lymphomas (DLBCL), Morin et al. (2010) detected a recurrently mutated codon in the gene *EZH2*, encoding a histone methyltransferase. Through DNA sequencing of the *locus* containing the mutation in 251 follicular lymphoma samples (FL) and 320 DLBCL samples, it was determined that the codon was recurrently mutated mainly in DLBCL samples specific to the germinal-center origin subtype, and to FL samples. These two studies exemplify the utility of transcriptome sequencing in the study of cancer.

Genome rearrangements can also be detected using transcriptome sequencing. One of the effects seen as a result of genome translocations is the joining of two genes into a fusion gene, expressed at the transcriptome level as a "chimera transcript" (Mitelman et al. 2004). As part of the genome rearrangements in cancer study carried out by Campbell et al. (2008), putative fusion genes were investigated using reverse

transcription polymerase chain reaction (RT-PCR), two such fusion events were validated: one translocation between chromosomes 2 and 12 resulted in the fusion CACNA2D4-WDR43, and the second fusion PVT1-CHD7 formed by a rearrangement t(8;8)(q12;q24). Transcriptome sequencing has been used directly to detect such gene fusions in cancer. One strategy (Maher et al. 2009) included the use of long reads (200–500 bp) from a 454 instrument to detect possible chimera transcripts and short reads from an Illumina sequencer to provide coverage across the putative breakpoint. Maher et al. sequenced the transcriptomes of tumor samples with known fusion genes and were able to successfully detect them using this approach. In a more recent study, Steidl et al. (2011) sequenced two lymphoma cell lines and used a novel fusion discovery tool called deFuse (unpublished, http://compbio.bccrc.ca) to identify gene fusion events. Four events of interest were successfully validated using experimental methods. Follow-up studies were focused on a specific fusion candidate containing the gene *CIITA*, which was further found to be frequently fused to a variety of genes. This gene was found to be rearranged in primary Hodgkin Lymphomas (8 out of 55) and primary mediastinal B cell lymphoma (29 out of 77) but very low in other lymphomas like DLBCL (4 out of 131). Additionally, the presence of *CIITA* fusions was significantly correlated with shorter survival times. This study clearly illustrates how state-of-the-art technology and bioinformatic analysis can be effectively used to further understanding of cancer and can have a direct impact in clinical applications.

3.2.1.1 Noncoding RNAs

In addition to mRNA, which typically encodes protein, there are numerous other RNA types such as micro RNA (miRNA), small interfering RNA (siRNA), ribosomal RNA (rRNA), transfer RNA (tRNA), small nucleolar RNA (snoRNA), and small nuclear RNA (snRNA). These comprise the so-called noncoding RNA (ncRNA) class. Massively parallel signature sequencing (Brenner et al. 2000) was used to explore ncRNAs in the plant *Arabidopsis thaliana* by sequencing short 17 bp signatures and mapping the reads back to the genome to determine their origin (Lu et al. 2005). The study observed that many small RNAs appeared to map back to regions of the genome previously considered devoid of genes, and determined that higher throughput is necessary to fully classify these species. The same group later updated the protocol to allow the sequencing of short RNAs on 454 or Illumina instruments (Lu et al. 2007). HTS has also been used to detect small RNAs in *C. elegans* (Ruby et al. 2006), a total of 394,926 reads were generated using a 454 instrument, and these reads confirmed 80 miRNAs previously seen in the library. Additionally, the high coverage provided by the HTS reads allowed the detection of an extra 13 annotated miRNAs not previously seen in the library, as well as the discovery of 18 novel miRNAs.

Mammalian genomes contain not only small RNAs but have also been found to express large noncoding RNAs located in intergenic space. An indirect method of determining the activity and location of these genes was employed by Guttman et al. (2009), by analyzing, across four mouse cell types, the distribution of histone

modifications indicative of active transcription, they were able to identify more than 1,600 long noncoding RNAs. Whole transcriptome shotgun sequencing was subsequently used (Guttman et al. 2010) to characterize these and other RNA species at the single nucleotide level. This study revealed an additional 1,140 multiexonic transcripts mapping in intergenic regions; 88% of which do not seem to be protein coding, while 12% appear to be novel genes due to high conservation and a valid open reading frame of more than 200 amino acids. These applications illustrate the high-throughput discovery capabilities of HTS.

3.2.2 Analysis Strategies

Data processing for whole transcriptome sequencing resembles that employed for genome sequencing, in that there are alignment-based approaches and de novo assembly-based approaches. A confounding factor when working with alignment-based methods is the existence of RNA splicing, which removes introns from transcripts during mRNA maturation. Splicing is a common feature of eukaryotic transcription; it has been estimated that 95% of multiexon genes are alternatively spliced (Pan et al. 2008) in a tissue- and developmental stage-specific manner (Huang et al. 1993). Thus, a significant proportion of sequence reads generated from mature mRNA molecules will represent "junction sequences," which span exon–exon junctions created during the splicing process. These junction sequences are not encoded as linear strings in the genome, and therefore reads that emanate from these sequences will not align to the genome and thus will not be detected. Given the large quantity of short HTS reads and the relatively large size of the introns, a gapped alignment strategy to detect junction read sequences is computationally expensive. New approaches have thus been developed that address this issue.

One approach to address complications arising from inefficiencies of read mapping due to mRNA splicing is to construct a database containing all the sequences formed by the possible combinations between exons. This database can then be appended to the reference genome, and aligners such as Maq (Li et al. 2008), BWA (Li and Durbin 2009), Bowtie (Langmead et al. 2009), Soap (Li et al. 2009b), or others can be used. This approach is efficient, but is constrained to the knowledge of the existing transcripts. Annotation databases that may serve as the sources for exons include Ensembl, UCSC Genes, RefSeq, CCDS, Vega, Havana, Encode, and AceView. As each of these source databases has individual pipelines and quality metrics, any one of them may individually include or exclude some isoforms found in other databases.

Another appealing set of tools provides a means for discovery of novel exon junctions. These include tools such as TopHat (Trapnell et al. 2009), HMMSplicer (Dimon et al. 2010), and SpliceMap (Au et al. 2010). These tools attempt to discover junction sequences using modified versions of aligners. Tophat and HMMSplicer,

for example, use the Bowtie aligner at an initial stage to align the reads onto the reference genome. Exon reads are thus aligned, and any reads unaligned after this stage are used for splice junction discovery.

Another challenging problem related to splicing is the determination of the sequences of complete transcripts, in which the appropriate exons and exon junctions are correctly assembled with respect to each other. Two approaches based on alignments have been proposed: Cufflinks (Trapnell et al. 2010) and Scripture (Guttman et al. 2010). Both rely on splice site predictions and alignments done by Tophat and subsequently apply different statistical and probabilistic methods to determine the combination of exons that most likely explain the reads observed in the sequencing data. Trans-ABySS (Robertson et al. 2010) applies the *de Bruijn* graph method to assemble transcriptome reads de novo, thereby allowing transcript isoform reconstruction and also allowing the detection of gene fusions and other novel transcript structures. Unlike splicing, the sequences that form gene fusions may not be adjacent in the reference genome. For example, when chromosomal rearrangements, such as translocations, occur in the genome, two genes may be placed adjacent to each other, and a hybrid transcript, containing sequences from both genes, may be expressed from that *locus* (Mitelman et al. 2004; Maher et al. 2009; Steidl et al. 2011).

3.2.2.1 Expression Analysis

Once reads have been assigned to transcripts and to genes, the number of reads mapping to a gene can be used to approximate transcript abundance (i.e., "gene expression"). One of the first and most popular methods for transforming read counts to gene expression measurements is known as reads per kilobase of gene model per million mapped reads or *RPKM* (Mortazavi et al. 2008). The idea behind RPKM is to normalize the number of reads against two factors: (1) the size of the gene, so as to avoid bias resulting from the increased number of reads that map to large genes and (2) the total number of reads in the library, so that measurements from libraries with deeper coverage do not get artificially inflated during interlibrary comparisons. Other methods, such as the one implemented in Cufflinks (Trapnell et al. 2010), resort to probabilistic models where each read is assigned to the isoform most likely to have spawned it. Once gene expression values are obtained, comparisons of mRNA abundance between samples can be made.

RNA-Seq data can be used to measure RNA abundance at the level of the entire gene, but also at the more granular level of individual exons. In this way, it is possible to search for both genes and exons that are enriched or depleted in sample set comparisons. An open source platform called Alexa-Seq (Griffith et al. 2010) is available, which automates the analysis of RNA-Seq libraries for alternative expression analysis. Although it relies on known annotations to speed up the analysis, it is able to detect some novel events such as exon skipping, retained introns, alternative 5′ and 3′ splice sites, alternative polyadenylation sites as well as alternative transcription start sites. Results can then be made available through a comprehensive web-based interface for visualization and downstream analysis.

Although discovery of SNVs, including mutations, using transcriptome sequence data is limited to those genes that are expressed, such data have been a rich source of mutated transcripts implicated in cancer progression. As described previously, this has been successfully done in the analysis of HeLa S3 cell transcriptomes and in the identification of a recurrently mutated codon in *EZH2* in DLBCL tumors of germinal-center origin (Morin et al. 2008, 2010), as well as in ovarian cancer by identifying a recurrent mutation in the *FOXL2* gene specific to the GCT subtype (Shah et al. 2009a). Although mutation discovery approaches in transcriptome are similar to those applied in genome data, care should be taken as changes in the ratio of reference versus nonreference reads, unlike in the genomic case, is not directly dependent on the number of copies of the chromosome that exist in the genome, but on the expression of the gene itself. Deviations from a 50/50 ratio could be caused by different levels of expression of each allele, whether in the same cell or in a heterogeneous sample.

The ability to detect gene expression at single base resolution offers the opportunity to measure expression from each allele individually. For example, human genomes are composed of one copy of the genome inherited from each of the parents, and these copies differ substantially from each other in their nucleotide sequence. Using RNA-Seq data, it is possible to determine which of the two alleles is expressed more abundantly at all the expressed *loci* where sequence differences between the parental alleles exist. Such measurements can be extended to disease states, where a somatic mutation with a potential adverse effect on a gene's product is not expressed, making the mutation effectively transcriptionally silent. In cases when both alleles are being transcribed we can measure allele-specific expression by comparing the presence of each allele in the expressed mRNA (Yan 2002). Care should be taken when analyzing this type of data, as it has been shown that some mapping related bias exists that can alter the results (Degner et al. 2009). Additional DNA genotyping may be needed to determine the true genotype at a specific *locus*. Having matching information at the genome level may also help detect events of RNA editing, in which organism-specific mechanisms may directly alter the sequence of a transcript. In their study, Shah et al. (2009b) detected changes between the genome and transcriptome sequencing of a breast cancer sample. They detected the gene coding for ADAR enzyme, responsible for A to G edits to be highly expressed; additionally they observed two genes (*COG3* and *SRP9*) showing high frequency of RNA editing. Although presently this may be prohibitively expensive, this example illustrates the high value of using complementary approaches to better understand the machinery behind phenotypic traits.

3.3 Epigenomics

Cells throughout an organism share a common DNA sequence but can have substantially different functions and phenotypes caused by specific patterns of gene expression. These patterns have to be inherited across multiple cell divisions by factors

other than DNA to maintain a cell's lineage. The field of epigenomics is the study of those factors that are involved in the establishment and maintenance of such gene expression patterns, which do not impinge upon the DNA sequence per se, but rather on its regulation between "on" and "off" states. Two of the major epigenetic mechanisms by which genes can be activated or silenced are DNA methylation and histone modifications, both of which can be replicated along with DNA during mitosis. DNA methylation refers to the addition of a methyl group to the 5′ position of the base cytosine, which can silence genes by interfering with promoter recognition or by recruiting chromatin modifying proteins. Chromatin refers to a complex of DNA and proteins, predominantly histones. Histones serve to package the DNA, 147 bp of DNA wrap around a complex of four pairs of the core histones H3, H4, H2A, and H2B to form the nucleosome. The histones contain large unstructured tails that can be subjected to a variety of posttranslational modifications, which in turn can influence a variety of cellular processes, such as transcription and DNA repair (Kouzarides 2007). Nucleosome positioning can also influence transcription by altering RNA Polymerase II transcription rates (Hodges et al. 2009). Disruption of epigenetic mechanisms has lately become of increased interest due to linkages with cancer progression (Jones and Baylin 2007). HTS has accelerated the pace at which the epigenome can be studied.

3.3.1 DNA Methylation

Treating DNA with bisulfite causes the conversion of cytosines, but not 5′ methycytosines, to uracil residues (Wang et al. 1980), which then pair with adenines. Thus, un-methylated CG base pairs are converted to AT base pairs. After treatment the resulting DNA is sequenced, and any cytosines detected at this point will identify locations where the cytosine was methylated. This approach was used with Sanger sequencing to map the methylation patterns of human chromosomes 6, 20, and 22 (Eckhardt et al. 2006) as part of the Human Epigenome Project (Esteller 2006). A HTS pipeline for this method was termed BS-Seq (Cokus et al. 2008). By applying shotgun sequencing to bisulfite-treated DNA and generating ~3.8 billion mappable nucleotides using an Illumina instrument, Cokus et al. were able to generate a detailed methylation map of *A. thaliana*. The utility of BS-Seq on mammalian-sized genomes was also proven by obtaining and comparing the methylomes of two mouse embryonic stem cell samples: one wild type and one with a mutation in a gene involved in GC methylation maintenance; through the analysis of ~60 million nucleotides from each one, the mutant sample was found to have a lower (25%) methylation level than the wild type. In another study, a similar approach dubbed MethylC-Seq was proposed (Lister et al. 2008) and used to evaluate the methylome of *A. thaliana* as well as the effects of mutated methylation-related genes; by applying HTS to small RNA and messenger RNA, the group also explored the relationship between methylation and transcription. The same approach was later applied to a genome-wide comparison on methylation patterns between human embryonic stem cells and fetal fibroblasts, demonstrating the dynamic nature of epigenetic marks

(Lister et al. 2009); furthermore, the study suggested that stem cells possess a special silencing mechanism based on non-CpG context methylation. Approaches to lower the complexity of the sequenced data have also been proposed, such is the case of Reduced Representation Bisulfite Sequencing (RRBS), in which DNA is digested with a methylation-insensitive restriction enzyme in order to select fragments with informative CpG sites (Meissner et al. 2008). Although initially developed and applied in mouse cells, RRBS has more recently been used to analyze clinical samples, especially geared towards those with low genomic content, such as formalin-fixed, paraffin-embedded samples (Gu et al. 2010).

3.3.2 ChIP-Seq

Histones can be subjected to several posttranslational modifications, such as acetylation and methylation, which can affect the interaction between DNA and regulatory factors. Histone modifications have been studied using chromatin immunoprecipitation, in which DNA and its interacting proteins are crosslinked and an antibody specific to the protein of interest is used to recover the protein–DNA complex. The captured DNA is then analyzed using DNA sequencing to determine the distribution of the targeted protein across the genome. Other DNA-interacting proteins can also be studied using this method, including transcription factors. Sanger sequencing has previously been used to characterize sequences obtained through ChIP using an extension of the Serial Analysis of Gene Expression (SAGE) protocol (Roh et al. 2004). A related protocol, named Sequence Tag Analysis of Genomic Enrichment (Bhinge et al. 2007), was developed to use pyrosequencing to analyze binding sites of the transcription factor STAT1. As in the case of gene expression, the number of sequence tags generated from a genomic location can be used to infer the affinity the protein exhibits for regions of the genome. Robertson et al. (2007) introduced another method called ChIP-Seq to also study STAT1 binding sites; by combining chromatin immunoprecipitation with Illumina sequencing it was possible to obtain a whole genome map of protein–DNA interaction sites. By combining the use of antibodies able to bind to specific histone tail modifications and the HTS by synthesis technologies, genome-wide maps of chromatin modifications have been generated (Barski et al. 2007). Multiple cell types can also be compared to determine changes involved in cell differentiation. In one study, three mouse cell types (embryonic stem cells, neural progenitor cells, and embryonic fibroblasts) were characterized for several histone H3 methylation marks as well as RNA polymerase II (Mikkelsen et al. 2007). The relationship between specific histone modifications and active transcription was used in another study (Guttman et al. 2009) to determine the location of more than 1,600 large intervening noncoding RNAs (lincRNAs), the same group further explored the expression of these lincRNAs using RNA-Seq (Guttman et al. 2010). Recent studies in cancer have highlighted the importance of histone modifications in cancer, having found links between these epigenetic marks and breast cancer subtypes (Elsheikh et al. 2009), as well as identifying histone modification genes as cancer related (Morin et al. 2010).

3.4 Summary

For nearly 30 years, sequencing efforts revolved around Sanger sequencing with slow improvements in speed and throughput. During the last 5 years there has been a surge of new technologies that has allowed the field to advance at gigantic steps. With prices going steadily down and throughput increasing constantly, the bottleneck is quickly shifting from data generation to data analysis and interpretation. New computational approaches need to be developed to keep on par with emerging technologies and the integrated analyses required to further the study of complex biological systems.

References

Adams, M.D. et al., 1993. 3,400 new expressed sequence tags identify diversity of transcripts in human brain. *Nat Genet*, 4(3), pp.256–267.

Albert, T.J. et al., 2007. Direct selection of human genomic loci by microarray hybridization. *Nat Meth*, 4(11), pp.903–905.

Au, K.F. et al., 2010. Detection of splice junctions from paired-end RNA-seq data by SpliceMap. *Nucl. Acids Res.*, p.gkq211.

Bainbridge, M. et al., 2006. Analysis of the prostate cancer cell line LNCaP transcriptome using a sequencing-by-synthesis approach. *BMC Genomics*, 7(1), p.246.

Barski, A. et al., 2007. High-Resolution Profiling of Histone Methylations in the Human Genome. *Cell*, 129(4), pp.823–837.

Bhinge, A.A. et al., 2007. Mapping the chromosomal targets of STAT1 by Sequence Tag Analysis of Genomic Enrichment (STAGE). *Genome Research*, 17(6), pp.910–916.

Brenner, S. et al., 2000. Gene expression analysis by massively parallel signature sequencing (MPSS) on microbead arrays. *Nat Biotech*, 18(6), pp.630–634.

Campbell, P.J. et al., 2008. Identification of somatically acquired rearrangements in cancer using genome-wide massively parallel paired-end sequencing. *Nature Genetics*, 40(6), pp.722–729.

Campbell, P.J. et al., 2010. The patterns and dynamics of genomic instability in metastatic pancreatic cancer. *Nature*, 467(7319), pp.1109–1113.

Chaisson, M.J. & Pevzner, P.A., 2008. Short read fragment assembly of bacterial genomes. *Genome Research*, 18(2), pp.324–330.

Chen, W. et al., 2008. Mapping translocation breakpoints by next-generation sequencing. *Genome Research*, 18(7), pp.1143–1149.

Cokus, S.J. et al., 2008. Shotgun bisulphite sequencing of the Arabidopsis genome reveals DNA methylation patterning. *Nature*, 452(7184), pp.215–219.

Dahl, F. et al., 2005. Multiplex amplification enabled by selective circularization of large sets of genomic DNA fragments. *Nucleic Acids Research*, 33(8), p.e71.

Degner, J.F. et al., 2009. Effect of read-mapping biases on detecting allele-specific expression from RNA-sequencing data. *Bioinformatics*, 25(24), pp.3207–3212.

Dimon, M.T., Sorber, K. & DeRisi, J.L., 2010. HMMSplicer: A Tool for Efficient and Sensitive Discovery of Known and Novel Splice Junctions in RNA-Seq Data. *PLoS ONE*, 5(11), p.e13875.

Ding, L. et al., 2010. Genome remodelling in a basal-like breast cancer metastasis and xenograft. *Nature*, 464(7291), pp.999–1005.

Dinsdale, E.A. et al., 2008. Functional metagenomic profiling of nine biomes. *Nature*, 452(7187), pp.629–632.

Eckhardt, F. et al., 2006. DNA methylation profiling of human chromosomes 6, 20 and 22. *Nat Genet*, 38(12), pp.1378–1385.
Elsheikh, S.E. et al., 2009. Global Histone Modifications in Breast Cancer Correlate with Tumor Phenotypes, Prognostic Factors, and Patient Outcome. *Cancer Research*, 69(9), pp.3802–3809.
Esteller, M., 2006. The necessity of a human epigenome project. *Carcinogenesis*, 27(6), pp.1121–1125.
Ewing, B. et al., 1998. Base-Calling of Automated Sequencer Traces UsingPhred. I. Accuracy Assessment. *Genome Research*, 8(3), pp.175–185.
Fleischmann, R. et al., 1995. Whole-genome random sequencing and assembly of Haemophilus influenzae Rd. *Science*, 269(5223), pp.496–512.
Freeman, J.D. et al., 2009. Profiling the T-cell receptor beta-chain repertoire by massively parallel sequencing. *Genome Research*, 19(10), pp.1817–1824.
Gerhard, D.S. & et al., 2004. The Status, Quality, and Expansion of the NIH Full-Length cDNA Project: The Mammalian Gene Collection (MGC). *Genome Research*, 14(10b), pp.2121–2127.
Gibbs, R.A. et al., 2004. Genome sequence of the Brown Norway rat yields insights into mammalian evolution. *Nature*, 428(6982), pp.493–521.
Gnerre, S. et al., 2010. High-quality draft assemblies of mammalian genomes from massively parallel sequence data. *Proceedings of the National Academy of Sciences*, 108(4), pp.1513–1518.
Gnirke, A. et al., 2009. Solution hybrid selection with ultra-long oligonucleotides for massively parallel targeted sequencing. *Nat Biotech*, 27(2), pp.182–189.
Goffeau, A. et al., 1996. Life with 6000 Genes. *Science*, 274(5287), pp.546–567.
Goldberg, S.M.D. et al., 2006. A Sanger/pyrosequencing hybrid approach for the generation of high-quality draft assemblies of marine microbial genomes. *Proceedings of the National Academy of Sciences*, 103(30), pp.11240–11245.
Goya, R. et al., 2010. SNVMix: predicting single nucleotide variants from next-generation sequencing of tumors. *Bioinformatics*, 26(6), pp.730–736.
Green, R.E. et al., 2010. A Draft Sequence of the Neandertal Genome. *Science*, 328(5979), pp.710–722.
Greenman, C. et al., 2007. Patterns of somatic mutation in human cancer genomes. *Nature*, 446(7132), pp.153–158.
Gregory, S.G. et al., 2002. A physical map of the mouse genome. *Nature*, 418(6899), pp.743–750.
Griffith, M. et al., 2010. Alternative expression analysis by RNA sequencing. *Nat Meth*, 7(10), pp.843–847.
Gu, H. et al., 2010. Genome-scale DNA methylation mapping of clinical samples at single-nucleotide resolution. *Nat Meth*, 7(2), pp.133–136.
Guttman, M. et al., 2009. Chromatin signature reveals over a thousand highly conserved large non-coding RNAs in mammals. *Nature*, 458(7235), pp.223–227.
Guttman, M. et al., 2010. Ab initio reconstruction of cell type-specific transcriptomes in mouse reveals the conserved multi-exonic structure of lincRNAs. *Nat Biotech*, 28(5), pp.503–510.
Hanahan, D. & Weinberg, R.A., 2000. The Hallmarks of Cancer. *Cell*, 100(1), pp.57–70.
Healy, F.G. et al., 1995. Direct isolation of functional genes encoding cellulases from the microbial consortia in a thermophilic, anaerobic digester maintained on lignocellulose. *Applied Microbiology and Biotechnology*, 43(4), pp.667–674.
Hillier, L.D. et al., 1996. Generation and analysis of 280,000 human expressed sequence tags. *Genome Research*, 6(9), pp.807–828.
Hillier, L.W. et al., 2008. Whole-genome sequencing and variant discovery in *C. elegans*. *Nat Meth*, 5(2), pp.183–188.
Hodges, C. et al., 2009. Nucleosomal Fluctuations Govern the Transcription Dynamics of RNA Polymerase II. *Science*, 325(5940), pp.626–628.
Huang, J.P. et al. (1993). Genomic structure of the locus encoding protein 4.1. Structural basis for complex combinational patterns of tissue-specific alternative RNA splicing. *Journal of Biological Chemistry*, (268).5, pp.3758–3766.
ICGC, 2010. International network of cancer genome projects. *Nature*, 464(7291), pp.993–998.

Iwai, S. et al., 2009. Gene-targeted-metagenomics reveals extensive diversity of aromatic dioxygenase genes in the environment. *ISME J*, 4(2), pp.279–285.
Jones, P.A. & Baylin, S.B., 2007. The Epigenomics of Cancer. *Cell*, 128(4), pp.683–692.
Kim, M.Y. et al., 2010. Whole-genome sequencing and intensive analysis of the undomesticated soybean (Glycine soja Sieb. and Zucc.) genome. *Proceedings of the National Academy of Sciences*, 107(51), pp.22032–22037.
Kodzius, R. et al., 2006. CAGE: cap analysis of gene expression. *Nat Meth*, 3(3), pp.211–222.
Korbel, J.O. et al., 2007. Paired-End Mapping Reveals Extensive Structural Variation in the Human Genome. *Science*, 318(5849), pp.420–426.
Kouzarides, T., 2007. Chromatin Modifications and Their Function. *Cell*, 128(4), pp.693–705.
Lander, E.S. & {International Human Genome Sequencing Consortium}, 2001. Initial sequencing and analysis of the human genome. *Nature*, 409(6822), pp.860–921.
Langmead, B. et al., 2009. Ultrafast and memory-efficient alignment of short DNA sequences to the human genome. *Genome Biology*, 10(3), p.R25.
Levin, J.Z. et al., 2010. Comprehensive comparative analysis of strand-specific RNA sequencing methods. *Nat Meth*, 7(9), pp.709–715.
Ley, T.J. et al., 2010. DNMT3A Mutations in Acute Myeloid Leukemia. *New England Journal of Medicine*, 363(25), pp.2424–2433.
Li, H. & Durbin, R., 2009. Fast and accurate short read alignment with Burrows-Wheeler transform. *Bioinformatics*, 25(14), pp.1754–1760.
Li, H. et al., 2009. The Sequence Alignment/Map format and SAMtools. *Bioinformatics (Oxford, England)*, 25(16), pp.2078–2079.
Li, H., Ruan, J. & Durbin, R., 2008. Mapping short DNA sequencing reads and calling variants using mapping quality scores. *Genome Research*, 18(11), pp.1851–1858.
Li, R., Fan, W. et al., 2010. The sequence and de novo assembly of the giant panda genome. *Nature*, 463(7279), pp.311–317.
Li, R., Li, Y. et al., 2010. Building the sequence map of the human pan-genome. *Nat Biotech*, 28(1), pp.57–63.
Li, R. et al., 2009. SOAP2: an improved ultrafast tool for short read alignment. *Bioinformatics*, 25(15), pp.1966–1967.
Li, R. et al., 2010. De novo assembly of human genomes with massively parallel short read sequencing. *Genome Research*, 20(2), pp.265–272.
Lister, R. et al., 2008. Highly Integrated Single-Base Resolution Maps of the Epigenome in Arabidopsis. *Cell*, 133(3), pp.523–536.
Lister, R. et al., 2009. Human DNA methylomes at base resolution show widespread epigenomic differences. *Nature*, 462(7271), pp.315–322.
Lu, C., Meyers, B.C. & Green, P.J., 2007. Construction of small RNA cDNA libraries for deep sequencing. *Methods*, 43(2), pp.110–117.
Lu, C. et al., 2005. Elucidation of the Small RNA Component of the Transcriptome. *Science*, 309(5740), pp.1567–1569.
Maher, C.A. et al., 2009. Transcriptome sequencing to detect gene fusions in cancer. *Nature*. Available at: http://dx.doi.org/10.1038/nature07638 [Accessed February 27, 2009].
Mardis, E.R. et al., 2009. Recurring Mutations Found by Sequencing an Acute Myeloid Leukemia Genome. *New England Journal of Medicine*, 361(11), pp.1058–1066.
Margulies, M. et al., 2005. Genome sequencing in microfabricated high-density picolitre reactors. *Nature*, 437(7057), pp.376–380.
Marra, M. et al., 1999. An encyclopedia of mouse genes. *Nat Genet*, 21(2), pp.191–194.
Marra, M.A. et al., 1997. High Throughput Fingerprint Analysis of Large-Insert Clones. *Genome Research*, 7(11), pp.1072–1084.
McPherson, A. et al., 2011. deFuse: An Algorithm for Gene Fusion Discovery in Tumor RNA-Seq Data. PLoS Comput Biol, 7(5), p.e1001138.
Meissner, A. et al., 2008. Genome-scale DNA methylation maps of pluripotent and differentiated cells. *Nature*, 454(7205), pp.766–770.
von Mering, C. et al., 2007. Quantitative Phylogenetic Assessment of Microbial Communities in Diverse Environments. *Science*, 315(5815), pp.1126–1130.

Mikkelsen, T.S. et al., 2007. Genome-wide maps of chromatin state in pluripotent and lineage-committed cells. *Nature*, 448(7153), pp.553–560.
Miller, W. et al., 2008. Sequencing the nuclear genome of the extinct woolly mammoth. *Nature*, 456(7220), pp.387–390.
Mitelman, F., Johansson, B. & Mertens, F., 2004. Fusion genes and rearranged genes as a linear function of chromosome aberrations in cancer. *Nat Genet*, 36(4), pp.331–334.
Morin, R.D. et al., 2008. Profiling the HeLa S3 transcriptome using randomly primed cDNA and massively parallel short-read sequencing. *BioTechniques*, 45(1), pp.81–94.
Morin, R.D. et al., 2010. Somatic mutations altering EZH2 (Tyr641) in follicular and diffuse large B-cell lymphomas of germinal-center origin. *Nat Genet*, 42(2), pp.181–185.
Morrissy, A.S. et al., 2009. Next-generation tag sequencing for cancer gene expression profiling. *Genome Research*, 19(10), pp.1825–1835.
Mortazavi, A. et al., 2008. Mapping and quantifying mammalian transcriptomes by RNA-Seq. *Nat Meth*, 5(7), pp.621–628.
Mouse Genome Sequencing Consortium, 2002. Initial sequencing and comparative analysis of the mouse genome. *Nature*, 420(6915), pp.520–562.
Ng, P.C. & Henikoff, S., 2003. SIFT: predicting amino acid changes that affect protein function. *Nucleic Acids Research*, 31(13), pp.3812–3814.
Ng, S.B., Bigham, A.W. et al., 2010. Exome sequencing identifies MLL2 mutations as a cause of Kabuki syndrome. *Nat Genet*, 42(9), pp.790–793.
Ng, S.B., Buckingham, K.J. et al., 2010. Exome sequencing identifies the cause of a mendelian disorder. *Nat Genet*, 42(1), pp.30–35.
Nielsen, K.L., Hogh, A.L. & Emmersen, J., 2006. DeepSAGE – digital transcriptomics with high sensitivity, simple experimental protocol and multiplexing of samples. *Nucleic Acids Research*, 34(19), pp.e133–e133.
Nowrousian, M. et al., 2010. De novo Assembly of a 40 Mb Eukaryotic Genome from Short Sequence Reads: Sordaria macrospora, a Model Organism for Fungal Morphogenesis. *PLoS Genet*, 6(4), p.e1000891.
Ota, T. et al., 2004. Complete sequencing and characterization of 21,243 full-length human cDNAs. *Nat Genet*, 36(1), pp.40–45.
Pan, Q. et al., 2008. Deep surveying of alternative splicing complexity in the human transcriptome by high-throughput sequencing. *Nat Genet*, 40(12), pp.1413–1415.
Polyak, K. & Riggins, G.J., 2001. Gene discovery using the serial analysis of gene expression technique: implications for cancer research. *Journal of Clinical Oncology: Official Journal of the American Society of Clinical Oncology*, 19(11), pp.2948–2958.
Porreca, G.J. et al., 2007. Multiplex amplification of large sets of human exons. *Nat Meth*, 4(11), pp.931–936.
Qin, J. et al., 2010. A human gut microbial gene catalogue established by metagenomic sequencing. *Nature*, 464(7285), pp.59–65.
Reinhardt, J.A. et al., 2009. De novo assembly using low-coverage short read sequence data from the rice pathogen Pseudomonas syringae pv. oryzae. *Genome Research*, 19(2), pp.294–305.
Robertson, G. et al., 2007. Genome-wide profiles of STAT1 DNA association using chromatin immunoprecipitation and massively parallel sequencing. *Nat Meth*, 4(8), pp.651–657.
Robertson, G. et al., 2010. De novo assembly and analysis of RNA-seq data. *Nat Meth*, 7(11), pp.909–912.
Roh, T. et al., 2004. High-resolution genome-wide mapping of histone modifications. *Nat Biotech*, 22(8), pp.1013–1016.
Ruby, J.G. et al., 2006. Large-Scale Sequencing Reveals 21U-RNAs and Additional MicroRNAs and Endogenous siRNAs in *C. elegans*. *Cell*, 127(6), pp.1193–1207.
Schmidt, T.M., DeLong, E.F. & Pace, N.R., 1991. Analysis of a marine picoplankton community by 16S rRNA gene cloning and sequencing. *J. Bacteriol.*, 173(14), pp.4371–4378.
Shah, S.P., Köbel, M. et al., 2009. Mutation of FOXL2 in Granulosa-Cell Tumors of the Ovary. *New England Journal of Medicine*, 360(26), pp.2719–2729.
Shah, S.P., Morin, R.D. et al., 2009. Mutational evolution in a lobular breast tumour profiled at single nucleotide resolution. *Nature*, 461(7265), pp.809–813.

Sherry, S.T. et al., 2001. dbSNP: the NCBI database of genetic variation. *Nucl. Acids Res.*, 29(1), pp.308–311.
Simpson, J.T. et al., 2009. ABySS: A parallel assembler for short read sequence data. *Genome Research*, 19(6), pp.1117–1123.
Sjöblom, T. et al., 2006. The Consensus Coding Sequences of Human Breast and Colorectal Cancers. *Science*, 314(5797), pp.268–274.
Smith, T.F. & Waterman, M.S., 1981. Identification of common molecular subsequences. *Journal of Molecular Biology*, 147(1), pp.195–197.
Staden, R., 1979. A strategy of DNA sequencing employing computer programs. *Nucleic Acids Research*, 6(7), pp.2601–2610.
Steidl, C. et al., 2011. MHC class II transactivator CIITA is a recurrent gene fusion partner in lymphoid cancers. *Nature*, 471(7338), pp. 377–381.
Stephens, P.J. et al., 2011. Massive Genomic Rearrangement Acquired in a Single Catastrophic Event during Cancer Development. *Cell*, 144(1), pp.27–40.
Stratton, M.R., Campbell, P.J. & Futreal, P.A., 2009. The cancer genome. *Nature*, 458(7239), pp.719–724.
Strausberg, R.L. et al., 1999. The Mammalian Gene Collection. *Science*, 286(5439), pp.455–457.
Sundquist, A. et al., 2007. Whole-Genome Sequencing and Assembly with High-Throughput, Short-Read Technologies. *PLoS ONE*, 2(5), p.e484.
The 1000 Genomes Project Consortium, 2010. A map of human genome variation from population-scale sequencing. *Nature*, 467(7319), pp.1061–1073.
The *C. elegans* Sequencing Consortium, 1998. Genome Sequence of the Nematode *C. elegans*: A Platform for Investigating Biology. *Science*, 282(5396), pp.2012–2018.
The Human Microbiome Jumpstart Reference Strains Consortium, 2010. A Catalog of Reference Genomes from the Human Microbiome. *Science*, 328(5981), pp.994–999.
The International HapMap Consortium, 2003. The International HapMap Project. *Nature*, 426(6968), pp.789–796.
Thomas, R.K. et al., 2006. Sensitive mutation detection in heterogeneous cancer specimens by massively parallel picoliter reactor sequencing. *Nat Med*, 12(7), pp.852–855.
Trapnell, C., Pachter, L. & Salzberg, S.L., 2009. TopHat: discovering splice junctions with RNA-Seq. *Bioinformatics*, 25(9), pp.1105–1111.
Trapnell, C. et al., 2010. Transcript assembly and quantification by RNA-Seq reveals unannotated transcripts and isoform switching during cell differentiation. *Nat Biotech*, 28(5), pp.511–515.
Tringe, S.G. et al., 2005. Comparative Metagenomics of Microbial Communities. *Science*, 308(5721), pp.554–557.
Turnbaugh, P.J. et al., 2006. An obesity-associated gut microbiome with increased capacity for energy harvest. *Nature*, 444(7122), pp.1027–131.
Turnbaugh, P.J. et al., 2007. The Human Microbiome Project. *Nature*, 449(7164), pp.804–810.
Tyson, G.W. et al., 2004. Community structure and metabolism through reconstruction of microbial genomes from the environment. *Nature*, 428(6978), pp.37–43.
Van Tassell, C.P. et al., 2008. SNP discovery and allele frequency estimation by deep sequencing of reduced representation libraries. *Nat Meth*, 5(3), pp.247–252.
Varela, I. et al., 2011. Exome sequencing identifies frequent mutation of the SWI/SNF complex gene PBRM1 in renal carcinoma. *Nature*, 469(7331), pp.539–542.
Velculescu, V.E. et al., 1995. Serial Analysis of Gene Expression. *Science*, 270(5235), pp.484–487.
Venter, J.C. et al., 2001. The Sequence of the Human Genome. *Science*, 291(5507), pp.1304–1351.
Venter, J.C. et al., 2004. Environmental Genome Shotgun Sequencing of the Sargasso Sea. *Science*, 304(5667), pp.66–74.
Wang, R.Y., Gehrke, C.W. & Ehrlich, M., 1980. Comparison of bisulfite modification of 5-methyldeoxycytidine and deoxycytidine residues. *Nucleic Acids Research*, 8(20), pp.4777–4790.
Warren, R.L., Nelson, B.H. & Holt, R.A., 2009. Profiling model T-cell metagenomes with short reads. *Bioinformatics*, 25(4), pp.458–464.
Warren, R.L. et al., Exhaustive T-cell repertoire sequencing of human peripheral blood samples reveals signatures of antigen selection and a directly measured repertoire size of at least 1 million

clonotypes. *Genome Research,* Published in Advance February 24, 2011, doi:10.1101/gr.115428.110.

Wheeler, D.A. et al., 2008. The complete genome of an individual by massively parallel DNA sequencing. *Nature*, 452(7189), pp.872–876.

Whiteford, N. et al., 2005. An analysis of the feasibility of short read sequencing. *Nucleic Acids Research*, 33(19), p.e171.

Yamamoto, M. et al., 2001. Use of serial analysis of gene expression (SAGE) technology. *Journal of Immunological Methods*, 250(1–2), pp.45–66.

Yan, H., 2002. Allelic Variation in Human Gene Expression. *Science*, 297(5584), pp.1143–1143.

Yassour, M. et al., 2010. Strand-specific RNA sequencing reveals extensive regulated long antisense transcripts that are conserved across yeast species. *Genome Biology*, 11(8), p.R87.

Zerbino, D.R. & Birney, E., 2008. Velvet: Algorithms for de novo short read assembly using de Bruijn graphs. *Genome Research*, 18(5), pp.821–829.

Chapter 4
Computational Infrastructure and Basic Data Analysis for High-Throughput Sequencing

David Sexton

Abstract Next-generation sequencing requires enormous computational infrastructure resources. A single sequencing run can generate up to 100 gigabases of sequence and requires almost 3 terabytes of data storage. In order to transfer and analyze data on this scale, the data infrastructure must be up to the task. There are many solutions to creating this infrastructure. Local servers and data storage can be purchased and installed by the group, which is using the sequencing instrument. Communal resources at the home institution can be used to create the necessary infrastructure, or newer external cloud-based solutions can be used to store and analyze data. Each sequencing platform has slightly different infrastructure requirements, and the IT solution must be tailored to the systems in use or generalized to cover all possible instruments. The majority of installed instruments are based on Illumina technology with a smaller portion being ABI or 454. What is presented here is a solution that is generalized for any instrument, including not just the IT requirements but the bioinformatics requirements as well.

4.1 Introduction

The computational and information technology requirements for next-generation sequencing are extensive. As such, you must be able to effectively react to new types of experimental technology. Recently faced with an unprecedented flood of data generated by the next generation of DNA sequencers, groups found it necessary to respond quickly and efficiently to the informatics and infrastructure demands. This challenge needs to be faced in order to anticipate time and design considerations

D. Sexton (✉)
Center for Human Genetics Research, Vanderbilt University, 511F Light Hall,
2215 Garland Avenue, Nashville, TN 37232, USA
e-mail: david.sexton@chgr.mc.vanderbilt.edu

N. Rodríguez-Ezpeleta et al. (eds.), *Bioinformatics for High Throughput Sequencing*,
DOI 10.1007/978-1-4614-0782-9_4,

of necessary components, including infrastructure upgrades, staffing, and tools for data analyses and management.

The evolution of the sequencing instrumentation is far from static. Sequence throughput from this new generation of instruments continues to increase exponentially at the same time that the cost of sequencing a genome continues to fall. These realities make the technology accessible to greater numbers of investigators while leading them to a greater usage of sequencing for a variety of experimental techniques, including whole genome, tagged sequence, variation discovery, whole transcriptome analysis, RNA-Seq, and CHiP-Seq analysis. This places unique challenges upon the Bioinformatics groups, whose mission could vary from the support of a single department or sequencing core to a facility that supports many disparate and independent groups that run their own sequencers but rely on the group to host the informatics, research cyber-infrastructures, or both. It is worth noting that the initial investment in the instrument is accompanied by an almost equal investment in upgrading the informatics infrastructure of the institution, hiring staff to analyze the data produced by the instrument, and storing the data for future use. Many investigators do not realize that these extensive investments are necessary prior to purchasing the new technology. This is why it is advantageous to have a group capable of putting in place platforms that acquire, store, and analyze the very large datasets created by these instruments. A group already familiar with data of this type and complexity, dedicated to investigators, and jointly working with IT personnel, can span multiple domains rather effortlessly.

The large sequencing centers (e.g., Sanger, Broad Institute, and Washington University) have automated processes and architectures not generally replicable in medium and small sequencing groups. However, as these smaller groups obtain next-generation technology they can nevertheless learn lessons from the larger centers. Through collaboration and sharing best practices, small and medium-sized groups can prepare for the arrival of the technology and develop methods to manage and analyze the data. Many smaller groups have been actively collaborating to formulate best practices to set up platforms for next-generation sequencing.

4.2 Background

Several new sequencing methodologies have been developed, most of which are loosely based on fixing DNA sequences to glass beads or slides, amplification and tagging of the bases with compounds for visualization, image capture, and subsequent image analysis to derive base calls. Some of the techniques and manufacturers include sequencing by synthesis as used by the Genome Analyzer II (GAIIx) and HiSeq 2000 from Illumina, sequencing by ligation as used by the ABI SOLiD sequencer and by the polony sequencing technique developed by the Church Lab at Harvard Medical School, sequencing by hybridization as used by Affymetrix, and single molecule sequencing as used by Helicos, VisiGen (Now part of Life Sciences), and Pacific Biosciences. As of the end of 2010, the preponderance of data has come

from the GAIIx, which currently has the largest market penetration and is clearly the most established next-generation sequencing technology among the majority of institutions.

The uniqueness of these data stems from the number of files created and the size of those files generated during a sequencing run. For the GAIIx system, approximately 115,200 Tiff formatted files are produced per run, each at about 8 megabytes (MB) in size. This is approximately 1 terabyte (TB) of data, which must be moved from the capture workstation to the analysis resource. Other systems have similar data and image yields. A decision must be made about archiving these "raw" data for future analysis or discarding them in favor of resequencing. A mere 10–20 sequencing runs could overwhelm any storage and archiving system available to individual investigators. Analysis of the image files is accomplished by Illumina-provided CASAVA software or by any number of third-party applications. Since the instrument is typically run for 36–100 cycles, sequences of about 36–100 bases are produced, resulting in what are called short read sequences. Sequence of this length creates major impediments to assembly of complex genomes without the use of a reference. Currently, de novo assemblies are restricted to prokaryotic and bacterial genomes.

Even after image processing, base calling, and assembly, there will be approximately 300 GB of uncompressed primary data that must be stored either in flat files or in a database. Then, using public databases and tools, biological significance can be assigned to the sequence. Many of the current algorithms and software programs are unable to handle the number and size of the sequence reads that must therefore be modified for use. Currently, software to reliably visualize the sequence data and its assemblies is evolving. Additionally, the long-term storage of primary and derived data may be difficult for the investigator, necessitating centralized solutions.

Solutions to these issues can be accomplished with a small, dedicated group within organizations that are familiar with data of this type and complexity. Within each area, we will describe specific challenges, along with some possible solutions we have experienced ourselves and from the experience of other institutions. These may not be the only solutions or architectures, and there are certainly many and varied sources of information on these topics as the target requirements continue to move, but this perspective can serve as a starting point for a set of best practices derived from facilities that have already solved many of these issues.

4.3 Getting Started with the Next-Generation Manufacturers

The current instrument manufacturers, Illumina, Roche, and Applied Biosystems (Fig. 4.1), all provide a foundation workflow for running their systems. Instruments typically ship with modest compute and IT resources providing the ability to support a single run of the machine. A small cluster, server, or workstation directly attached to the instrument provides data capture along with the principal data analysis pipelines necessary to process the raw data acquired into base calls and sequence

Next Generation Sequencing Informatics

Below is a table with informatics and IT statistics for the major next-generation/massively parallel sequencing platforms.

Next-Generation Sequencing Statistics

Vendor:	Roche				Illumina				ABI				PacBio
Technology:	454				GA & HiSeq				SOLiD				RS
Platform:	GS20	FLX	Ti	GS jr	I	II	IIx	2000	1	2	3	4	V1
Reads: (M)	0.5	0.5	1.25	0.1	28	100	250	2000	40	115	320	1400	
Fragment													
Read Length:	100	200	400	400	35	50	100	125	25	35	50	75	2000
Run Time: (d)	0.25	0.3	0.4	0.4	3	3	5	5	6	5	8	8	0.1
Yield: (Gb)	0.05	0.1	0.5	0.4	1	5	25	100	1	4	12	150	
Rate: (Gb/d)	0.2	0.33	1.25	1.0	0.33	1.67	5	20	0.34	1.6	4	18.75	
Images: (TB)	0.1	0.1	0.03	0.02	0.5	1.1	2.8	N/A	1.8	2.5	1.9	N/A	N/A
PA Disk: (GB)	3	3	15	12	175	300	300	400	300	750	1200	1000	
PA CPU: (hr)	10	140	220	200	100	70	N/A	N/A	N/A	N/A	N/A	N/A	N/A
SRA: (GB)	0.5	1	4	4	30	50	2.5	10	100	140	600	600	
Paired-end													
Read Length:		200	400	400	2x35	2x50	2x100	2x125	2x25	2x35	2x50	2x75	
Insert: (kb)		3.5	3.5	3.5	0.2	0.2	0.2	0.2	3	3	2	2	
Run Time: (d)		0.3	0.4	0.4	6	10	10	10	12	10	16	14	
Yield: (Gb)		0.1	0.5	0.1	2	9	50	200	2	8	32	300	
Rate: (Gb/d)		0.33	1.25	1.0	0.33	1.67	5	20	0.34	5	3.8	21	
Images: (TB)		0.01	0.03	0.02	1	2.2	5.6	N/A	3.6	5	3.8	N/A	N/A
PA Disk: (GB)		3	15	12	300	500	550	750	600	1500	2400	2000	N/A
PA CPU: (hr)		140	220	200	160	120	N/A	N/A	N/A	N/A	N/A	N/A	N/A
SRA: (GB)		1	4	4	60	100	3.5	15	200	280	1200	1200	

Notes:

- Units: B – bytes, b – bases
- d is days
- M is Millions
- PA is primary analysis (includes image feature extraction and base calling)
- PA CPU is calculated as the wall clock multiplied by the number of CPU cores
- ABI SOLiD data, except rate, are representative of a single slide
- ABI SOLiD and Illumina GA IIx/HiSeq primary analysis is done on instrument
- 454 paired-end reads vary in length depending on location of internal adapter
- SRA is the size of the files (SFF, SRF, or FASTQ) that are submitted to the NCBI Short Read Archive

Fig. 4.1 Next-generation sequencing statistics. The expected output of modern sequencing devices in both gigabases and gigabytes. Storage statistics for each file type are also listed

alignments from the run itself. Lately, manufacturers are also providing additional analysis modules, complete with technical support, to help streamline the Primary Analysis pipeline. In most buying considerations, the purchase of these additional modules provides an immense overall cost savings to the small- and medium-sized group. In the case of the Illumina GAIIx, this translates into a small incremental investment for dedicated servers, which significantly shortens the overall run time as well as providing diagnostics of the image analysis pipeline through bundled technical support.

As researchers and core groups obtain more sequencers and are required to capture and store more than a single run at a time, they will need to grow quickly

into larger compute and storage infrastructures capable of supporting these additional needs as well as information management systems to manage not only the workflow and derived information but also the data itself. Although the next-generation instruments are becoming widespread throughout academic institutions and medical centers, they are still an emerging technology. Illumina sequencing, for example, has been available to the small-to-medium-sized groups since the summer of 2007. To date, technologists, IT groups, and informaticians have had a relatively short period of time in which to develop processes, best practices, and additional, more rigorous quality control/quality assurance (QC/QA) and laboratory information management system (LIMS) environments specific to their environments. As these technologies and algorithms emerge into academic, open-source, and vendor-supported offerings, groups will evaluate them against existing practices using previous datasets.

Additionally, the manufacturers themselves are rapidly developing their platforms with frequent improvements to their technology and informatics solutions. This may require re-analysis using the technology for new insights or at minimum a QA of the new revisions against older software versions using previously acquired data. This will continue to be the case as the scientific community demands longer individual read sequences and the manufacturers respond with changes and updates to optics, software, and chemistry, placing larger demands on institutions' IT requirements.

Because of the necessarily tight integration with IT, those Bioinformatics Facilities that do not already maintain their own research IT infrastructures, including hardware and systems administration resources, will need to lean heavily outside themselves, either on centralized institutional services, specialized computer consulting groups, or both. A final consideration for startup is accessibility to the sequencing facilities by these additional personnel. Troubleshooting technical issues during setup, configuration, and operation of these instruments will be necessary to assist lab operations.

4.4 Infrastructure and Data Analysis

4.4.1 Computational Considerations

Moving beyond the initial installation, the transcendent requirement for a group's cyberinfrastructure is flexibility. Given the rapidly changing environment described, the manufacturer may or may not initially provide a modest computational environment, slating this environment for a subsequent release or update of the instrument. Consequently, the computational resources will need to fill technical gaps now and be able to scale for future demand.

The Illumina analysis pipeline, consisting of image analysis, base calling, and initial alignment against a reference sequence, initially was shipped without a computational platform upon which to run it. Most bioinformatics groups either bought a large multiprocessor server or a small cluster into which the pipeline was

configured. Illumina recommends a pipeline server, 16-cores, 48 Gb of RAM, and a 10 TB disk array that hosts the additional components of the pipeline.

This configuration provides a computational starting point. It usually becomes necessary either to scale up the vendor-provided system or to perform offline, primary analysis. Troubleshooting the analysis pipeline, manipulating configuration or parameter files, QAing revisions to the pipeline, or evaluating different algorithms requires a separate compute environment so that resources attached to the instrument can be used for the continued sequencing runs.

Two examples of initial configurations that have been successful are based on blades or discrete servers, respectively, and, through hardware miniaturization, products consisting of either solution can be initially hosted in a laboratory environment. The first is based on a small eight-node blade cluster (a node for each channel of the GAIIx) that can scale out as the number of instruments increase within the environment. In more modest environments, two identically configured generic 16-core servers with 6 TB storage and 48 GB RAM have been utilized to host the computational and storage needs. Additionally, these could serve for scale out through clustering at a later point.

4.4.2 Data Dynamics

Storage and management of these data is arguably the largest issue with which a group will struggle. The principal needs are threefold: scalable, highly dense, and inexpensive disk systems for massive online growth; high-performance disk systems that place the data near to the pipeline algorithms; and archival storage for the data that are required to be kept by the institution. The difficult challenge in building such systems is the dichotomy between being able to handle a very large number of files that are accessed infrequently after primary analysis – with the expectation of online accessibility when the demand arises – and the need to provide high-performance access during analysis. One solution does not fit all requirements. Tradeoffs between inexpensive, highly dense storage using commodity disks and higher cost, highly performing network attached storage (NAS) or storage area network (SAN) systems are dependent upon budget for many facilities. The balance between these is determined by reliability, performance, and budget. Prioritizing dollars can be difficult, but scalable systems that can grow along with storage requirements are most cost-effective for density along with purchasing a small yet high-performance NAS or SAN for transient analytical workloads. Many compromises can be made in the architectures, but we detail all components for completeness. Finally, centralized cyberinfrastructures make economical sense when scaling beyond two instruments and the manufacturers' initial offerings. This is especially true when a bioinformatics group is required to support several disparate scientific groups whose requirements are guaranteed to change as these instruments continue to evolve and new experimental uses for the systems are developed.

High-density storage systems allowing for *ad hoc* growth into the petabyte range exist. These modular yet integrated storage environments provide several hundred terabytes of inexpensive disk provisioned in modules or blocks, aggregated together through software. Based on inexpensive serial advanced technology attachment (SATA) or serial attached SCSI (SAS) disks, both commercial and open solutions are available. Both are based on defined storage modules that can be stacked together over time as storage demands increase. Commercial solutions are usually integrated with software that provides aggregation of disks across the modules into one or a few very large file system namespaces. The open solutions, such as Lustre or GlusterFS, provide the aggregation layer, with commodity storage servers providing the storage blocks. There are additional commodity solutions available based on independent storage servers integrated with open software such as Lustre or GlusterFS. This storage system will capture data while they are being processed through various analysis pipelines. Because the data may only need to exist in this environment during analysis phases, the data itself can be considered transient and temporary within this system. Initially for budget considerations, a small storage footprint could be purchased, enough to house three data runs per instrument (6 TB).

An important consideration for the online, massive storage environment is the length of time necessary for the facility to retain data. A group that understands the institutional requirements of the various sets of data (images, intensities, base pairs, and alignments) can develop reasonable data retention policies. Images, for example, may be retained long enough for primary analysis and QC to complete, then deleted – they may never touch a central file server. In some cases, the cost of the DNA sample and isolation is insignificant to the cost of DNA sequencing such that it will be cheaper to rerun than to store. However, in a clinical setting the DNA sample itself may be unique and therefore priceless, necessitating the need to store much of the upstream data.

Other facilities that serve larger and more diverse communities, operating under defined service levels, may set policies to retain images for a specific period of time – 3 months, for example. In these situations, it will be necessary to initially determine the amount of storage required for 3 months of images and accompanying derived data. In an average three-instrument environment operating during research business hours, this policy would require approximately 65 TB of usable storage, 200 TB if running the instruments at maximum throughput with maximum data capture, probably an unrealistic scenario in practical usage. Adding post-image analysis data, this figure can climb modestly to 75 TB. If images are removed immediately after processing, these figures drop to 10 TB.

Archival needs depend entirely upon the data-retention requirements. It is reasonable to retain all derived data within a terabyte-scale file system. However, due to regulatory or sample cost, it might be necessary to maintain a larger petabyte-scale tape or high-density disk storage system for diagnostics or personalized medicine, for example.

In addition to storage, there are other significant technical considerations that need to be resolved, primarily in networking and routine management of very large file systems. The systems and storage need to be simultaneously connected to several

different networks. These range from institutional LAN connections to private networks. The centralized high-density storage will need to accept data arriving to it via LAN-connected instruments. Additionally, it may need to be connected to private networks serving computational or general-purpose cloud computing environments for further analysis or dissemination of derived information, respectively. A 1 GB network is essential within this environment, with 10 GB networks becoming more prevalent as the demands increase (and cost decreases). Raw network bandwidth, however, can be a small determinant to overall performance. Many technical decisions will be required during design and growth; and with the network interface typically outside the domain of a bioinformatics group, collaboration and careful negotiation, in balance with security, may play a role.

Finally, recovering a very large file system poses some very interesting challenges that certain IT vendors are addressing. A file system check on several hundred terabytes may require weeks to perform.

4.4.3 Software and Post-analysis

This area is by far the most rapidly evolving and most critical to providing useful information from these instruments as well as managing lab processes and data management of the raw and derived data. Software and informatics pipelines for principal analysis and visualization are in rapid development from both commercial sources and from the academic community.

The early adopters of these technologies, the very large sequencing centers, and later the medium-sized Core Facilities, understand the challenges they face with instruments of this type. The immediate challenge comes with a lack of adequate vendor-supported software and Laboratory Information Management Systems (LIMS). Early-stage groups rely heavily on custom-developed LIMS and informatics platforms. Given the tremendous cost and complexity of developing commercial-class LIMS modules with adequate flexibility built into the system for integration to internal business processes across many organizations, most instrument manufacturers do not provide such systems. However, some do provide an API or Web service interface to their software.

To the small- and mid-sized groups, however, this is a very large gap in support, but that gap is shrinking. There exists a plethora of workflow applications, algorithms, and analysis pipelines in the public domain as well as commercial products coming to market. It is not reasonable to attempt to summarize all the available software offerings, but, through Internet resources, other blogs, and the recent flurry of new publications within the scientific and informatics literature, more than enough information is available (Dooling 2008).

For the purposes of this perspective, the critical area for a group will be in the integration of the principal analysis pipelines with data management and information delivery systems within organizations (Fig. 4.2). Groups are tasked with delivering data to research projects for additional analysis. The format of the data delivered

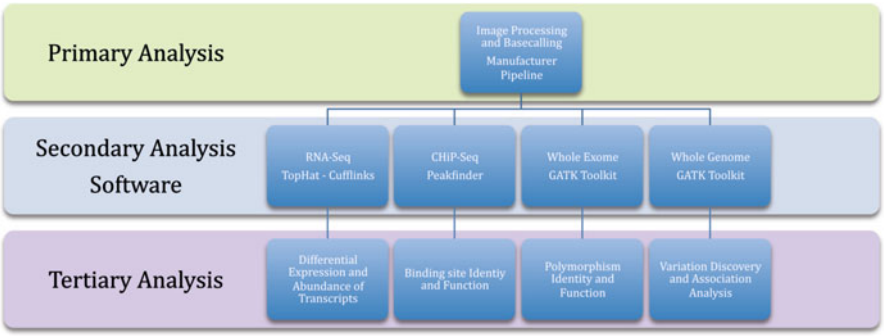

Fig. 4.2 Analysis Workflow. Workflow of analysis from primary analysis to tertiary functional analysis

range from short sequence reads to sequence that has been aligned to a reference. As the data volumes increase, there will be a greater demand on groups to fundamentally understand the uses of these machines in research so as to deliver the data in more useful ways other than raw sequence. Assignment of biological function and annotation of the sequence with features of interest will still be critical tasks. The methods to perform these tasks are still in the initial phases of development, with a few tools showing early promise (Galaxy: http://main.g2.bx.psu.edu).

As the cost of sequencing continues to decline, these technologies will translate into clinical settings, where the integration of this information with enterprise and personalized medical records, sample repositories, and knowledge management systems within medical institutions will be an absolute requirement to healthcare delivery and diagnostics. Other research environments are likely to encounter similar challenges soon.

4.4.4 Staffing Requirements

There are many challenges in integrating next-generation sequencing instruments into the information technology infrastructure. Along with technology considerations, it is additionally critical to have a well-trained cadre of bioinformatics specialists operating within the group, accessible to the entire institution in order to best serve the needs of those using this new technology. If the Core Facility has expertise in IT or can leverage other institutional resources for architecting and managing the IT systems described, then much of the operational work will involve bioinformatics analysis and systematizing the infrastructure. Specifically, these involve optimizing data analysis pipelines in the parallel computing environment, automating bulk transfers of large volumes of data, filtering data and assigning biological significance, interacting with investigators to understand the purpose of sequencing projects, and the ability to suggest analysis methods to investigators.

The skills necessary within the Facility include the following:

1. An intimate knowledge of UNIX-based operating systems.
2. Understanding of a scripting language such as Perl.
3. An understanding of parallel computing environments for UNIX clusters.
4. Knowledge of network-based data storage.
5. General knowledge of biology and genome sciences.
6. Ability to derive data analysis and software requirements from investigators who do not have a sophisticated understanding of information technology.
7. Ability to develop software encapsulating new analysis methods.
8. Understanding of relational databases and database architecture.
9. Ability to seek out and test novel bioinformatics software and analysis routines.

Finding a single staff member with all these skills would be extremely difficult, but finding members who have a subset of these skills and overlapping them in a team will be a more reasonable prospect. Individuals with these skill sets are rare and demand for their services is high, so compensation for such individuals is above that of laboratory technicians and bioinformaticians who have not operated in a high-performance computing environment. As such, a significant portion of the total cost of ownership for a next-generation sequencing operation will comprise staff member salaries.

4.5 Applications of the Infrastructure

Applications of the infrastructure described above would be limited to next-generation sequencing technology. While it is applicable to all sequencing and genotyping, it also holds for the majority of applications of next-generation sequencing technology.

4.6 Perspectives

The application of next-generation sequencing technologies have led to a rapid increase in the amount of base pairs per experiment. It also resulted in a similar increase in the amount of data generated by these experiments. This data volume requires a new cyberinfrastructure in the majority of institutions to handle this increased data load. New sequencing technologies such as single molecule sequencing and nanopore technologies will only increase this data volume. The need to capture, store, transport, and analyze these data will require an increasing commitment to the information technology that underpins its operation. This commitment is extensive and will rival the cost of the sequencing experiment as those costs are decreasing rapidly. It is expected that new methods and algorithms will be developed for analyzing these data, as the current tools are crude and much insight is lost in the current methods of analysis. This analysis will become increasingly automated as

new methods are developed. This will require a robust storage system in order to allow new applications to be applied to older data. Overall, the investment in information technology infrastructure will be critical and will be the only way to effectively preserve and rapidly analyze sequence data.

Acknowledgments DS acknowledges the Vanderbilt Center for Human Genetics Research.

Reference

Dooling D, 2008 Dec 4. *Next Generation Sequencing Informatics Table*. http://www.politigenomics.com/next-generation-sequencing-informatics.

Chapter 5
Base-Calling for Bioinformaticians

Mona A. Sheikh and Yaniv Erlich

Abstract High-throughput platforms execute billions of simultaneous sequencing reactions. Base-calling is the process of decoding the output signals of these reactions into sequence reads. In this chapter, we detail the facets of base-calling using the perspective of signal communication. We primarily focus on the Illumina high-throughput sequencing platform and review different third-party base-calling implementations.

5.1 Introduction

Over the last several years, we have observed a Moore's law-like trajectory in the power of high-throughput sequencing platforms. Output volumes have increased several-fold, effective sequencing read lengths have grown, and error rates have dropped significantly. As a result, these platforms have become the ultimate backend for biological experiments. High-throughput sequencing has been harnessed in a variety of applications beyond conventional genomic sequencing tasks, including expression analysis (Wang et al. 2009), profiling of epigenetic markers (Lister et al. 2011), and the spatial analysis of the genome in 3D (Lieberman-Aiden et al. 2009).

In essence, a DNA sequencing platform is a communication device. It takes an input library of DNA molecules, a set of messages, encodes the DNA using chemical reactions and imaging, and conveys the results to a base-caller algorithm. The task of the base-caller is dual: recover the original DNA sequence from the chemical reactions and report a quality score that reflects the confidence in each called nucleotide. The similarities between sequencing and communication enable the development

M.A. Sheikh • Y. Erlich (✉)
Whitehead Institute for Biomedical Research, 9 Cambridge Center,
Cambridge, MA 02142, USA
e-mail: yaniv@wi.mit.edu

N. Rodríguez-Ezpeleta et al. (eds.), *Bioinformatics for High Throughput Sequencing*,
DOI 10.1007/978-1-4614-0782-9_5,

and analysis of base-calling strategies using powerful tools from signal processing and information theory.

Base-calling software is usually provided by the manufacturer of the sequencing platforms. However, various third party groups, mostly from academia, have also made efforts to develop enhanced base-calling algorithms. The most notable example is the enhanced ABI base-caller, Phred, which played a pivotal role in the Human Genome Project (Ewing and Green 1998; Ewing et al. 1998). The sheer amount of data in high-throughput sequencing means that even a slight improvement in base-calling translates to millions more correct nucleotides, benefiting downstream applications such as SNP calling, methylation profiling, and genome assembly. Besides enhancing the sequencing results, developing base-calling algorithms provides insight into the inner workings of sequencing platforms and the fundamental principles and challenges of the technology.

Currently, there are ten high-throughput sequencing platforms: 454 (Roche), Illumina, SOLiD (Life technologies), Pacific Biosciences, Ion Torrent (Life Technologies), DNA Nanoball Arrays (Complete Genomics), Polonator, Heliscope (Helicos), Oxford Nanopore, and Nabsys (manufacturers are in parentheses; the last five platforms are not commercially available). Of these technologies, the Illumina platform has become the leading platform for high-throughput sequencing (see a map of installation of high-throughput sequencing platforms here: http://pathogenomics.bham.ac.uk/hts/). The HiSeq2000 platform has the highest throughput and the lowest cost-per-nucleotide in today's market. Furthermore, the platform has received maximal attention from the community developing base-calling strategies. Therefore, we chose Illumina to illustrate the informatics challenges of base-calling. The interested reader can find information specific to 454 base-calling in Quinlan et al. (2008) and SOLiD base-calling in Wu et al. (2010).

5.1.1 Illumina Sequencing

The working concepts of Illumina sequencing are reviewed in detail in Chap. 2, and also by Bentely et al. (2008) and Metzker (2010). Here, we provide a brief overview of Illumina sequencing, while elaborating on details critical to base-calling.

The Illumina platform employs cyclic reversible termination (CRT) chemistry for DNA sequencing. The process relies on growing nascent DNA strands complementary to template DNA strands with modified nucleotides, while tracking the emitted signal of each newly added nucleotide. Each nucleotide has a 3′ removable-block and is attached to one of four different fluorophores depending on its type. Sequencing occurs in repetitive cycles, each consisting of three steps: (a) extension of a nascent strand by adding a modified nucleotide; (b) excitation of the fluorophores using two different lasers, one that excites the A and C labels and one that excites the G and T labels; (c) cleavage of the fluorophores and removal of the 3′ block in preparation for the next synthesis cycle. Using this approach, each cycle interrogates a new position along the template strands.

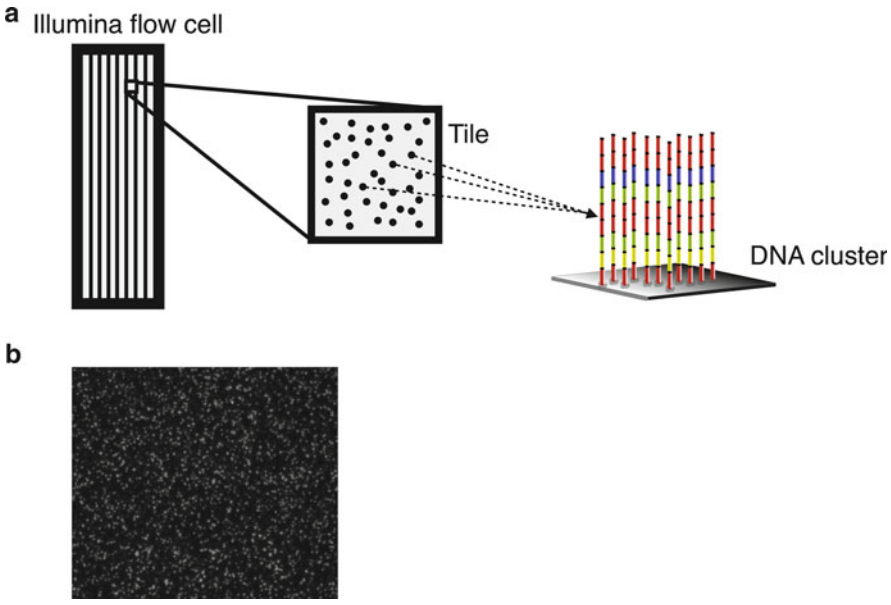

Fig. 5.1 Physical hierarchy in Illumina sequencing. (**a**) The Illumina flowcell contains eight lanes which are further broken down into tiles. Each tile contains clusters of identical DNA fragments. (**b**) A cropped section of an Illumina tile image. *White spots* are DNA clusters

The tremendous power of the Illumina platform stems from its ability to simultaneously execute and sense billions of CRT reactions. The sequencing process occurs in a *flow cell* – a small glass slide that holds the input DNA fragments in fixed and distinct positions during the sequencing process. The flow cell consists of eight chambers called *lanes*. The lanes are physically separated from each other and may contain different sequencing libraries without sample cross-contamination. The imaging device cannot capture an entire lane in a single snapshot. Instead, it takes snapshots at multiple locations along the lanes called *tiles*. There are 100 tiles per lane in Genome Analyzer II and 68 tiles per lane in HiSeq2000. A tile holds hundreds of thousands to millions of *DNA clusters*. Each of these clusters consists of approximately one thousand identical copies of a template molecule. The DNA clusters are constructed prior to the sequencing run by bridge amplification of the input library. The purpose of the amplification is to increase the intensity level of the emitted signal since the imaging device cannot reliably sense a single fluorophore. However, the physical distance of the DNA fragments within a cluster is below the diffraction limit, permitting the imaging device to perceive the fragments as a single spot. Figure 5.1 illustrates the physical hierarchy of the Illumina platform.

The output of a sequencing run is a series of images each depicting the emission of millions of DNA clusters for a specific combination of lane, tile, cycle,

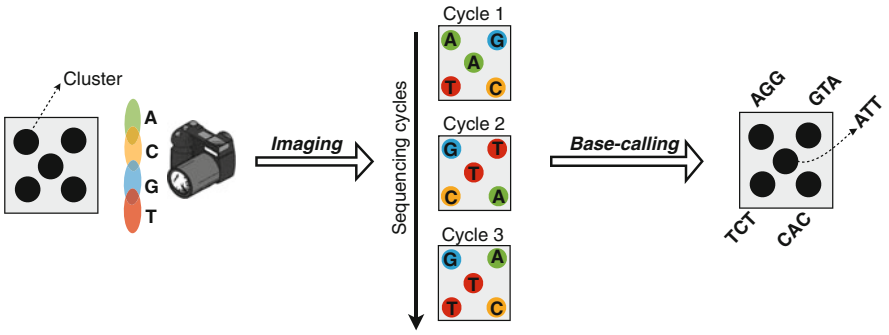

Fig. 5.2 A toy example of base-calling with Illumina data. Five DNA clusters (*left*) are subject to three cycles of sequencing (*middle*). Four images are recorded in each cycle, one for each fluorescence channel; for clarity of illustration, these are consolidated into a single tile image per cycle. Base-calling can be done by tracking the fluorophore signals from each cluster (*right*)

and fluorophore. These images comprise the inputs to the base-calling algorithm. As shown in Fig. 5.2, by tracking the emission signal from a cluster across cycles, a base-caller can recover the input DNA template.

5.2 Analyzing the Illumina Sequencing Channel

The sequencing process above describes the ideal situation without any noise or signal distortion. In such a situation the base-caller's task would be simple: identify the base according to its fluorophore color. In reality, however, as with any communication procedure, the sequencing signal is subject to noise and distortion due to imperfections in the chemical reactions and imaging procedure. The first challenge in devising a robust base-calling algorithm is to determine the *channel model* that describes the factors that distort the sequencing signal.

5.2.1 General Terminology of Distortion Factors

Information theory provides several useful classifications for signal distortion (Kailath and Poor 1998). First, a distortion factor can have a deterministic or stochastic effect on a signal. For instance, yield differences between the four fluorophores can create a deterministic effect on the sequencing signal; the number of transmitted photons from a single DNA cluster is a stochastic process that follows a Poisson distribution.

Second, a distortion factor can be stationary or nonstationary. A stationary distortion has the same characteristics during the sequencing process, whereas a nonstationary distortion will evolve from cycle to cycle. An example of a stationary distortion

factor is additive Gaussian noise. An example of a nonstationary distortion factor is the progressive loss of template DNA strands in each cycle creating variable signal decay. Stationary and nonstationary distortions hamper the recovery of the original sequence. Additionally, nonstationary distortions escalate with successive sequencing cycles and limit the useful sequence reading length.

Third, a distortion factor is classified according to variations in its input parameters. At one extreme are run-independent factors that have the same characteristics irrespective of the sequencing run. Otherwise, a distortion factor can be run-dependent, lane-dependent, or tile-dependent. An example of a run-dependent distortion is variability in the manufacturing quality of the nucleotides. Distortions that vary within a narrow scope (lane or tile) increase the number of parameters in the channel model and are harder to definitively model. A special case of this category is sequence-dependent distortion that creates statistical dependency between the distortions of different cycles. In this case, an accurate characterization of the distortion can only be done during the actual base-calling.

Fourth, a distortion factor can either have a memory or be memoryless. Memoryless distortion means that the degradation is only influenced by the interrogated nucleotide. A distortion with memory means that the degradation is due to residual signals from other nucleotides in the physical vicinity of the interrogated nucleotide.

5.2.2 Constructing a Sequencing Channel Model

In-depth knowledge about the sequencing chemistry and imaging process of a platform provides an initial hypothesis for the possible distortion factors affecting it. The primary challenge in dissecting a sequencing channel is in finding the right setting to isolate and measure its distortion factors. This is especially hard for third-party researchers compared to platform manufacturers, since the former do not have full access to the building blocks of the sequencing chemistry and are unaware of the finer details of platform operation. The best option, therefore, is to conduct controlled sequencing experiments and observe intermediate products of the platform.

In the next two subsections, we first discuss different intermediate types of data output from an Illumina sequencer, and then present strategies for controlled experiments to analyze the sequencing channel.

5.2.2.1 Outputs of Illumina Sequencing

Illumina provides a large number of intermediate files that can be used to perform a fine-scale analysis of distortion factors. The most useful among them are imaging files, intensity files, and sequencing files.

Imaging files are the immediate output of the imaging device. As such, they represent a level of pure, raw data prior to any processing. Illumina currently uses

16-bit TIFF images of size 2,048 × 1,794 pixels in GAII and 2,048 × 2,816 pixels in HiSeq2000. A single-end (SE) run with 100 cycles on a GAII produces 3.2 million images. This translates into tens of terabytes of imaging information. Therefore, it is a challenge to collect the entire set of photos for further analysis. The platform currently stores photos only from the first few cycles.

Illumina's intensity files are the output of its Firecrest demultiplexing software. The intensity files contain the position of each DNA cluster in a "lane–tile–X coordinate–Y coordinate" format, along with a matrix of the cluster's four imaging channel intensities in every cycle. The intensity values show a linear correlation with the raw data in the imaging files.

Illumina's sequencing files are the product of its base-calling algorithm, Bustard. The files are in a FASTQ format that describes the sequencing results of each DNA cluster in four lines. The first line starts with an "@" sign and contains a unique tag name for the cluster in the form: sequencing instrument : flow-cell lane : tile number in lane : x-coordinate of cluster on tile : y-coordinate of cluster on tile : index number for multiplexed sample : member of pair. For example, @ SEQUENCERNAME : 1 : 1 : 1029 : 21234#0/1 could be a valid ID for a sequencing cluster. Since the tag name preserves the lane, tile, X, and Y positions of a cluster, it is possible to link an entry in the intensity file to an entry in the sequence file. The second line contains the sequence read. The third line starts with a "+" sign and contains the unique tag name again. The fourth line shows the *quality score* of each base, thus providing an estimate of the base caller's confidence in the called nucleotide. In newer versions of the pipeline (v.1.3 and beyond), the reported value uses ASCII encoding to report the Phred quality score, and is given by: $-10log10\ p+64$, where p is the confidence of the base caller. The allowed values are between 66 and 126.

5.2.2.2 Deriving a Channel Model from Controlled Experiments

Impulse Response Analysis

Impulse response analysis is a powerful signal processing tool for characterizing distortion in communication channels (Shenoi 2006). A sharp pulse, called a *delta function*, is input to the system under investigation. The output, measured as a function of time, reflects the system's transfer function, and thus the characteristics of the channel. In the case of a stationary distortion, the structure of the output pulse should remain the same irrespective of the time point at which the delta function is input. Therefore, it follows that a change in the structure of the pulse with injection time indicates the presence of nonstationary noise factors. The width of the pulse in the output signal reveals the memory of the channel. If the channel is memoryless, the pulse should have the same width in both the input and output. However, if the channel has some memory, the width of the output pulse will reflect the memory window of the channel. By varying the amplitude of the impulse, we can learn whether the channel is linear and also find its dynamic range.

In sequencing, a delta function corresponds to sequences in which a single type of nucleotide resides within a long stretch of other nucleotide types. For instance, the sequence …AAAAATAAAAA… can be thought of as a T-nucleotide delta function in a homogenous background. The sequence …ACGGCAATCCGGAA… can be thought of as a T-delta function in a heterogeneous background. We can sequence such fragments where the T nucleotide occurs at different positions and track the output signal in the four fluorophore channels. In an ideal channel – both memoryless and stationary – the energy of the T channel would only increase when the T nucleotide is interrogated, and the output would remain the same irrespective of the T's position in the sequence or the type of nucleotides neighboring it.

Impulse response analysis may use more complicated inputs to examine specific distortion build-ups. For example, one can sequence alternating homopolymer repeats such as GGGGGGAAAAAAAGGGGGGGAAAAAAA… Such repeats expose residual nucleotide-specific signal build-up that might not be detectable with short, single-nucleotide impulses. Another example is a dinucleotide tandem repeat chain, like ACACAC.... Dinucleotide tandem repeats are relatively immune to distortions with memory. The output signal will always converge to one half, regardless of the size of the memory. Any deviation from a half indicates the presence of other noise factors.

Impulse response analysis necessitates the construction of accurate DNA impulse sequences. One solution is to sequence a rich genomic library and restrict analysis to DNA fragments that inherently contain delta function sequences. Another solution is to artificially synthesize and sequence a set of DNA oligonucleotides. Unfortunately, DNA synthesis followed by PCR-based amplification has a high error rate for templates with homopolymer sequences or tandem repeats. As an alternative, bacterial cloning can be used to select and accurately amplify such difficult templates for impulse response analysis (Erlich et al. 2008).

Genomic Library Analysis

Another useful strategy to reveal distortion factors is the analysis of sequencing data from genomic libraries with an accurate reference genome. Genomic libraries complement impulse response analysis because of the natural input signals they provide, ensuring a realistic setting for analyzing the channel model. Moreover, the massive amount of data from genomic libraries reduces the risk of overfitting when evaluating complex models.

Sequencing the genome of the Φx-174 virus is particularly useful for channel model analysis. First, the genome length of the virus is only 5.5 kb, enabling ultra-fast alignment to the reference genome, even for noisy sequence reads with a large number of mismatches. Second, the GC content of the Φx-174 genome is 44%, similar to the 46% GC content of the human genome (Romiguier et al. 2010), creating a realistic situation for modeling nucleotide-dependent distortions in human libraries. Third, according to Illumina's product guidelines, one lane in each flow cell must contain the DNA library of the Φx-174 virus as a control in order to keep the warranty valid for that sequencing run. This allows the tracking of distortion in a large number of Φx-174 samples across different sequencing runs.

5.3 Major Signal Distortion Factors in Illumina Sequencing

Here we detail the major distortion factors in Illumina sequencing. We restrict our review to factors that appear after the image analysis step. The interested reader can find more information about challenges in the image analysis step in Whiteford et al. (2009) and Kriseman et al. (2010).

5.3.1 Fluorophore Crosstalk

In an ideal CRT reaction, the four fluorophores would have distinct emission spectra and similar yields. However, the emission spectra of the fluorophores used for sequencing are broad and overlap with one another (Fig. 5.3a). Thus, when one fluorophore is excited, its signal also passes through the optical filters of the other channels. Fluorophore crosstalk is not unique to Illumina sequencing, but is also found in other sequencing platforms (Li and Speed 1999) and flow cytometry systems (Sklar 2005).

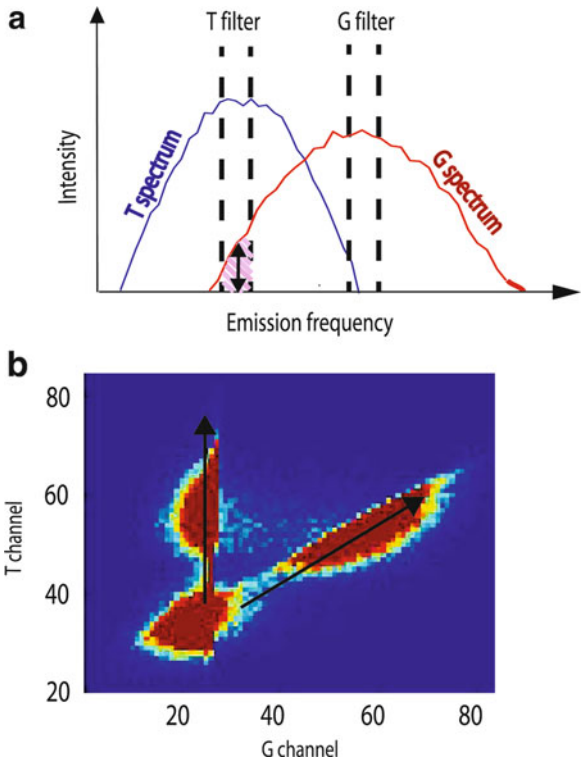

Fig. 5.3 Crosstalk creates a deterministic distortion. (**a**) An illustration of the crosstalk phenomenon. The spectrum of the G fluorophore (*red*) bleeds into the T filter (*pink hatched region*). As a result, a T signal will also be detected when a G fluorophore is excited. (**b**) A two-dimensional histogram of real intensity data from Illumina. The G fluorophores (*right arrow*) strongly transmit to the T channel. On the other hand, the T fluorophores (*left arrow*) do not transmit to the G channel

Fluorophore crosstalk is a deterministic distortion factor that is represented by the 4×4 matrix G. The ijth element in G denotes the emission of the jth fluorophore in the ith channel. An example of typical values of the matrix is (small values were rounded to zero for simplicity):

$$G=\begin{bmatrix}1.24 & 0.20 & 0.00 & 0.00\\ 0.71 & 0.72 & 0.00 & 0.00\\ 0.00 & 0.00 & 1.35 & 0.00\\ 0.00 & 0.00 & 0.73 & 1.00\end{bmatrix} \tag{5.1}$$

Note that there is strong crosstalk between the "A" and the "C" channels, and the "G" and "T" channels (Fig. 5.3b) – each pair of fluorescence channels is excited by the same laser.

5.3.2 Phasing

In the ideal situation, the lengths of all nascent strands within a DNA cluster would be the same. Imperfections in the CRT chemistry create stochastic failures that result in nascent strand length heterogeneity, which is referred to as phasing. Phasing reduces the purity of the signal from an interrogated position by contamination from signals from *lagging* or *leading* strands. A failure to remove the 3′ block or to incorporate a new nucleotide creates a lagging nascent strand, shorter than its counterparts. In addition, a small fraction of input nucleotides do not have a 3′ block due to manufacturing imperfections. Incorporation of a block-free nucleotide will cause a nascent strand to race ahead, become too long and lead the other strands in phase. Phasing is a nonstationary distortion. Length heterogeneity escalates with each cycle, lowering the precision of base-calling, and limiting the length of useful sequence reads (Fig. 5.4a).

Phasing is essentially a random walk process given by the following equation, where t cycles have elapsed and n is the strand length:

$$P(t,n)=\int_{-\pi}^{\pi}\left[1-p_1+\frac{p_1p_2e^{i\omega}}{1-(1-p_2)e^{i\omega}}\right]^t e^{-i\omega n}\frac{\mathrm{d}\omega}{2\pi} \tag{5.2}$$

where p_1 denotes the probability of removing the 3′ block and p_2 denotes the probability of incorporating a single blocked nucleotide. Thus, with probability $1-p_1$, the nascent strand does not grow in a given cycle, creating a lag; with a probability of $p_1\times p_2$, the nascent strand grows with exactly one nucleotide, representing a successful CRT reaction; with a probability of $p_1\times(1-p_2)\times p_2$, the nascent strand grows with two nucleotides, creating a racing strand, and so on. P is a $T\times T$ matrix corresponding to a total of t cycles, whose ijth element describes the probability that a nascent strand is i nucleotides long after j cycles. In the ideal situation with no phasing imperfection ($p_1=1$ and $p_2=1$), P is the identity matrix (Fig. 5.4b).

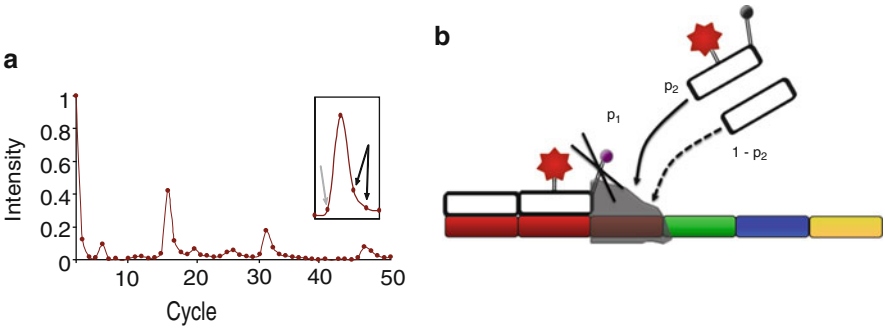

Fig. 5.4 Phasing induces a nonstationary distortion. (**a**) The intensity output of DNA fragments with "C" impulses every 15 cycles in a heterogeneous background. Notice the anticipatory (*gray arrow*) and memory signals (*black arrows*) due to phasing. (**b**) A molecular representation of the random walk model. The 3′ block is removed with a probability of p_1 and a blocked nucleotide is added with a probability of p_2. *Purple ball* – 3′ block, *gray shape* – DNA polymerase, *Red star* – fluorophore, *colored box* – template strand, *open box* – nascent strand

We collected data on the values of p_1 and p_2 from hundreds of Illumina GAII runs. We found that the mean block removal probability (p_1) is around 0.994 and the mean blocked nucleotide incorporation probability (p_2) is around 0.996. Using these values, P forms a band-diagonal matrix, where the bulk of the energy after 90 cycles is within a window of five nucleotides around the interrogated position – but fewer than half the nascent strands display the length that corresponds to the interrogated position.

5.3.3 *Fading*

The sequencing process takes several days. During that time, the DNA strands are washed excessively, exposed to laser emissions that create reactive species, and are subject to harsh environmental conditions. All of these lead to a gradual loss of DNA fragments in each cluster, decreasing its fluorescent signal intensity – a process termed fading (Fig. 5.5a).

Fading follows an exponential decay that is given by the following equation (see also Fig. 5.5b):

$$D(i,j) = \begin{cases} e^{-p_3 t} & i = j \\ 0 & \text{otherwise} \end{cases} \tag{5.3}$$

where D is a T-by-T diagonal matrix that represents the exponential decay of the signal, t denotes the number of elapsed cycles, and p_3 denotes the decay rate. A reduction in decay rate is one of Illumina's major sequencing improvements over

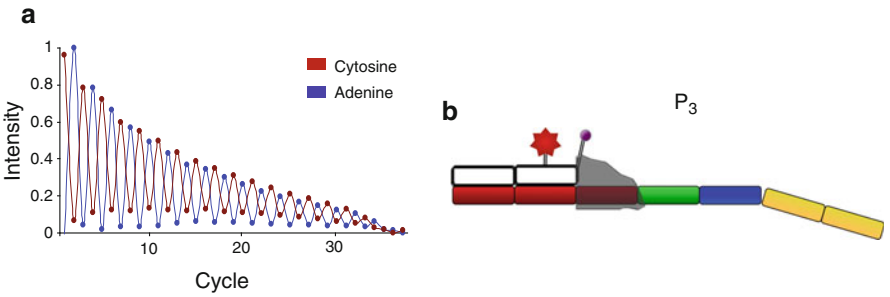

Fig. 5.5 Fading. (**a**) The intensity values of DNA fragments with AC microsatellites show exponential decay (data were obtained from a GAI platform). (**b**) A molecular model of fading. A DNA strand is washed away with a probability of p_3. *Purple ball* – 3′ block, *gray shape* – DNA polymerase, *Red star* – fluorophore, *colored box* – template strand, *open box* – nascent strand

the years. In the GAI, the decay rate was around 3–5% in each cycle, in the GAII it dropped to 1%, and in the HiSeq2000, a decay rate of 0.7% was measured, meaning that the average half-life of a DNA strand is now around 90 cycles.

5.3.4 Insufficient Fluorophore Cleavage

Multiple lines of evidence have shown that even after the correction for the distortion factors above, there is still a residual signal that looks like a change in the crosstalk matrix over cycles (Fig. 5.6a). This distortion, which escalates with time, creates a strong bias toward a specific nucleotide and reduces the performance of base callers in later cycles (Fig. 5.6b). The exact characteristics of this noise factor have yet to be fully analyzed. However, it has been suggested that the residual signal is caused by imperfect fluorophore cleavage. In every cycle, a small fraction of fluorophores is left behind (Fig. 5.6c), creating sequence-dependent distortion. Different types of fluorophores can have different cleavage probabilities. If one fluorophore is more "sticky" than the others, it can create an overall calling bias toward this nucleotide that will escalate with time.

In our preliminary results with a single GAII run, we found fluorophore cleavage rates of 99.62%, 99.62%, 99.14%, and 99.61% for the A, C, G, and T fluorophores, respectively. Indeed, that run showed a base-calling bias toward the G nucleotide (data not shown). Kao et al. (2009) have calculated the overall residual signal that can be attributed to insufficient fluorophore cleavage for all four channels. Their results show a linear increase in the residual signal, as expected, from insufficient fluorophore cleavage. Furthermore, the increase in the residual signal matches our 99.6% success rate in fluorophore cleavage, providing further support for the characterization of this distortion factor.

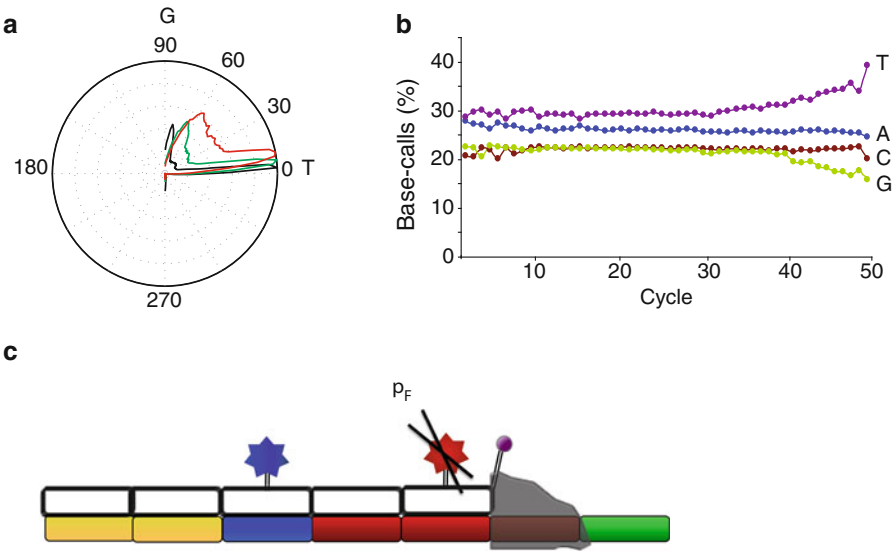

Fig. 5.6 Insufficient fluorophore cleavage. (**a**) The polar histogram displays the ratio between the G and the T channel after crosstalk correction. A strong G signal with a weak T one corresponds to bins that are close to 90 degrees, and the opposite occurs close to zero degrees. In the first cycle (*black*), the two lobes are orthogonal which indicates correct crosstalk correction. In later cycles (*green* and *red*), the G lobe starts to migrate toward the T lobe, and creates a cycle-dependent residual signal. (**b**) The percentage of called bases in the phi-X library is plotted as a function of cycle number using the Illumina base caller. The T and the G calls have strong opposite trends. (**c**) A molecular model for insufficient fluorophore cleavage. In each cycle, the last fluorophore is cleaved with a probability of p_F that depends on the fluorophore type. *Purple ball* – 3′ block, *gray shape* – DNA polymerase, *Red star* – fluorophore, *colored box* – template strand, *open box* – nascent strand

The probabilities of fluorophore cleavage are described by the diagonal 4×4 matrix F. Each element along the diagonal denotes the probability of cleaving a different fluorophore.

5.3.5 *Overall Model*

The overall intensity distorted by crosstalk, phasing, fading, and insufficient fluorophore cleavage for a given DNA sequence can be formulated by the following model (Valente et al. personal communication). The left term of (5.4) represents the distortion without insufficient fluorophore cleavage and the right term adds its effect.

$$I \approx \mathrm{DPSG} + \mathrm{DQSFG} \tag{5.4}$$

where I is a T×4 intensity matrix that denotes the signal received in each optical filter for a total of T sequencing cycles; ≈ denotes equality up to a normalization

constant; S is a $T \times 4$ binary matrix that denotes an input DNA sequence of length T. Its columns correspond to A, C, G, T respectively from left to right. For instance, the DNA sequence "ATC" is given by:

$$S = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

Q is a $t \times t$ matrix given by:

$$Q(m,t) = \sum_{n>m} P(n,t) \tag{5.5}$$

Q represents the cumulative probability that a nascent strand is above a certain length; G, F, P and D are defined previously in this chapter.

5.4 Decoding Algorithms

The overall goal of the base-caller is to infer the matrix S given the intensity output of I in the presence of distortion factors. This task is a classification problem. We describe in chronological order, different successful solutions to base-calling, contrasting the general themes of each approach. Ledergerber and Dessimoz (2011) provide an excellent review of different available base-calling software that includes performance comparisons.

5.4.1 Alta-Cyclic

The Alta-Cyclic algorithm (Erlich et al. 2008) relies on supervised learning and a combination of parametric and ad-hoc modeling. The algorithm starts the learning step by inferring the phasing parameters. This it does so by performing a grid search over possible values of p_1 and p_2 (notation from 5.3.2). At each (p_1, p_2) coordinate, a phasing matrix is calculated, and the intensity data are deconvolved by multiplying I with $P+$, the pseudo-inverse matrix of P. The algorithm calls the bases of the last cycles and determines the total error rate. The (p_1, p_2) coordinate that displays the lowest error rate is selected. Alta-Cyclic does not infer the other parameters of the model. Instead, it trains a support vector machine (SVM) in each cycle to find the margins in the intensity space that optimally discriminate between the nucleotides. The output of the training step is an optimized P matrix and a set of trained SVM machines.

In the calling step, the algorithm deconvolves the phasing distortion and uses the SVMs to classify the nucleotides. The output of the data is in FASTQ, and a quality score is assigned based on the distance of the intensity vector from the SVM margins.

Alta-Cyclic is available from http://hannonlab.cshl.edu/Alta-Cyclic/main.html.

5.4.2 Rolexa

The Rolexa base-caller (Rougemont et al. 2008) relies on unsupervised learning and a parametric model. Rolexa forces the crosstalk matrix to be a block diagonal matrix, where the only nonzero entries are the crosstalk between the A and C channels and the G and T channels. In order to deal with the residual signal, Rolexa generates a new crosstalk matrix in each cycle. In addition, it uses a simplified version of the phasing matrix that only allows lagging strands ($p_2 = 1$). After learning the cycle-dependent crosstalk and phasing, the intensity data are deconvolved and ready for base-calling.

In the calling step, the algorithm uses the expectation-maximization (EM) algorithm to classify the intensity data. It takes data from one or a few tiles, and iteratively fits four Gaussians to the intensity data, with each Gaussian corresponding to a different type of nucleotide. The distance between the intensity vector and the Gaussians gives the uncertainty about calling the base and can be used to calculate the quality score.

A useful feature of Rolexa is that it can report the output sequences in an IUPAC code format. For instance, if there is high uncertainty whether a base is "A" or "C," the caller will report "M." IUPAC format gives richer information than the FASTQ format, since data about the second most likely base are preserved.

Rolexa is available under the GNU Public License (GPL) from http://www.bioconductor.org/packages/2.5/bioc/html/Rolexa.html.

5.4.3 Swift

Swift (Whiteford et al. 2009) is a freely available base-calling pipeline that performs both image analysis and base-calling. It relies on unsupervised learning and a parametric model for base-calling and supervised learning and an ad-hoc model for quality score calculation.

The parametric model for base-calling only includes crosstalk and phasing correction. The crosstalk estimation relies on the Li and Speed (1999) method that uses L1 regression. The phasing model is slightly different from the random walk model above and it assumes that a strand cannot grow by more than two nucleotides in a given cycle. In order to find the phasing parameters, the algorithm selects a subset of 400 bright and clean clusters. Then, it scans the clusters and looks for signals that resemble an impulse shape – a strong peak in one channel that does not appear in adjacent cycles. The ratio between this peak and the previous and subsequent cycles gives p_1 and p_2.

Base-calling is achieved by multiplying the intensity matrix by the inverse of the crosstalk and phasing matrices and selecting the channel with the highest energy. In order to calculate the quality scores, the algorithm first determines the ratio of the maximal intensity to the sum of intensities. Then, it aligns reads to a reference genome and builds a lookup table that matches the intensity ratio to the probability of error.

Swift is available under the LGPL3 from http://swiftng.sourceforge.net.

5.4.4 *BayesCall, NaiveBayesCall*

BayesCall (Kao et al. 2009) and NaiveBayesCall (Kao and Song 2011) algorithms rely on unsupervised learning and a full parametric model. A notable advance is the use of Bayesian networks for base-calling.

In the context of the channel model above, BayesCall takes into account crosstalk, phasing, fading, and insufficient fluorophore cleavage. It assumes that all factors are tile-specific. As in Swift's model, phasing is restricted to a maximal growth of two nucleotides per cycle. BayesCall sets a more complicated fading distortion that depends on two parameters: (a) a tile-specific, cycle-specific parameter that models fluctuation in fading due to extrinsic factors such as room temperature, and (b) a cluster-specific, cycle-specific term that represents the number of active fragments in the decoded cluster. In addition, the algorithm assumes a cycle-specific residual signal, which is reminiscent of insufficient fluorophore cleavage.

In order to infer the parameters, BayesCall uses a combination of methods: direct computation for the tile-specific, cycle-specific fading, interior point method for phasing, and EM for the rest of the parameters, except for the cluster-specific fading term which is determined during base-calling.

In BayesCall, base-calling is achieved by simulated annealing, a heuristic that enables convergence to a (near) optimal solution when the search space is overwhelming. The problem with this approach is that it requires at least 10,000 iterations for a single call. The NaiveBayesCall algorithm is highly similar to BayesCall. It relaxes the fading structure and uses a faster heuristic for base-calling. First, it obtains an initial guess for the sequence by deconvolving the effect of phasing and crosstalk and finding the strongest intensity. This is used to reduce the search space. Second, it iteratively calls a base, finds the most likely residual signal, and propagates the information to the next cycle.

Both algorithms report quality scores utilizing exact calculations that are derived from their underlying probabilistic models.

BayesCall is available under the GPL from http://www.cs.berkeley.edu/yss/bayescall/. NaiveBayesCall is available under the GPL from http://bayescall.sourceforge.net

5.4.5 *Ibis*

Ibis (Kircher et al. 2009) is a fully ad hoc, supervised learning base-calling algorithm. It does not attempt to evaluate any of the parameters in (5.4) but uses an array of SVM machines, each of which corresponds to a different cycle. The input of each SVM is a 12-feature vector that consists of intensity values from the previous cycle, current cycle, and following cycle. During the training stage, the SVMs look for the best decision boundaries between the bases. Since data from the previous and following cycles are incorporated, phasing and fluorophore cleavage are implicitly taken into account in the ad hoc model.

The calling stage of Ibis uses the trained SVM to classify the raw intensity data. Quality scores are calculated according to the distance of the intensity vector from the decision boundaries.

Ibis base-calling software is available under GPL from http://bioinf.eva.mpg.de/Ibis/.

5.4.6 TotalReCaller

TotalReCaller algorithm is a work in progress by Bud Mishra's group at the Courant Institute. It has three unique features: first, the algorithm couples together base-calling and alignment. The iterative alignment is used to generate prior probabilities about subsequent nucleotides and enhance the calling. Second, the algorithm does not call each base individually, but evaluates several bases together. As mentioned above, the residual signal due to insufficient fluorophore cleavage is sequence-dependent. This creates statistical dependency between the distortions in different cycles. Considering several bases at once can therefore boost the accuracy of the decoding. Third, the algorithm has been successfully prototyped in FPGA hardware, which has the potential for ultra-fast performance.

TotalReCaller will be available from http://bioinformatics.nyu.edu/wordpress/projects/totalrecaller/.

5.5 Conclusion

High-throughput DNA sequencing will soon become a standard clinical diagnostic tool. From cancer treatment to prenatal diagnosis, sequencing will provide critical datasets, but errors can lead to severe consequences. Processing time is also a limiting factor; sequencing results need to be ready as quickly as possible. Medical sequencing, therefore, demands extremely accurate, near-online base-calling algorithms. This poses the next challenge for the field.

Acknowledgments The authors would like to thank Fabian Menges, Giuseppe Narzisi, and Bud Mishra for sharing their early TotalReCaller results, for Dan Valente for formulating the unified distortion model, and for Dina Esposito for useful comments on the chapter. Yaniv Erlich is an Andria and Paul Heafy family fellow.

References

Bentley DR, Balasubramanian S, Swerdlow HP, Smith GP, Milton J, Brown CG, Hall KP, Evers DJ, Barnes CL, Bignell HR et al. 2008. Accurate whole human genome sequencing using reversible terminator chemistry. *Nature* 456(7218): 53–59.

Erlich Y, Mitra PP, delaBastide M, McCombie WR, Hannon GJ. 2008. Alta-Cyclic: a self-optimizing base caller for next-generation sequencing. *Nat Methods* 5(8): 679–682.

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.

Kailath T, Poor HV. 1998. Detection of stochastic processes. IEEE T. *Inform Theory* 44(6): 2230–2259.

Kao WC, Song YS. 2011. naiveBayesCall: An Efficient Model-Based Base-Calling Algorithm for High-Throughput Sequencing. *J Comput Biol* 18(3): 365–377.

Kao WC, Stevens K, Song YS. 2009. BayesCall: A model-based base-calling algorithm for high-throughput short-read sequencing. *Genome Res* 19(10): 1884–1895.

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

Kriseman J, Busick C, Szelinger S, Dinu V. 2010. BING: biomedical informatics pipeline for Next Generation Sequencing. *J Biomed Inform* 43(3): 428–434.

Ledergerber C, Dessimoz C. 2011. Base-calling for next-generation sequencing platforms. *Brief Bioinform*.

Li L, Speed TP. 1999. An estimate of the crosstalk matrix in four-dye fluorescence-based DNA sequencing. *Electrophoresis* 20(7): 1433–1442.

Lieberman-Aiden E, van Berkum NL, Williams L, Imakaev M, Ragoczy T, Telling A, Amit I, Lajoie BR, Sabo PJ, Dorschner MO et al. 2009. Comprehensive mapping of long-range interactions reveals folding principles of the human genome. *Science* 326(5950): 289–293.

Lister R, Pelizzola M, Kida YS, Hawkins RD, Nery JR, Hon G, Antosiewicz-Bourget J, O'Malley R, Castanon R, Klugman S et al. 2011. Hotspots of aberrant epigenomic reprogramming in human induced pluripotent stem cells. *Nature* 471(7336): 68–73.

Metzker ML. 2010. Sequencing technologies – the next generation. *Nat Rev Genet* 11(1): 31–46.

Quinlan AR, Stewart DA, Stromberg MP, Marth GT. 2008. Pyrobayes: an improved base caller for SNP discovery in pyrosequences. *Nat Methods* 5(2): 179–181.

Romiguier J, Ranwez V, Douzery EJ, Galtier N. 2010. Contrasting GC-content dynamics across 33 mammalian genomes: relationship with life-history traits and chromosome sizes. *Genome Res* 20(8): 1001–1009.

Rougemont J, Amzallag A, Iseli C, Farinelli L, Xenarios I, Naef F. 2008. Probabilistic base-calling of Solexa sequencing data. *BMC Bioinformatics* 9: 431.

Shenoi BA. 2006. *Introduction to digital signal processing and filter design*. Wiley ; John Wiley [distributor], Hoboken, NJ.

Sklar LA. 2005. *Flow cytometry for biotechnology*. Oxford University Press, New York.

Wang Z, Gerstein M, Snyder M. 2009. RNA-Seq: a revolutionary tool for transcriptomics. *Nat Rev Genet* 10(1): 57–63.

Whiteford N, Skelly T, Curtis C, Ritchie ME, Lohr A, Zaranek AW, Abnizova I, Brown C. 2009. Swift: primary data analysis for the Illumina Solexa sequencing platform. *Bioinformatics* 25(17): 2194–2199.

Wu X, Ding L, Li Z, Zhang Y, Liu X, Wang L. 2010. Determination of the migration of bisphenol diglycidyl ethers from food contact materials by high performance chromatography-tandem mass spectrometry coupled with multi-walled carbon nanotubes solid phase extraction. *Se Pu* 28(11): 1094–1098.

Chapter 6
De Novo Short-Read Assembly

Douglas W. Bryant Jr. and Todd C. Mockler

Abstract An imperative first step in the characterization of a species or individual is the sequencing and subsequent assembly and analysis of its genome. High-throughput sequencing technology has ushered in a new way of thinking about this fundamental undertaking. Next-generation sequencing machines produce reads through highly parallel operation and produce a much greater quantity of data per experiment, at drastically reduced cost per base, with the drawback of short-read length. A new generation of de novo assembly algorithms and applications has arisen to meet the challenges inherent to this new type of sequence data. Many de novo assembly algorithms have been implemented, each with its own set of assumptions, strengths, and weaknesses. While the details of each such assembler are unique, all of these assemblers share a common conceptual foundation and all must contend with the same set of complexities presented by next-generation sequencing data. This chapter discusses each type of de novo short-read assembly algorithm with emphasis on the similarities and differences between them.

D.W. Bryant Jr. (✉)
Department of Botany and Plant Pathology, Center for Genome Research and Biocomputing, Oregon State University, Corvallis, OR 97331, USA

Department of Electrical Engineering and Computer Science, Oregon State University, Corvallis, OR 97331, USA
e-mail: bryantjr@eecs.oregonstate.edu

T.C. Mockler
Department of Botany and Plant Pathology, Center for Genome Research and Biocomputing, Oregon State University, Corvallis, OR 97331, USA
e-mail: tmockler@cgrb.oregonstate.edu

N. Rodríguez-Ezpeleta et al. (eds.), *Bioinformatics for High Throughput Sequencing*,
DOI 10.1007/978-1-4614-0782-9_6,

6.1 De Novo Short-Read Assembly

6.1.1 Next-Generation Sequencing

An imperative first step in the characterization of a species or individual is the sequencing and subsequent assembly and analysis of its genome. High-throughput sequencing (HTS) technology, reviewed for example in Schadt et al. (2010), Kircher and Kelso (2010), and Zhao and Grant (2011), has ushered in a new way of thinking about this fundamental undertaking. Many commercial next-generation sequencing (NGS) platforms exist, all of which together possess several key characteristics distinguishing them from first-generation sequencing platforms. NGS machines produce data through highly parallel operation and produce a much greater quantity of data per experiment, at drastically reduced cost per base. Unfortunately, reads produced by NGS platforms are dramatically shorter than those generated by first-generation platforms, requiring much higher coverage in order to satisfy overlap detection criteria. Of course, high coverage leads to larger data sets and higher complexity.

With the emergence of high-throughput sequencing technology, a new generation of de novo assembly algorithms and applications has arisen to meet the challenges inherent to NGS data. Many de novo assembly algorithms have been implemented, each with its own set of assumptions, strengths, and weaknesses. While the details of each such assembler are unique, all of these assemblers share a common conceptual foundation and all must contend with the same set of complexities presented by NGS data.

6.1.2 From the Beginning

Assembly of a target genome starts with generation of data through a whole-genome shotgun sequencing experiment. In shotgun sequencing, reads are sampled from random positions along a target molecule (Sanger et al. 1980). In whole-genome shotgun (WGS) sequencing the target DNA molecules are the chromosomes, resulting in reads taken from random positions along a genome. WGS assembly aims to reconstruct the original sequence, up to chromosome length, from these reads. Assembly of reads from a WGS sequencing experiment is possible when the genome is over-sampled, resulting in many overlapping reads. A computer program designed for this assembly task is called an assembler.

Meaning literally “from the beginning,” the Latin expression “de novo” has a specific denotation when applied to whole-genome shotgun assembly. De novo WGS assembly refers to an assembly process based entirely on reads from the target genome. Here, no previously resolved sequence data are considered, such as pre-existing sequences from genomes, transcripts, and proteins, or homology information.

6.1.3 Assembly

6.1.3.1 Contigs, Scaffolds, and Chromosomes

Assemblers rely directly on the assumption that reads share common substrings, which implies they originate from the same genomic location. Through analysis of these overlapping substrings, a putative reconstruction of the target genome can be created. Reads are assembled into contigs, contigs into scaffolds, and scaffolds into chromosomes. Contigs represent a consensus sequence from reads, while scaffolds, also known as supercontigs and metacontigs, define contig order and direction as well as the gaps between contigs. Once all scaffolds have been ordered, chromosomes can be inferred.

Assemblies are generally output in multi-FASTA format, in which each contig is listed and associated with a header. Consensus sequences are represented using the International Union of Pure and Applied Chemistry (IUPAC) symbols and contain at least the four characters A, C, G, and T, although additional characters, each with some special meaning, may also be present. For example, scaffold consensus sequences may contain Ns in the gaps between scaffolds, with the number of consecutive Ns representing a gap-length estimate. IUPAC notation additionally contains characters for ambiguous bases where, for example, the character Y denotes either of the pyrimidines C or T while the character R denotes either of the purines A or G.

6.1.3.2 Challenges

Several factors confound WGS sequence assembly. First, regardless of the technology used to generate them, WGS sequencing results in reads drastically shorter than the molecules from which they originate. Second, the computational complexity of assembly is severely enhanced by the vast amounts of data produced by NGS experiments. Third, sequencing error can induce assembly error, leading to incorrect and drastically shortened contigs. Fourth, repeat sequences in the target can be indistinguishable from each other, especially true if the repeat regions are longer than the length of the reads. Finally, nonuniform coverage of the target can lead to the invalidation of statistical tests and diagnostics. Each of these factors is discussed here.

6.1.3.3 Short Reads

Reads produced by Sanger sequencing, while longer than those produced in NGS experiments, are even themselves each only a small fraction the length of even the smallest genomes. WGS attempts to overcome this limitation through significantly over-sampling the target genome, producing reads from random positions along the molecule. Through this deep, random over-sampling assemblers attempt to reproduce the original target sequence.

6.1.3.4 Dataset Size

NGS experiments produce a massive amount of data, increasing the computational complexity of assembly simply by the size of such datasets. Many NGS assemblers manage these large volumes of data through use of K-mers. A K-mer is simply a series of contiguous base calls of length *K*, where *K* is any positive integer. Instead of searching for overlaps, assemblers search reads for shared K-mers, being generally easier to identify shared K-mers than overlaps. While K-mer-based algorithms are less sensitive than overlap-based algorithms and so may miss some true overlaps, the computational complexity associated with detecting shared K-mers is significantly lower than an all-against-all overlap search. It is important to choose a *K* such that most false overlaps do not share a K-mer by chance and small enough that most true overlaps do share a K-mer.

6.1.3.5 Sequencing Error

Regardless of the NGS platform used to generate reads, each base of output has some probability of being incorrectly called. Base-calling errors increase assembly difficulty, in particular by confounding repeat resolution. Assemblers must be robust against imperfect sequence alignments to avoid construction of false-positive joins.

6.1.3.6 Repeats

Regions in the target which share perfect repeats can be indistinguishable, especially when the repeated regions are longer than the read length. Inexact repeats can be separated through careful correlation of reads by patterns in their different base calls (Kececioglu and Ju 2001). Repeat separation is made easier by high coverage and made harder by base-calling error. Resolving repeats shorter than reads requires sufficient unique read sequence on either side of the repeated region. For resolving repeats longer than the read length, paired ends or "mate-pairs" are necessary.

6.1.3.7 Nonuniform Coverage

As WGS sequencing randomly samples the target genome, through chance alone it is unlikely that the target will be sequenced at uniform coverage. Coverage nonuniformity is also induced by variation in cellular copy number between source molecules and through biases inherent in the different sequencing technologies. Too low coverage can result in assembly gaps, and coverage variability invalidates coverage-based statistical tests and diagnostics used, for example, to detect over-collapsed repeats.

6.1.3.8 Comparing Assemblies

Assemblies are measured by size and accuracy of their contigs and scaffolds, communicated in statistics such as maximum contig or scaffold length and combined total length. Assemblies can also be measured by their N50, defined as the smallest contig in the set of largest contigs whose combined length accounts for at least 50% of the total assembly. Also important is the assembly's degree of paired-end gap-length constraint satisfaction. Finally, alignments to similar previously curated reference sequence are valuable when such reference sequences exist.

6.2 Graphs

6.2.1 What is a Graph?

NGS assemblers can be organized into three broad categories, including greedy assemblers, overlap-layout-consensus assemblers, and de Bruijn graph assemblers. While details differ significantly between categories, each of these techniques is based implicitly or explicitly on graphs.

A graph is an abstract representation of a set of objects, depicted as vertices, or nodes. Nodes may be connected, represented by an edge between two nodes. Edges may be directed or undirected. Directed edges may only be traversed in one direction between its connected vertices, while undirected edges may be traversed in either direction; directed edges connect a source node to a sink node. A graph may be conceptualized as a set of dots representing its vertices, with interconnecting lines representing its edges. Given a graph, a path through the graph is an ordered set of nodes representing a traversal of edges and nodes.

6.2.2 Graphs Types

Several different types of graphs are used by NGS assemblers, primarily distinguished by how a node is defined and implemented. Here we describe three different graph types, including overlap graphs, de Bruijn graphs, and K-mer graphs.

6.2.2.1 Overlap Graph

An overlap graph explicitly represents reads and the overlaps between them. Read overlaps are computed by a series of all-versus-all pair-wise read alignments. Here reads are represented by nodes, while overlaps between reads are represented by edges. The graph may use distinct elements to represent forward and reverse

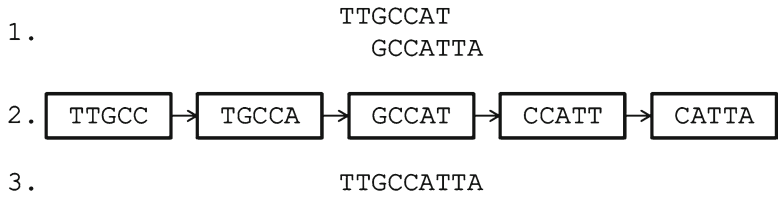

Fig. 6.1 A pair of aligned reads represented by a K-mer graph with a *K* of size 4. (1) Two reads share an overlap of five bases. (2) One K-mer graph represents both reads. (3) A contig is defined through a simple path through the graph, without need for explicit pair-wise sequence alignment calculations

complement sequences, read directionality, and overlap lengths. Potential contigs are derived from paths through the graph which can be converted directly to sequence. To preserve the semantics of DNA's double-stranded structure, the graph can be constructed in one of two ways. First, the graph can have separate nodes for read ends in which case paths must exit the opposite end of the read than they enter. Second, the graph can have separate edges for the forward and reverse strands, in which case paths must exit a node on the same strand that they enter.

6.2.2.2 De Bruijn Graph

A De Bruijn graph (DBG) is a directed graph used to represent overlaps between sequences of symbols. Here nodes represent all possible fixed-length strings producible by a finite alphabet, while edges represent suffix-to-prefix perfect overlaps between these strings.

6.2.2.3 K-mer Graph

A K-mer graph is a construct similar to the DBG. In the K-mer graph, nodes represent all fixed-length subsequences drawn from a larger sequence while edges represent all fixed-length overlaps between these subsequences. Nodes may represent overlaps of $K-1$ bases, in which case an edge would exist for the K-mer starting at each base.

Instead of representing a single sequence, a K-mer graph may represent a set of sequences. Here the K-mer graph may represent the set of input reads, each of which induces a path. One implicit advantage in this case is the discovery of perfect overlaps without any pair-wise sequence alignment calculations as shown in Fig. 6.1.

K-mer graphs are more sensitive to repeats and sequencing errors than overlap graphs due to the shorter length of K-mers versus the length of reads. Each single-base sequencing error induces up to *K* false nodes, with each false node having some chance of matching some other node spuriously, leading to false paths convergence.

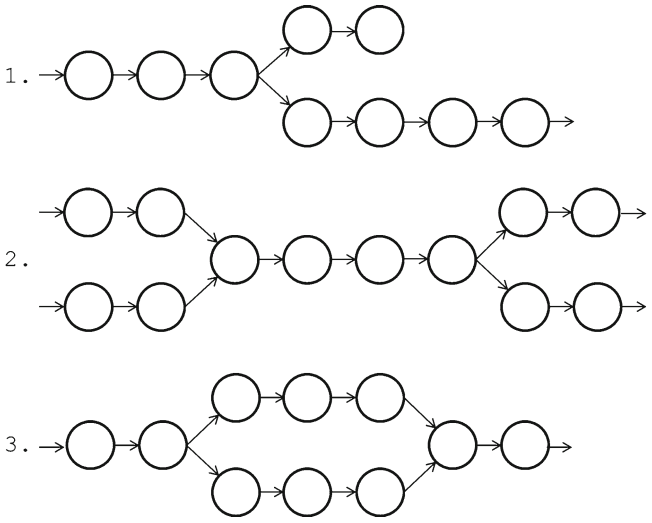

Fig. 6.2 Several different types of graph complications are shown here. (1) Spurs are short dead-end subpaths, usually caused by a base-call error near the end of a read. (2) Repeat sequences induce the frayed-rope pattern. (3) Bubbles are usually caused by base-call error in the middle of reads

6.2.3 Challenges

In building overlap and K-mer graphs from next-generation sequencing data, several key complications are commonly encountered. These complications include short, dead-end subpaths known as spurs, paths that converge then diverge resulting in a frayed-rope pattern, paths that diverge then converge known as bubbles, and cycles which cause paths to converge on themselves. Each of these issues is discussed here.

6.2.3.1 Spurs

Induced by sequencing error toward one end of a read, spurs are short, dead-end subpaths branching from the main path. Spurs may also be a result of gaps in coverage, illustrated in Fig. 6.2(1).

6.2.3.2 Frayed-Rope

Paths that converge then diverge form a frayed-rope pattern. Frayed-rope patterns are induced by repeats in the target, illustrated in Fig. 6.2(2).

6.2.3.3 Bubbles

Bubbles occur when a path diverges and then converges. Bubbles are induced by sequencing error in the interior of a read and by true polymorphism in the target. As exact bubble detection is intractable, heuristics are generally used when searching for bubbles, illustrated in Fig. 6.2(3).

6.2.3.4 Cycles

Paths that converge on themselves form cycles.

6.3 Approaches to Assembly

6.3.1 Greedy Assemblers

6.3.1.1 Introduction

Early NGS assembler offerings were based on naïve, greedy algorithms, extending contigs based only on current information available at each step. These algorithms worked on the simple principle of iteratively extending a read or contig by adding one more read or contig, based on sequence overlaps. This operation was repeated until the current read or contig could be extended no further, after which the next read or contig was extended in the same manner, until there were no more contigs to extend. Each extension operation chooses the current highest-scoring sequence overlap, scoring overlaps based on traits such as the number of matching bases. These types of naïve algorithms can easily become stuck at local maxima if, for example, the current contig is extended by reads which may have helped other contigs to grow even longer. While greedy assembler algorithms are implicitly based on graphs, no explicit graph is saved and the graph operations are drastically simplified by considering only the highest-scoring edges during contig assembly.

All assemblers need mechanisms to reduce the incorporation of false-positive overlaps into contigs, and greedy algorithms are no exception. If an assembler scores overlaps based only on read depth, for example, false overlaps based on repetitive sequence could score higher than true overlaps. In this case, the resulting contigs would be the result of joining unrelated sequences on either side of a repeat. Implementing techniques for avoiding such situations is important.

6.3.1.2 SSAKE

As the first short-read assembler, SSAKE (Warren et al. 2007) was designed to assemble unpaired reads of uniform length. SSAKE begins by indexing reads by

their prefixes in a lookup table. Reads are searched iteratively for those whose prefix overlaps the end of the growing contig over some minimum length.

During contig extension, SSAKE chooses among reads with equally long overlaps based on several factors. First, reads whose sequence is confirmed by other reads are preferred. This helps avoid the incorporation of error-containing reads which should have very low sequence multiplicity. Second, SSAKE detects when the set of candidate reads presents multiple different extensions by identifying when candidate read suffixes contain differences confirmed in other reads, cases resulting in graph branching. Here, SSAKE ceases extension of the current contig. This default behavior may be overridden by the user.

If no reads satisfy the initial minimum overlap threshold SSAKE decrements the threshold length until a second minimum is reached, allowing the user to specify how aggressively SSAKE pursues extensions through possible repeat boundaries and low-coverage regions. Extensions to SSAKE have allowed the software to exploit paired-end reads and reads with mismatches.

6.3.1.3 SHARCGS

Like SSAKE, SHARCGS (Dohm et al. 2007) was designed to assemble high-coverage, unpaired reads of uniform length and operates in a manner similar to SSAKE while adding pre- and postprocessing to SSAKE's basic iterative extension algorithm. Its preprocessor filters potentially erroneous reads by requiring that exact full-length matches be present in other reads. A more stringent filter is available, one which requires that the combined quality values of matching reads exceed some minimum threshold.

SHARCGS filters reads three times, resulting in three different filtered read sets. Each read set is assembled separately through iterative contig extension, followed by a postprocess. In this postprocess SHARCGS merges the three resulting contig sets through sequence alignment, aiming to extend contigs from highly confirmed reads through the integration of longer contigs from lower-stringency read sets.

6.3.1.4 VCAKE

Another assembler based on the iterative extension algorithm, VCAKE (Jeck et al. 2007) differentiates itself from other similar assemblers by possessing the ability to incorporate imperfect matches during contig extension. When extending contigs, VCAKE utilizes a voting mechanic. All reads whose prefixes match the end of the growing contig for at least some minimum length and with at least some minimum coverage are aligned to the growing contig. The contig is extended by one base at a time, at the contig's $n+1$ position. Each time the contig is extended, the identity of the base added to the end of the contig is determined by votes cast by the aligned reads. Each aligned read casts a vote for its base at the $n+1$ position, with the

identity of the added base determined as the base with the highest number of votes. In this way VCAKE favors reads containing sequences with higher multiplicity in an attempt to minimize the incorporation of errors, while preserving its ability to use error-containing reads.

6.3.1.5 QSRA

Created in an effort to lengthen contigs past low coverage regions, QSRA (Bryant et al. 2009) extends the VCAKE voting algorithm through the use of quality scores. Contig extension proceeds as in the VCAKE algorithm until extension would otherwise halt due to low coverage. In this case, QSRA continues contig extension if the bases aligned to the contig suffix match or exceed some minimum quality score. While still a greedy implementation, in some cases QSRA manages to extend contigs significantly further than possible using the same algorithm but without this use of quality scores.

6.3.2 Overlap-Layout-Consensus

6.3.2.1 Introduction

Optimized for large genomes and for read lengths of at least one hundred base-pairs (bp), assemblers based on the overlap-layout-consensus (OLC) approach operate in three stages. First, read overlaps are catalogued. Second, an overlap graph is built. Finally, the consensus sequence is determined.

6.3.2.2 Overlap Discovery

In this first stage, read overlaps are computed. This involves an all-against-all pairwise comparison of reads, accomplished through use of a seed-and-extend heuristic. Here, K-mer content is precomputed across all reads. Reads which share K-mers are selected as overlap candidates, followed by computation of alignments using K-mers as alignment seeds.

6.3.2.3 Layout and Manipulation

Based on discovered overlaps, in this stage the overlap graph is built and optimized leading to an approximate read layout. Original input reads are no longer needed at this point and may be purged from computer memory.

6.3.2.4 Consensus

Finally, multiple sequence alignment (MSA) determines the layout of all the reads after which the consensus sequence is inferred. As there is no known efficient method to compute the optimal MSA (Want and Jiang 1994), progressive pair-wise alignments are used. Multiple contigs may be assembled in parallel during this stage.

6.3.2.5 Newbler

Distributed by 454 Life Sciences as a closed-source application, Newbler (Margulies et al. 2005) is a widely used OLC-based assembler. Newbler's first release targeted unpaired reads of approximately 100 bp but has since been revised to exploit paired-end constraints and to operate on much longer reads.

Newbler uses two rounds of OLC. In the first OLC round mini-assemblies, or unitigs, are generated from reads. Unitigs are ideally uncontested by overlaps to reads in other unitigs and serve as preliminary, high-confidence contigs used to seed later assemblies. In the second OLC round, larger contigs are generated from these unitigs in a process that joins unitigs into a contig layout based on their pair-wise overlaps. Unitigs may be split in cases where one's prefix and suffix align to different contigs. In these cases splits may actually represent the splitting of individual reads, leading to individual reads being placed in different contigs.

When possible Newbler exploits instrument metrics to overcome base-calling error. With 454 data, there is some associated uncertainty in homopolymer run length. Newbler uses the 454 platform-supplied signal strength associated with each nucleotide call to accurately determine the number of contiguous bases in these homopolymer repeats. Unitig and contig consensus sequences are determined in "flow space," in which the normalized signal is proportionally correlated to the number of nucleotide repeats at that position in the read. Newbler calculates the average signal for each column in the MSA to form the consensus.

6.3.2.6 Celera Assembler/CABOG

Originally implemented for the assembly of Sanger reads, the Celera Assembler has been revised for 454 data (Batzoglu et al. 2002) in a pipeline called CABOG (Miller et al. 2008). To overcome the homopolymer run-length uncertainty inherent to 454 data, CABOG collapses homopolymer repeats to single bases.

Unitigs are initially built out of only those reads which are not substrings of larger reads, a concern due to the highly variable read length in 454 data. These substring-reads are initially avoided due to their higher susceptibility to repeat-induced false overlaps.

A base-call error correction scheme is also applied by CABOG. Here, each read is compared to its set of overlapping reads, with errors inferred where bases are

contradicted by many overlaps. Instead of fixing the read directly, CABOG modifies its error rates in overlaps spanning the inferred error. A user-supplied threshold for error rates is then applied, filtering out reads whose error rates are too high. From the surviving overlaps a filter for minimum alignment length is applied, followed by the selection of the overlap with the most aligned bases for each read end. This simple overlap selection method, choosing the overlap with the most aligned bases, eliminates many of the same suboptimal overlaps by the more complex transitive edge removal algorithm (Myers 1995), implemented in the original Celera Assembler.

CABOG then constructs its overlap graph from these reads and overlaps. Unitigs are built from this graph out of maximal simple paths which are free of branches and intersections, after which a graph of unitigs plus paired-end constraints is constructed. Within this unitig graph unitigs are joined into contigs and contigs into scaffolds, applying a series of graph reductions including removal of transitively inferable edges. CABOG finally derives consensus sequences through the computation of multiple sequence alignments from the scaffold layouts and reads.

6.3.2.7 Edena

Whereas most OLC assemblers target assembly of Sanger or 454 data, Edena (Hernandez et al. 2008) was designed to assemble homogeneous-length short reads from the Illumina and SOLiD platforms. In a similar algorithm to other OLC assemblers, Edena first discards duplicate reads, after which Edena finds all perfect overlaps of at least some minimum length. Individual overlaps that are redundant with pairs of other overlaps are removed, followed by the removal of spurs and bubbles. Finally, contigs are assembled by following unambiguous paths in the graph.

6.3.3 De Bruijn Graph

6.3.3.1 Introduction

Most widely applied to data from the Illumina and SOLiD platforms, the DBG approach takes its name from the ideal scenario where, given perfect, error-free K-mers spanning every repeat and fully covering the target genome, the graph would be a de Bruijn graph containing a path that traverses every edge exactly once. While real NGS data does not fit this ideal scenario, this approach remains an attractive one for dealing with large quantities of data. As the DBG relies on K-mers, an all-against-all overlap search is unnecessary. Further, individual reads need not be stored, and redundant sequence is compressed. Still, for large genomes the K-mer graph can require a massive amount of computer memory.

Graph construction occurs through an exhaustive K-mer search over the input reads. K-mers are generally catalogued in a hash-table, allowing for constant-time lookups during graph construction. While this hash-table does require some computer

memory, only a single copy of each K-mer is stored, leading to a smaller memory footprint than the input reads given that the reads share K-mers. Sequence assembly follows naturally from graph construction, though assembly is complicated by several real-world factors, including the double-stranded nature of DNA, palindromes, sequencing error, and genomic repeats. Each of these factors is discussed here.

6.3.3.2 Double Strandedness

When searching for read overlaps it is important to account for the double-stranded nature of DNA, recognizing that the forward sequence of a read may overlap either the forward sequence or the reverse complement sequence of other reads. Several different K-mer graph implementations have been developed to facilitate this type of search. One implementation stores the forward and the reverse complement strands together in a single node with two halves, forcing paths to enter and exit the same half (Zerbino and Birney 2008). Another implementation creates nodes for both strands, later ensuring that the resulting sequence is not output twice (Idury and Waterman 1995).

6.3.3.3 Palindromes

DNA sequences which are themselves their own reverse complements are known as palindromic sequences. Palindromic sequences induce paths that fold back into themselves. Some assemblers including Velvet (Zerbino and Birney 2008) and SOAPdenovo (Li et al. 2009) prevent this issue by requiring that K-mer lengths be odd.

6.3.3.4 Sequencing Error

DBG assemblers employ several techniques to mitigate issues resulting from sequencing error. First, many assemblers preprocess the reads and either remove errors through base substitution, remove the error-containing reads themselves, or catalogue all encountered errors for later use. Second, graph edges which represent a higher number of K-mers are more highly trusted than those representing low numbers. Third, paths through the DBG are converted to sequence, and sequence alignment algorithms are used to collapse highly similar paths.

6.3.3.5 Repeats

Several types of complex repeat structures are present in real genomes, including inverted repeats, tandem repeats, inexact repeats, and nested repeats. K-mers shorter than the length of a repeat lead to unresolvable sequences, inducing graph

complications. Perfect repeats of length at least *K* lead to frayed-rope patterns within the graph, where paths converge then diverge. One technique for resolving such repeat-induced graph complications is through the use of mate pair constraints.

Several DBG assemblers are discussed here, with emphasis on specific implementation details and differences.

6.3.3.6 Euler

Developed originally to assemble Sanger reads (Pevzner et al. 2001), DBG-based Euler has been modified to operate on 454 reads (Chaisson et al. 2004), single-end Illumina reads (Chaisson and Pevzner 2008), and paired-end Illumina reads (Chaisson et al. 2009).

Prior to building its DBG, Euler first applies a preprocess filtering step dubbed spectral alignment to the input reads in an attempt to detect base-call errors. Relying on both read redundancy and on the randomness of sequencing errors, this filter attempts to detect erroneous base calls by identifying K-mers with low frequency. Where most true K-mers will be repeated over many reads, K-mers resulting from base-call errors should occur with a much lower frequency.

Euler's filtering step is implemented by first associating each K-mer with its observed frequency over the input reads, after which K-mers with frequency below some threshold are either corrected or removed from the input set. This threshold is calculated after computing the distribution of K-mer frequencies over all input reads, a distribution which is usually bi-modal. In this bi-modal distribution, the first peak entails low-frequency K-mers assumedly present due to sequencing errors, while the second peak entails true positive K-mers, present due both to redundant read coverage and genomic repeats. On this distribution, Euler bases its frequency threshold. K-mers below this threshold are presumed to be and labeled as false while K-mers above this threshold are presumed and labeled true.

Once all K-mers present in the input set have been labeled as either false or true, Euler examines each input read. For each read which contains one or more false K-mers, Euler attempts to substitute bases in a greedy manner such that the false K-mers are eliminated. After this correction step if the read is fully corrected it is accepted, otherwise it is rejected. Rejected reads may be used later following assembly to bridge low-coverage regions.

While correction is important for reads with both high error and high coverage, correction can mask true polymorphism and can destroy true K-mers whose low representation is due only to chance. Further, correction may create an incorrect read by settling on K-mers which individually occur in several reads but which never otherwise occur together in a single read. Assemblers based on the OLC approach have an analogous base-call correction step based on overlaps instead of on K-mers.

From these filtered and corrected reads, Euler creates a K-mer graph which subsequently undergoes a series of manipulations. As the graph is based on K-mers rather than on reads directly some information is lost. Euler attempts to recover this information in a read-threading step by laying entire reads onto its K-mer graph following graph construction, where reads map to unique nodes and are consistent with some graph traversal. Reads ending within a repeat are consistent with any path which exits the repeat, while reads spanning a repeat are consistent only with a subset of these paths. Each read within this second group, those spanning repeats, are used to resolve one copy of the collapsed repeat, allowing for the resolution of repeats whose length is between K and the read length.

Paired-end reads are treated by Euler as long reads with interior unknown bases and are used to resolve repeats which are longer than individual reads. Paired-end reads which span a repeat are used to join one path which enters the repeat with one path which exits the repeat, and to resolve some complex repeat-induced tangles. Multiple paths may exist in the graph between two nodes where each of the two nodes corresponds to one end of a paired-end read, with each path indicating a different DNA sequence. In some cases, only a single such path will satisfy the paired-end length constraint whereby the correct DNA sequence is easily identified. Otherwise, as exhaustive search over all paths between a mate pair may be intractable, Euler restricts the search space by using the paired-end length as a bound on path length.

In the next step, Euler applies graph simplifications. Assuming that spurs are due to sequencing errors unidentified by its spectral alignment filter, Euler applies spur erosion which reduces path branching and results in the lengthening of simple paths. Edges that appear repetitive are then identified and removed.

As many NGS platforms produce reads with lower quality base calls near their 3′ ends, Euler trusts read prefixes more than it does read suffixes. Trusted prefixes of variable length are identified during the spectral alignment step. Prefixes and suffixes can map to multiple paths during read threading. Euler favors mappings that are significantly better than the second-best choice. Suffixes are allowed to contribute to connectivity only.

In a K-mer graph, larger values for K work to resolve longer repeats but can contribute to fragmentation in regions of low coverage. Euler addresses this tradeoff through the construction and processing of not one but two different K-mer graphs, each with its own value for K. Edges which are present in the smaller-K graph but missing in the larger-K graph are added as pseudo-edges to the larger-K graph. Paths in the larger-K graph which are extended by these pseudo edges work to elongate contigs during assembly. In effect this technique creates initial, reliable contigs through a large K value and then bridges gaps with shorter K-mers. This technique is analogous to gap-filling in OLC assemblers.

Initially implemented as a method for converting genomic sequence to a repeat graph, a data structure known as an A-Bruijn graph is a combination of an adjacency matrix and a DBG. Used by some of the Euler software, in this structure graph nodes represent consecutive columns in multiple sequence alignments. These nodes can be less sensitive to sequencing error as compared to nodes representing K-mers.

6.3.3.7 Velvet

A very popular choice among DBG assemblers due both to its effectiveness and ease of use, Velvet (Zerbino and Birney, 2008) performs graph simplifications which collapse simple paths into single nodes. This graph simplification process can yield much simpler graphs while avoiding information loss and is invoked during initial graph construction in addition to several times during its assembly process. Introduced as elimination of singletons for K-mer graphs (Idury and Waterman 1995), this step is analogous to unitig formation in overlap graphs (Myers 1995) and OLC assemblers (Myers et al. 2000).

Spurs are removed iteratively in a manner similar to Euler's erosion procedure, but a procedure equivalent to Euler's spectral alignment filter is not applied by Velvet. While Velvet does allow user specification of a minimum number of occurrences for a K-mer to qualify as a graph node, use of this naive parameter is discouraged in the Velvet documentation.

To reduce graph complexity, Velvet performs a bounded search for bubbles in the graph. Velvet's tour bus algorithm begins at nodes with multiple outgoing edges and, utilizing a breadth-first search, seeks out bubbles within the graph. Since it is possible in graphs of real data for bubbles to be nested within bubbles an exhaustive search for all bubbles would be intractable. Candidate paths are traversed in parallel, stepping ahead one node on each path every iteration until the path lengths exceed a threshold, thereby bounding the search. Bubble candidates are narrowed to those which have sequence similarity on the alternate paths. Velvet removes the path represented by fewer reads and then re-aligns reads from the removed path to the remaining path. This read re-alignment step utilizes what is essentially a column-by-column voting mechanic to call consensus bases which may end up masking genuine sequence differences due to polymorphism or over-collapse of near-identical repeats. This procedure is similar to bulge removal in Euler and analogous to bubble detection and bubble smoothing in OLC assemblers.

To further reduce graph complexity, Velvet removes paths represented by fewer reads than a threshold in a procedure called read threading. While this operation does risk removing true low-coverage sequence, it removes mostly spurious connections induced by convergent sequencing errors. If long reads were provided, Velvet exploits these reads through an algorithm called "Rock Band." This algorithm forms nodes out of paths confirmed by two or more long reads provided that no other two long reads provide a consistent contradiction.

In its final graph reduction step, Velvet addresses mate pairs. Early versions of Velvet used an algorithm called "breadcrumb" similar to mate pair threading in DBG algorithms and gap filling in OLC algorithms. This algorithm operated on pairs of long simple paths, or contigs, connected by mate pairs. These long contigs were used to anchor the gap between which Velvet attempted to fill with short contigs. All short contigs linked to either long contig were gathered. Over the DBG, a breadth-first search was conducted in an attempt to find a single path linking the long contigs by traversing these short contigs. Later versions of Velvet use an algorithm

called “pebble” in which unique and repeat contigs substitute for breadcrumb’s long and short contigs, respectively. The decision to classify a contig as unique or repeat is based on read coverage per contig via a statistical test similar to the A-stat in Celera Assembler and a given coverage expectation. The given insert length distribution is exploited to build an approximate contig layout. The DBG is then searched for a path that is consistent with this layout.

Three of Velvet’s parameters are of critical importance. The first such parameter is the length of the K-mers, constrained to be an odd integer to preclude nodes representing palindromic repeats. Second is the minimum expected frequency of K-mers in the reads, which determines those K-mers to be pruned *a-priori*. Finally, the expected coverage of the genome in read depth controls spurious connection breaking.

6.3.3.8 ABySS

Designed specifically to address memory limitations when assembling large genomes, ABySS (Simpson et al. 2009) is a distributed DBG assembler implementation. ABySS distributes its K-mer graph and graph computations across a set of computers in an effort to use their combined memory.

When designing a task to function on a computational grid as ABySS does, several issues must be addressed. It must be possible to partition the problem evenly into subtasks, to distribute each subtask evenly across the grid, and to finally combine sub-results into an overall solution. ABySS partitions the assembly process at the individual node level, assigning multiple nodes to any individual CPU, and then processes each node separately. Graph nodes are assigned to CPUs by converting the node’s K-mer to an integer in a strand-neutral manner such that both a K-mer and its reverse complement map to the same integer.

Parallel implementations of the graph simplifications used by Euler and Velvet are applied by ABySS. Spurs shorter than a threshold are removed iteratively, bubbles are smoothed by a bounded search which prefers paths supported by more reads, simple nonintersecting paths are transformed into contigs, and mate threading is performed.

ABySS uses a representation of the K-mer graph in which each node represents a K-mer and its reverse complement. Each graph node maintains an additional 8 bits of information representing the existence or nonexistence of each of the four possible one-base extensions at either end, upon which graph edges are inferred. Graph paths are followed in parallel starting at arbitrary graph nodes per CPU. Successors are determined by converting the node’s last $K-1$ bases plus its one-base extension numerically to the address of the successor node. When a path traverses a node located on a separate CPU the process requests the remote information, working on other graph nodes while waiting for a response. ABySS exploits paired-end reads to merge contigs in a postprocess.

6.3.3.9 AllPaths/AllPaths-LG

Like AbySS, AllPaths (Butler et al. 2008) is a DBG implementation targeted at large genome assembly.

AllPaths begins by applying a read-correcting preprocessor, similar to Euler's spectral alignment step, in which AllPaths trusts K-mers occurring with high frequency and with overall high quality. Here, each base must be confirmed by a minimum number of base calls with a quality value above a threshold. This filter operates on K-mers for three different values of *K*, retaining only those reads containing trusted K-mers. Rejected reads may be retained in one of two ways. First, if by substituting up to two low-quality base calls makes its K-mers trusted, and second, if a read is essential for building a path between paired-end reads.

In a second preprocessor step "unipaths" are created. This process begins with the calculation of perfect read overlaps seeded by K-mers. A numerical identifier is assigned to each K-mer such that K-mers seen consecutively in reads and in overlaps receive consecutive identifiers. A database is populated with identifier intervals and the linking reads between them after which appropriate intervals are merged. Based on this database AllPaths builds its DBG.

To its DBG AllPaths first applies spur erosion, or what it calls "unitig graph shaving," followed by a partitioning of the graph designed to resolve genomic repeats by assembling regions that are locally nonrepetitive. Heuristics are then applied to choose partitions that form a tiling path across the genome. Partitions are seeded with nodes corresponding to long, moderately covered, widely separated contigs, after which partitions are populated with nodes and reads linked by alignments and mate pairs. Gaps between paired-end reads are filled by searching the graph for instances where the mate-pair distance constraint is satisfied by only a single path. Each partition is assembled separately and in parallel. Partitions are then bridged where they have overlapping structure in an operation analogous to joining contigs based on sequence overlaps.

AllPaths heuristically removes spurs, small disconnected subgraphs, and paths which are not spanned by paired-ends. Cycles are explicitly defined to match paired-end distance constraints. Paired-ends are used to resolve repeats displaying a frayed-rope pattern.

AllPaths-LG adds several improvements to the AllPaths algorithm. Better error-correction capabilities preserve true SNPs while removing as many sequencing errors as possible, gap filling and scaffolding are improved, and graph simplification has been tuned for better eukaryotic sequence assembly.

6.3.3.10 SOAPdenovo

A third DBG implementation and one targeted at large genome assembly, SOAPdenovo (Li et al. 2009) is a freely available but closed-source application that draws from both OLC and DBG techniques, with an emphasis on minimizing its

memory footprint. Like Velvet, SOAPdenovo requires an odd K-mer size to preclude the creation of nodes representing palindromic repeats.

SOAPdenovo begins by applying a read-correcting preprocessor using preset thresholds for K-mer frequencies, after which its DBG is built. Spur erosion and read threading are applied, followed by the splitting of paths displaying a symmetrical frayed-rope pattern. Bubbles are removed in a manner similar to Velvet's tour bus where higher read coverage determines surviving paths. SOAPdenovo's DBG is more space-efficient than those generated by Euler and Velvet, electing not to store read-tracking information.

Contigs are built from reads via its DBG after which SOAPdenovo discards its DBG and builds scaffolds. Paired-end reads, including those not used in the DBG, are mapped to the contig consensus sequences. A contig graph is then built whose edges represent mate pair constraints between contigs. Complexity is reduced in the contig graph through the removal of edges transitively inferable from others and by isolating contigs traversed by multiple, incompatible paths which are assume to be collapsed repeats. In order to preclude the construction of interleaving scaffolds, SOAPdenovo, like AllPaths, processes contig graph edges in order from small to large. Mate pairs are used to assign reads to gaps between neighbor contigs within a scaffold in a manner similar to CABOG's "rocks and stones" technique and to Velvet's "breadcrumbs" and "pebble" techniques. DBGs are used to assemble the reads assigned to each gap.

6.4 Summary

De novo assembly of next-generation sequence data is imperfect due to complexities inherent in large genomes and sequence error and will remain so. While many different assemblers have been implemented, all successful assemblers draw from a common set of features. First, assemblers use implicit or explicit graphs to represent reads and their overlaps. Second, while accomplished in a variety of ways, assemblers must have error detection and correction capabilities to address sequencing error. Preprocesses may eliminate errors in reads, and error-induced paths may later be removed. Third, nonintersecting paths are collapsed into single nodes. Finally, assemblers convert paths to contigs, and contigs to scaffolds.

Until now, DBG assemblers have enjoyed the most success on NGS data. Optimized for reads of length 100 bp and below, these assemblers have proliferated in the current NGS environment. Originally designed and implemented to assemble longer reads, OLC assemblers may begin to overtake DBG assemblers as NGS read lengths continue to grow.

As technology improves, next-generation platforms produce longer reads, higher data volumes, and fluctuating error rates. Assemblers will continue to evolve along with these continually shifting requirements, and will remain an exciting topic of investigation.

References

Batzoglu, S., Jaffe, D. B., Stanley, K., Butler, J., Gnerre, S., Mauceli, E., et al. (2002). ARACHNE: a whole-genome shotgun assembler. *Genome Research, 12*(1), 177–189.

Bryant, D. W., Wong, W. -K., & Mockler, T. (2009). QSRA – a quality-value guided de novo short read assembler. *BMC Bioinformatics, 1*, 69.

Butler, J., MacCallum, I., Kleber, M., Shlyakhter, I. A., Belmonte, M. K., Lander, E. S., et al. (2008). ALLPATHS: de novo assembly of whole-genome shotgun microreads. *Genome Research, 18*, 810–820.

Chaisson, M. J., & Pevzner, P. A. (2008). Short read fragment assembly of bacterial genomes. *Genome Research*, *18*(2), 324–330.

Chaisson, M. J., Brinza, D., & Pevzner, P. A. (2009). De novo fragment assembly with short mate-paired reads: Does the read length matter? *Genome Research, 19*, 336–346.

Chaisson, M., Pevzner, P., & Tang, H. (2004). Fragment assembly with short reads. *Bioinformatics, 20*(13), 2067–2074.

Dohm, J. C., Lottaz, C., & Borodina, T. (2007). SHARCGS, a fast and highly accurate short-read assembly algorithm for de novo genomic sequencing. *Genome Research, 17*, 1697–1706.

Hernandez, D., Francois, P., Farinelli, L., Osteras, M., & Schrenzel, J. (2008). De novo bacterial genome sequencing: Millions of very short reads assembled on a desktop computer. *Genome Research, 18*(5), 802–809.

Idury, R. M., & Waterman, M. S. (1995). A new algorithm for DNA sequence assembly. *Journal of Computational Biology, 2*, 291–306.

Jeck, W. R., Reinhardt, J. A., Baltrus, D. A., Hickenbotham, M. T., Magrini, V., Mardis, E. R., et al. (2007). Extending assembly of short DNA sequences to handle error. *Bioinformatics*, *23*(21), 2942–2944.

Kececioglu, J., & Ju, J. (2001). Separating repeats in DNA sequence assembly. *Proceedings of the fifth annual international conference on Computational biology*, (pp. 176–183). Montreal, Quebec, Canada.

Kircher, M., & Kelso, J. (2010). High-throughput DNA sequencing – concepts and limitations. *Bioessays, 32*(6), 524–536.

Li, R., Zhu, H., Ruan, J., Qian, W., Fang, X., Shi, Z., et al. (2009). De novo assembly of human genomes with massively parallel short read sequencing. *Genome Research, 20*, 265–272.

Margulies, M., Egholm, M., Altman, W. E., Attiya, S., Bader, J. S., Bemben, L. A., et al. (2005). Genome sequencing in microfabricated high-density picolitre reactors. *Nature, 437*, 376–380.

Miller, J. R., Delcher, A. L., Koren, S., Venter, E., Walenz, B. P., Brownley, A., et al. (2008). Aggressive assembly of pyrosequencing reads with mates. *Bioinformatics, 24*(24), 2818–2824.

Myers, E. W. (1995). Toward Simplifying and Accurately Formulating Fragment Assembly. *Journal of Computational Biology, 2*, 275–290.

Myers, E. W., Sutton, G. G., Delcher, A. L., Dew, I. M., Fasulo, D. P., Flanigan, M. J., et al. (2000). A Whole-Genome Assembly of Drosophila. *Science, 287*(5461), 2196–2204.

Pevzner, P. A., Tang, H., & Waterman, M. S. (2001). An Eulerian path approach to DNA fragment assembly. *PNAS, 98*(17), 9748–9753.

Sanger, F., Coulson, A. R., Barrell, B. G., Smith, A., & Roe, B. A. (1980). Cloning in single-stranded bacteriophage as an aid to rapid DNA sequencing. *Journal of Molecular Biology, 143*(2), 161–178.

Schadt, E. E., Turner, S., & Kasarskis, A. (2010). A window into third-generation sequencing. *Human Molecular Genetics, 19*(R2), R227–240.

Simpson, J. T., Wong, K., Jackman, S. D., Schein, J. E., Jones, S. J., & Birol, I. (2009). ABySS: a parallel assembler for short read sequence data. *Genome Research, 19*(6), 1117–1123.

Want, L., & Jiang, T. (1994, Winter). On the complexity of multiple sequence alignment. *Journal of Computational Biology, 1*(4), 337–348.

Warren, R. L., Sutton, G. G., Jones, S. J., & Holt, R. A. (2007). Assembling millions of short DNA sequences using SSAKE. *Bioinformatics, 23*(4), 500–501.

Zerbino, D., & Birney, E. (2008). Velvet: Algorithms for De Novo Short Read Assembly Using De Bruijn Graphs. *Genome Research, 18*, 821–829.

Zhao, J., & Grant, S. F. (2011). Advances in whole genome sequencing technology. *Current Pharmaceutical Biotechnology, 12*(2), 293–305.

Chapter 7
Short-Read Mapping

Paolo Ribeca

Abstract Present-day high-throughput sequencing techniques routinely produce a flood of genomic information (as high as 540-600 Gbases/machine/week for some technologies). The output comes under the form of short sequence reads; in a typical resequencing application (where the knowledge of a reference genome for the organism being studied is assumed) the sequence reads need to be aligned to the reference.

Such high yields make the use of traditional alignment programs like BLAST unpractical; while resequencing, on the other hand, one is usually interested in considering only matches showing a very high sequence similarity with the original read. This new working setup required the development of a generation of new high-throughput lower-sensitivity alignment programs, called short-read mappers.

Influenced by the standpoint of the algorithm designer, published literature tends to overemphasize speed, and standard working conditions, at the expense of accuracy. In this chapter we attempt to review the state-of-the-art of short-read alignment technology, focusing more on the user's standpoint, and on what is necessary to know to be able to design a high-quality mapping analysis workflow, rather than on purely technical issues.

7.1 Introduction

Present-day high-throughput sequencing (HTS) techniques routinely produce a flood of genomic information. For instance Illumina/Solexa sequencing (Metzker 2010), one of the most widespread technologies at the time of this writing, is able to

P. Ribeca (✉)
Centro Nacional de Análisis Genómico, Baldiri Reixac 4, Barcelona, Spain
e-mail: pribeca@pcb.ub.es; paolo.ribeca@gmail.com

N. Rodríguez-Ezpeleta et al. (eds.), *Bioinformatics for High Throughput Sequencing*,
DOI 10.1007/978-1-4614-0782-9_7, © Springer Science+Business Media, LLC 2012

provide impressive yields: one single HiSeq 2000 machine produces 150–200 Gbases/run – each run lasting about one week, and relatively minor hardware upgrades are expected to boost the technology to 540–600 Gbases/run/machine in a few months from now (late 2011). The output of HTS machines comes under the form of short *sequence reads*, typically 75–150 nt for Illumina, and 400–1,000 nt for 454/Roche (Rothberg and Leamon 2008) (the latter at the price of a much lesser yield).

A typical application of short reads generated by HTS is to use them in a resequencing setup: the knowledge of a *reference genome* for the organism being studied is assumed, and the sequence reads need to be aligned (or *mapped* in HTS parlance) to the reference genome. The yield of HTS technologies makes the use of traditional alignment programs like BLAST (Altschul et al. 1990) unpractical; while resequencing, on the other hand, one is usually interested in considering only matches having a very low maximum possible number of differences from the reference – that is, matches showing a very high sequence similarity with the original read. Such new working setup required the development of a generation of new high-throughput lower-sensitivity alignment programs, called *short-read mappers*.

Influenced by the standpoint of the algorithm designer, published literature tends to overemphasize speed at the expense of accuracy. In particular, little or no heed is usually paid to the fact that mapping performance strongly depends on the parameter space one wishes to explore – the unpleasant truth being that in several situations mapping parameters are dictated by the biological problem at hand, and do not coincide with those chosen by the algorithm developer.

This leads to the wrong perception that something like an "always-ultrafast" mapping method can exist, while in real life, quite to the contrary, all alignment methods always embed complicated trade-offs between speed and accuracy. In turn, such erroneous statements generate in the users a nonchalant attitude, following which mapping programs are often selected on the basis of hearsay beliefs ("I have been told that this method is very fast") rather than on their aptness to the specific task at hand.

In this chapter we attempt to review the state-of-the-art of short-read alignment technology, focusing more on the user's standpoint, and on what is necessary to know to be able to design a high-quality HTS mapping analysis workflow, rather than on the point of view of the algorithm implementor.

7.2 The Problem of Short-Read Mapping

To recover the genomic location which has generated the short read at hand, in our resequencing setup one has to perform on it sequence alignment to the reference genome, selecting afterwards the best candidate(s) among the possibly many matches which show a high similarity to the read.

Although this workflow is in principle very clear, its practical implementation requires choices which are often overlooked. For instance, the criteria usually

adopted to select the "best" matches are key to the subsequent analysis and yet strictly empirical; and some important decisions about the short read can be taken only when the alignments for all the reads are known. In this section, we explore such issues.

7.2.1 Making Provision for Errors

Since we start from the assumption that the reference genome is known, in our setup there are only two possible sources of mismatches between the original sequence in the genome and the short read which originated from it.

1. Small local genomic variants of the individual being considered with respect to the genomic reference known for its species. A typical example are single-nucleotide polymorphisms (SNPs) – which are detected as nucleotide *substitutions* – or short insertions/deletions of nucleotides (*indels*). Their frequency is low (typically between 1/100 bases and 1/1,000 bases in the case of human).
2. Sequencing errors. Usually these are relatively rare in the reads produced by HTS technologies – in particular Illumina/Solexa, where at length 100 nt the majority of the reads align with at most two to three nucleotide substitutions, and only about the 1% of the reads show indels due to base-calling dephasing at sequencing time (the *base calling* is the process after which the machine determines the nucleotide sequence of the short read). In addition, with technologies like Illumina/Solexa and Roche/454 sequencing errors can to some extent be distinguished from variants: they tend to accumulate toward the right end of the read, and to correlate negatively with the qualities of the base calling (see Sect. 7.2.4.1).

As a matter of fact, at least when analyzing sequence produced by the HTS technologies which are nowadays most pervasive, one will usually look for almost perfect matches (although the reader should be warned that the assumptions just examined do not hold in general, and notable exceptions, especially for technologies producing longer reads, do exist – see Sect. 7.4.1). For instance, by default Li and Durbin (2009) search the reference for matches having with the read sequence similarity ≥96%; this corresponds to a maximum of four mismatches in the case of reads of 100 nt.

7.2.1.1 Substitutions and Indels Have a Different Nature

It should be emphasized that from a combinatorial standpoint substitutions (possible replacements of a base with another one) and indels (some nucleotides added to, or removed from, the read) are not on the same footage. In fact, usually the substitution of a single nucleotide does not increase significantly the number of matches which

can be found for a relatively long read, and anyway the increase is roughly independent on the position of the substitution; on the other hand, inserting an indel too close to either hand of the read can have the disastrous effect of generating an essentially unbound number of spurious matches. This is due to the fact that an indel too close to the boundary partitions the read into two blocks, the shortest one of which can be arbitrarily short: thus, the latter will match more and more short sequences, randomly appearing in the genome close to the position of the match of the longest block.

For instance, in the special case of a sequence read mapping without errors, thanks to the mechanism just explained one is bound to find arbitrarily many subdominant alignments (having one indel more than the perfect match) obtained splitting the read in two parts. Hence, particular care should be exercised when choosing alignment parameters.

7.2.2 A Compromise Between Speed and Accuracy

As a matter of fact, the yield provided by present-day sequencing technologies rules out the possibility of using "traditional" sequence alignment programs like BLAST: neither the original BLAST nor any of its several reimplementations would be able to provide enough throughput when processing HTS data. A typical short-read aligner run with default parameters provides a performance of several tens of millions of reads mapped per hour per CPU, which is orders of magnitude faster than BLAST.

However, as previously discussed, such speed necessarily comes at the price of accuracy: in particular, the high sensitivity provided by BLAST and other similar tools (for instance, Kent 2002) must be sacrificed. When aligning proteins with BLAST, it is common to take into account matches having a sequence similarity as low as the 40%; on the other hand, finding such distant hits with HTS mappers would be completely impossible, since only candidate matches differing by a few mismatches from the original sequence are considered (as mentioned before, typically they must have with the read a sequence similarity of more than the 95%). Although the assumption of high similarity is partially justified by the considerations worked out in Sect. 7.2.1, the reader should be aware that such a limitation cannot be lifted, since mappers are designed and optimized to work in the region of a small number of mismatches.

In addition, while traditional alignment tools usually output the full list of significant hits, sorted by decreasing relevance, most HTS alignment programs embed arbitrary rules as to which results to report. Defining as *stratum* the set of all the matches in the reference having the same number of mismatches from the read, most mappers report only (a subset of) the best stratum, that is only (a subset of) the set of matches having the minimum number of differences from the read; other methods (for instance, Li and Durbin 2009) implement more complicated rules, but in general such rules are arbitrary and hardwired – they are decided once for all by the algorithm implementor, and the user is not permitted to change them. In fact,

another substantial difference with respect to traditional alignment tools is that mappers usually do not offer the user the possibility of a full fine-tuning of alignment parameters: in some cases (notably, Langmead et al. 2009, which implements a seed-based search, but also many other programs) one is not even able to freely specify the desired number of mismatches.

Finding out how the accuracy of the results depends on the particular set of matches reported by each mapper is still an active research topic (see for instance, Ribeca and Valiente 2011; RGASP). In particular, the algorithmic definition of "best" matches to be reported has a strong impact on the definition of uniquely-mapping reads, which we will examine in the next section.

7.2.3 Not All Reads are Created Equal

Quite interestingly, the short reads obtained after a HTS experiment are not all on the same footage: in particular, some of them can be mapped uniquely to the reference, while for other reads several regions in the genome can be identified which show a high sequence similarity to them. In the latter case, the reads are said to be *multiply mapping*.

The problem with multiply mapping reads is that their assignment to the reference is ambiguous, thus rendering them useless for applications – like RNA-seq quantification – which require each read to be attributed to a precise location due to normalization/counting purposes. In fact, in such applications the traditional "solution" to the problem is just to discard multiply mapping reads, although doing so kills a relevant part of the signal and introduces biases in the subsequent analysis (some locations in the genome are intrinsically multiple, since duplicated genes and regions are frequent in most genomes, see Sect. 7.2.4.4).

It should be emphasized that the concept of uniquely mapping reads is ill-defined if one omits to specify the parameters which have been used for the alignment: in fact, there is no such a thing like a "uniquely mapping read" in absolute terms, since the more the mismatches allowed, the more the matches that will be found (and eventually, if enough mismatches are tolerated, the read is bound to match at each position in the genome). This fact can create hidden incompatibilities between different analysis protocols, since typically the definition of what is considered "unique" by a mapper does not coincide with the corresponding definition used by another mapper – the reason being that very likely the algorithms employed by the two mappers will not be exactly the same, thus producing different sets of matches even if formally the alignment parameters for both methods coincide. Last but not least, the reader should also be aware that it is not possible to decide whether a read is unique or not without an exhaustive mapping algorithm (see Sect. 7.3.3.1).

Even if a fraction of the reads cannot be assigned unambiguously to the genome only on the basis of their sequence, it is sometimes possible to get to a better decision considering additional information, like the quality of the base calling, paired-end information or the correlation between the positions of all the reads in the dataset. We examine such possibilities in the following section.

7.2.4 Additional Information Which Can Help

Additional information other than the read sequence can sometimes be used to obtain more accurate alignments. Ideally, as many of the following sources of information as possible should be taken into account by the mapping program/pipeline, if we are to maximize the quality of the obtained results.

7.2.4.1 Qualities

As a result of the base-calling process, most modern sequencing machines accompany in their output the read sequence with a list of *qualities*, usually one per called base. Qualities are small numbers, typically between 0 and 40 (encoded as ASCII characters in FASTQ-like formats), which express the probability that the calling of the corresponding base went wrong. One possible encoding for the qualities is the so-called Phred scale (Ewing and Green 1998), defined by $q = -10\log_{10} e$, or $e = 10^{-q/10}$, or: a quality of 30 corresponds to an error of 1/1,000, a quality of 40 to an error of 1/10,000, and so on.

Several mappers (for instance, Ribeca 2009) can take advantage of the quality scores to obtain more precise alignments. The idea is to use the quality declared by the machine for each base as a guide, allowing an easier replacement of a nucleotide in the read when the quality is lower – or alternatively, considering as "best" alignment the one such that the highest possible number of mismatches coincide with nucleotides in the read having a low-quality base calling. Quality information is usually quite effective in improving the success rate of the alignment.

7.2.4.2 Paired-End Information

Paired-end information (i.e., the ability of a HTS platform to sequence both ends of a long DNA molecule rather than just one, possibly leaving an unsequenced insert in the middle) can be useful to alleviate the problem of multiply mapping reads. When one of the ends of the molecule maps unambiguously, and the second one falls into a repetitive region, it is sometimes possible to use the statistical information derived from the dataset about the insert size (i.e., about the typical distance between the two ends) to select an unambiguous mapping for the second end of the molecule too.

7.2.4.3 The Pileup

When they can be made at all, several choices about the read (for instance, where to assign it in case it aligns to multiple locations in the genome) require external information which is not available when mapping the read separately from the rest of the dataset.

In particular, one can successfully answer several of the most interesting high-level biological questions only after having performed a global analysis of the dataset. Some examples follow.

- The identification of single-nucleotide polymorphisms (also known as *SNP calling*) at any given position in the genome is only possible after the *pileup* at that locus (i.e., the nucleotide/quality readings at that position as determined by all the reads encompassing the locus) is known. The pileup specifies a vector of nucleotide frequencies/qualities for each position in the genome; out of it, an appropriate Bayesian model can be used to derive the genotype of the locus.

  Very similar considerations hold for *variant calling*, which consists of detecting in the genome under consideration structural variations (like insertions, deletions and more complicated rearrangements) with respect to the reference. Especially haplotype-specific variants (for instance, small deletions which are present in only one of the two copies of a chromosome of a diploid genome) can be called with a reasonable certainty only after a pileup-level analysis of the whole dataset: depending on the allele they come from, some reads might support the version present in the reference, with other ones highlighting the variant. The only way to obtain a reliable understanding of the situation, then, will be to run on the set of all the reads spanning the locus under consideration a program able to perform *local realignment* (see for instance, McKenna et al. 2010 or Li et al. 2009a), which will take care of distinguishing and calling both haplotypes.
- In some cases it is possible to rescue reads mapping multiple times by excluding some of the possible alternatives. For instance, among the various matches one might prefer the ones which are surrounded by more reads, thus discarding isolated hits. Once again, performing this assignment is possible only after all the reads have been aligned.
- When mapping RNA-seq data, some reads will be *spliced reads*, that is reads which span the junction between two exons. Such reads will map to the transcriptome annotation but not to the genome, since the separating (and possibly very long) intron has been removed by the splicing mechanism operating in the cell. In principle, although with lower probability (and depending on the length of the read), even more than one junction could be present in the same read. When the transcriptome annotation is not known and one is interested in performing a de-novo identification of splice junctions, some programs called *splice aligners* exist which are able to identify the exonic blocks and map them back to the genome (for instance, Ribeca 2009). However, one read is usually not enough to get a high confidence about the effective presence of the junction, and hence in this case too dataset-level analysis is essential to validate the candidate splice junctions.
- Techniques like bisulfite sequencing, ChIP-seq, micrococcal nuclease digestion, and histone methylation probes allow to perform genome-wide assays about chromatin status. However, for each of these signals one is usually interested in locating regions which exhibit intensity peaks given by the superposition of many reads: once more, this kind of analysis can only be performed if the whole dataset is considered.

The list just presented is far from being complete.

7.2.4.4 The Mappability

As mentioned above, not all the loci in the genome show the same degree of uniqueness: some of them are the result of a recent duplication event, and as a consequence bear a high sequence similarity with other regions in the genome. Interestingly, the degree of uniqueness at each position (or *k-mappability*, defined as the inverse of the number of times the *k*-mer starting at each position appears in either the genome or its reverse complement) is a property of the organism: it varies from genome to genome, and when the reference for the organism is known it can be determined a priori before mapping the reads (it should also be emphasized that if mismatches are considered, the mappability of a read is in general different from the mappability of the region it maps to).

Mappability should be taken into account in many situations, notably when designing the parameters of a HTS experiment (to maximize the number of uniquely mapping reads and the access to some specific regions of the genome), and when quantitative studies are being performed (like in ChIP-seq, where this quantity first came into widespread use, see Rozowsky et al. (2009), and in RNA-seq). Some programs exist which allow to easily compute the mappability (see for instance, Ribeca 2009); precomputed tracks can sometimes be accessed through the UCSC genome browser (Karolchik et al. 2007).

7.3 The Algorithmic Standpoint

In this section, we cursorily review the most important algorithmic techniques which have so far been used to implement fast short-read mapping. As a disclaimer, we point out that at the time of this writing an impressive number of mapping programs has been developed [about 40 are listed on the Wikipedia page (Sequence alignment software) on sequence alignment programs, which surely represents an underestimation], and hence it would be impossible to give in this limited space a completely comprehensive survey of all the algorithmic solutions adopted.

In addition, we believe that such a detailed description would be excessive in this context, since here, as previously explained, we are mostly interested in focusing on the user's standpoint: the purpose of this section is not to turn the reader into in an expert in the design of alignment algorithms, but rather to enable them to better appreciate the strengths and limitations of each method when building optimal mapping pipelines. The reader interested in a more technical introduction is referred to Li and Homer (2010).

7.3.1 Possible Indexing Strategies

When aligning sequence reads to a reference genome, the first computational step usually consists of transforming part of the data into a suitable form, so that it

becomes easily searchable afterwards. In general, three different high-level indexing strategies are possible:

1. *Indexing the reads and scanning the genome*. The reads are separated into groups – compatibly with the limitations set by the amount of available memory – and indexed. Subsequently, the genome is scanned to find occurrences of substrings of the genome in the reads. Examples of such a setup are ELAND (see Sect. 7.3.3.1) and Lin et al. (2008).
2. *Indexing the genome, and scanning the reads*. The genome is preindexed. Subsequently, one or more reads at the time are scanned, to find their occurrences in the genome. Examples of such a setup are Langmead et al. (2009), Li et al. (2009b), Li and Durbin (2009), and Ribeca (2009).
3. *Indexing the genome and the reads*. The genome and the reads are indexed together (either over and over again each time, or combining a precomputed index of the genome with some newly computed index for the reads) to find common sequences occurring both in the reads and in the genome. Examples of such a setup are Malhis et al. (2009) and Hach et al. (2010).

No matter which high-level indexing strategy is chosen for the data, at low level any algorithm will ultimately rely on some more or less sophisticated string-indexing technique to do the job. Although in principle different choices would be possible, from the point of view of the mapper developer schemes 1 and 3 are usually quite straightforward to implement using *hash tables*, while scheme 2 lends itself very well to be implemented as an *FM-index*. We will examine both low-level string-indexing frameworks in the following section.

7.3.2 Implementation Techniques

Hash tables are employed since a very long time in the alignment of biological sequences, BLAST possibly being the program which popularized most their use. The FM-index came into widespread use in bioinformatics only a few years ago; however, it is the heart of some of the most innovative and most used HTS mappers (for instance, Langmead et al. 2009; Li and Durbin 2009; Li et al. 2009b; Ribeca 2009), thanks in particular to the small memory footprint it requires, and to the high performance it can deliver in the parameter region usually considered when aligning short reads.

One key problem which needs to be solved in all indexing/alignment schemes is that of efficiently accommodating mismatches in the searches. This is a complicated technical problem, which is still the object of a very active research in the field.

7.3.2.1 Hash Tables

Internally, computers represent character strings as numbers; however, such numbers are huge – much larger than the size of available memory. The basic idea behind hash tables is that of transforming a character string into a small(er) number, like a

32-bit or 64-bit integer, via a suitable *hashing function H*, which ideally should be as clash-free as possible. In computer science, this concept is used to attribute to each string a reproducible offset in a table, so that it is possible to implement sets and dictionaries; in such a context, the choice of a good general hashing function is a difficult one, and still an active research topic. In sequence mapping, the idea of encoding a string as a number is interesting precisely because it offers a quick and efficient way of checking whether two (sub)sequences might be the same or not: for them to be, the value of their hashes must be identical. In practice, in most mapping implementations there is no need for a sophisticated hashing function, since direct *binary encoding* for DNA can be used instead: each nucleotide can in principle be encoded using 2 bits, and hence a single 64-bit integer can accommodate 32 bases, offering in addition all the advantages of *bit parallelism* – using some specialized techniques, one can perform in parallel tasks like counting the number of mismatches, expressing them in terms of very few operations on 64-bit integers.

In a typical HTS mapping setup based on hash tables, the workflow would then be as follows:

1. One takes the sequence to be indexed, scans it via a sliding window of dimension k (k typically being the number of symbols fitting in a word), computes the hash H for each k-mer $\boldsymbol{K}_i$, and installs the k-mer in the table at position $H(\boldsymbol{K}_i) \bmod d$, being d the size of the hash table (which is usually fixed to the length of the text to be indexed). As in ordinary hash tables, some additional algorithm also needs to be specified to resolve collisions.
2. Once the table has been created, the query is scanned in a similar way, using a sliding window of dimension k, and matches for the k-mer $\boldsymbol{Q}_j$ under analysis are looked up in the table at position $H(\boldsymbol{Q}_j) \bmod d$. If a sufficient number of k-mers of the query is found in the table, a match is called (in particular, matching k-mers can be used as *seeds* for the search).

The scheme just presented does not take mismatches into account. In hash table-based setups, one can make provision for them in different ways:

1. By replicating hash tables – for instance, with the technique of the *spaced seeds*. A seed generalizes the concept of k-mer: it is a binary pattern specifying that either the next character should be considered to compute the hash (case noted with a 1) or that it should be skipped (case noted with a 0). With this notation, for instance, when $k=6$ the case of an ordinary hash table, which is filled with contiguous k-mers, would correspond to the seed 111111. On the other hand, the seed 110000110011 would mean "compute the hash by considering 2 characters, skipping 4, considering another 2, skipping 2 and considering another 2": in this case one would use a sliding window of 12 characters, but only 6 of them would contribute to the hash.

   In such a setup, one would create several hash tables, each one corresponding to a different seed: given a predetermined number of mismatches, one can find relatively small sets of seeds such that, if the reference has been installed in all the tables, a string differing from it by less than the given number of mismatches

will be found in at least one of the tables. The initial versions of ELAND (see Sect. 7.3.3.1) are an example of this strategy. The reader should be aware of the fact that finding an optimal (i.e., smallest) set of seeds for any given read length and number of mismatches is a complicated problem.
2. By looking for more than one single k-mer in the query at the same time – for instance, the technique of q-gram filtering introduced in Jokinen (1991): in a query of length l and at most m mismatches, one is bound to find at least $(l+1)-(m+1)$ k exact substrings of length k which also appear in the reference and a match can be excluded if not enough common k-mers are found.

In general, the following properties are shared, at least up to some extent, by all the implementations of indexing based on hash tables.

- Pros: In a framework based on hash tables, it is relatively easy to implement search algorithms which are able to accommodate a large number of mismatches, and at the same time scale quite well with the number of mismatches – that is, such that the running time when searching with more mismatches will not be much higher than when searching with less mismatches. Also, generating an index is usually fast.
- Cons: Hash tables are bulky, in particular when mismatches are taken into account; this usually translates to longer search times with respect to other methods. In addition, hash tables are not very flexible when frequent changes in search parameters are needed: since such parameters determine the content of the tables, changing them usually involves the recomputation of the index.

Combining the two points, one can see that a typical implementation based on hash tables has a sweet performance spot when searches with many mismatches are performed (it is slower when few mismatches are considered, but scales better when moving to a larger number of mismatches).

7.3.2.2 FM Indices

Ferragina–Manzini indices (Ferragina and Manzini 2000) were invented in relatively recent years, in the attempt of obtaining a compressed representation of large amounts of text which would nonetheless provide fast searching capabilities for relatively small queries. In the original intentions of the authors, such a model would suit perfectly the needs of Internet search engines; in practice, it proved to be equally appropriate for mapping short sequence reads to large genomic references.

FM indices are the outcome of the evolution of other data structures (like *suffix trees* and *suffix arrays*) which had been proposed and used in the past to perform fast searches on character strings. Albeit such structures provide appealing algorithmic properties [in particular suffix trees, whose multiple benefits in terms of the solution of biological problems are examined in detail in Gusfield (1997)], they are usually too bulky to store large amounts of sequence, as one needs to do in biology: for instance, a typical suffix tree implementation requires about 15 bytes/base, with the

result that an index for the human genome would need about 45 GB of memory to be efficiently stored and queried.

The FM-index is based on the Burrows–Wheeler transform (Burrows et al. 1994) (BWT), a reversible permutation of the original text string which lends itself very well to excellent compression [for instance, BWT compression is exploited by the popular archiving tool `bzip2` (Seward 1998)]. In short-read mapping, however, compressibility is not a major issue: one is usually happy with a simpler packed representation of the BWT, which allows for a compact storage of the index anyway (the typical amount of space required being between 0.5 and 2 bytes/base), and offers better performance.

A major technical point about the FM-index is that, due to its very definition, the BWT is intimately connected to the suffix array. In fact, once the BWT is available it is also possible to emulate the suffix array at almost no additional cost in terms of storage: to this end, it is enough to sample the suffix array at regular intervals. Furthermore, one can locally invert the BWT to reproduce the indexed text, either fully or in chunks: this implies that when using an FM-index there is no need to separately store the original text, thus effectively turning the FM-index into a *self-index*.

The combination of all these factors has an interesting consequence: thanks to the technique called *backward search*, it is possible to perform very fast exact searches in the index. In fact, the backward search takes advantage of the contemporary presence of the BWT and the (emulated) suffix array, allowing to express a range search in the suffix array in terms of fast character-counting queries on the BWT. Remarkably, the number of counting queries required will be proportional to the length of the query, and (theoretically, if one neglects the cost of memory access) independent on the size of the indexed text; the search is said to be "backward" since the characters in the query are added right-to-left rather than left-to-right.

The scheme just presented does not take mismatches into account. In fact, one can make provision for them in different ways, as follows:

1. By expressing the mismatched search in terms of many exact searches. Conceptually, this can be done by modifying the query in all the possible ways compatible with the parameters of the search: for instance, one can perform a search with two substitutions by searching for all possible strings which can be obtained from the query if one operates on it at most two substitutions. In a practical implementation, one would likely express such set of exact searches as a tree.

   This technique has evident limitations: due to the combinatorial explosion in the number of possibilities which should be considered, it is usually viable only for a small number of mismatches. Even in the latter case, suitable pruning criteria should be provided, so to avoid that the huge search space needs to be explored in its entirety.

2. By adopting a seed-and-extend strategy. According to the parameters of the search, some exact substrings of the query are searched for in the index, and added to a list of candidate matches; the candidates are then examined around their positions for compatibility with the complete query, and possibly discarded.

The interested reader can find in Navarro and Baeza-Yates (2000) a more detailed description about possible ways of accommodating mismatches into FM-indexes. It should be noted that (unlike what happens with hash tables) in such searching schemes one does not modify the index to incorporate mismatches: the extension is always done algorithmically, by performing a more complicated search.

In general, the following properties are common, at least up to some extent, to all mapping implementations based on the FM-index.

- Pros: Thanks to its design properties, the index is remarkably compact. In addition, since its contents do not depend on the parameters of the search, in a framework based on the FM-index it is relatively easy to implement search algorithms such that parameters can be changed freely. Finally, and owing to the fact that the FM-index provides very fast exact searches, one can usually implement very fast searches with mismatches as long as the number of mismatches is low.
- Cons: Generating the index is slow, since it requires the computation of the BWT, and this is a global operation on the whole content of the index. Also, it is quite difficult in this framework to design search algorithms which scale well when moving to a large number of mismatches.

Combining the two points, it is easy to see that FM-indexes usually offer special advantages when searches with a small number of mismatches need to be performed.

7.3.3 Indexing/Searching Algorithms are Not All Equivalent

In spite of what many nontechnical people often think, it is far from being true that "all alignment programs essentially do the same job." The following points are noteworthy.

1. The different basic indexing schemes are sometimes not completely equivalent. For instance, while it is possible for an FM-index to completely emulate the features of a hash table (in particular, it is possible to perform very fast exact searches for any k-mer in the reference genome), the converse is not true (hash tables have a minimum granularity determined by the k-mer size chosen at indexing time).
2. Each basic indexing scheme typically lends itself very well to implement in a simpler way some particular feature or class of algorithms. For instance, as mentioned above, the use of FM-indices will have as a natural consequence the possibility of producing a well-packed representation of the genomic reference; similarly, it is straightforward to implement seed-based searches with hash tables, since hash-table indexing is based on k-mers, and the same k-mers can naturally be seen as seeds.

On the other hand, in many cases algorithms which appear less "natural" in a given setup can be implemented perfectly well in another, although at the price of a possibly much harder work: apart from some real differences, as those highlighted

under the previous point in the list, many algorithms are actually independent of the basic indexing scheme used, and the task of the algorithm designer is precisely that of abstracting from the details of the underlying indexing engine.

In conclusion, different HTS mappers will tend to share common features depending on the indexing technique they are based upon; however, such similarities are expected to become less apparent as programs reach maturity, thus benefiting from more complicated and more refined implementations. As a general rule, one should be extremely suspicious when hearing statements about a mapper being better due to its "superior" indexing approach: what ultimately matters to the user are not the personal beliefs of the algorithm writer, but the technical specifications of the produced tool. As far as the user's standpoint is concerned, the possibility of performing searches with more mismatches, the better throughput and the ability of satisfying the requirements imposed by the biological problem at hand should be the only motivations behind the choice of an alignment program.

7.3.3.1 Finding Them All: Exhaustiveness

Although different algorithmic techniques may offer different advantages depending on the mapping parameters one intends to employ, and no matter which low/high level indexing scheme is used, some properties are fundamental to assess the acceptability of an implementation. Above them all is sensitivity: the larger the number of matches within the same stratum found by the mapper, the better its sensitivity.

As a matter of fact, a special class of algorithm exists: *exhaustive* short-read mappers are those able to find all the existing matches within the specified number of mismatches. It is worth noting that when their platform was producing reads of 32–36 nt, Illumina/Solexa used to provide an exhaustive aligner: at that read length, the initial versions of ELAND (the default mapper supplied with the standard Illumina pipeline) were capable of finding all the matches up to two nucleotide substitutions.

In fact, the vast majority of mapping algorithms (notably, Langmead et al. 2009; Li and Durbin 2009) sacrifice exhaustiveness for speed, in particular when longer reads and more mismatches are considered; another common situation is that some form of exhaustiveness is provided, but at the price of not being able to freely tune search parameters. To the best of the author's knowledge, as of this writing only a few short-read aligners (for instance, Lin et al. 2008; Hach et al. 2010; Ribeca 2009, albeit with major differences in memory usage and performance) are able to report all the matches as far as nucleotide substitutions are concerned.

The consequences of the lack of exhaustiveness are often overlooked: in particular, we emphasize that running an exhaustive mapping algorithm is the only way to decide whether a read maps uniquely or not [it goes without saying that the "alignment score" provided by some mappers like Li and Durbin (2009) cannot be considered a substitute for the inability to enumerate all the matches]. In this respect, one should be aware that a high sensitivity is not enough to guarantee a good

discrimination of unique matches (see Ribeca and Valiente 2001, where it is shown that in several situations even a small fraction of missed matches can be enough to misclassify the 20–30% of the reads).

7.4 The User's Standpoint

What is the correct workflow to follow to set up an optimal mapping pipeline for a given biological experiment? In this section we take advantage of the facts so far examined, and formulate a checklist which one should follow when trying to optimally set up his/her own HTS mapping pipeline.

7.4.1 Analyzing the Problem: What are the Requirements?

First of all, different HTS platforms, and/or different biological problems, may require radically different setups in alignment parameters. Typical examples are the following ones:

- The type and number of alignment errors heavily depends on the technology used. For instance, while Illumina/Solexa reads, as mentioned in Sect. 7.2.1, virtually show no indels and a relatively low number of single-nucleotide substitutions due to sequencing errors (which tend to become relatively more frequent toward the right end of the read), Roche/454 has a very high incidence of short indels (they tend to happen in correspondence of the presence of homopolymers in the original sequence). Reads produced by other platforms [like colorspace SOLiD (Metzker 2010), Pacific Biosciences (Eid et al. 2009), and Complete Genomics (Drmanac et al. 2010)] require even more complicated, specifically tailored, alignment strategies: to mention just one, each Complete Genomics read is made of four different parts, separated by three inserts of different sizes. In general, mapping methods and parameters should be chosen accordingly to the platform being used.
- The type of the experiment being run – and the protocol employed – can contribute as well to the determination of the alignment method to use. For instance, as mentioned in Sect. 7.2.4.3, RNA-seq data usually requires specialized tools like splice mappers; bisulfite sequencing demands an even more complicated alignment strategy, since the sequencing protocol transforms unmethylated C nucleotides to Ts, thus producing reads which do not belong to the original reference even in the absence of sequencing errors (see for instance, Li and Homer 2010 for a more detailed explanation). The support for such extended alignment methods is not always provided by all mappers.
- Last but not least, biology itself can play a major role in the choice of alignment methods and parameters. For instance, an unusually large number of mismatches might be necessary in several scenarios (when studying RNA editing, or when aligning to a distant species because the reference for the organism under

investigation is not known); sometimes such a number of mismatches will be excessive (unsupported or leading to too long alignment times) for a particular mapping algorithm.

Hence, a first screening of the available tools will eliminate the ones which are not able to cope with the requirements imposed by the problem at hand. Depending on the situation, the categories of constraints we just presented might be enough to already rule out most mapping tools.

7.4.2 Separating Planes

An important but sometimes overlooked fact is that many mappers (in particular, those suitable for processing the short reads produced by RNA-seq) often embed into their alignment algorithms assumptions about biology. This is due to different and sometimes legitimate reasons – for instance, increasing the S/N ratio and/or optimization: imposing rules on the splice-site consensus obeyed by sequences flanking exons acts as a powerful filter to avoid considering too many candidates when doing spliced alignment, and such candidate alignments should be removed anyway at some later point. This observation should come as no surprise: already in BLAST, assumptions about biology (substitution matrices, penalties for gap opening and so on) get embedded in the alignment engine as parameters given to the linear programming stage; depending on the values of such parameters, some parts of the alignment space are either explored or silently discarded, with consequences on sensitivity and performance. When considering mappers, however, the setup can be much more radical than the one provided by BLAST: sometimes the additional biological assumptions are so deeply entangled into the algorithms that neither can one distinguish them as separate bits of data anymore, nor modify the relevant parameters at user level.

This situation notwithstanding, the reader should be aware that the two stages of finding alignments and selecting the best candidate match(es) on the basis of biological information are separate ones, at least from a logical point of view. In fact, there are exactly three conceptual processing stages when aligning short reads:

1. *The algorithmic plane*. Speaking from a purely combinatorial standpoint, there is no such a thing as a "biology-aware" alignment: given the maximum number of mismatches/edit distance specified by the user, either a match is there (and its distance from the reference can be measured) or it is not. If it is, it is also by definition a valid alignment, and in principle it must be found and output by any mapper – albeit sometimes this does not happen in practice, as seen in Sect. 7.3.3.1.
2. *The biological plane*. After all the alignments compatible with the parameters of the search have been obtained, it might be worth filtering out some of them due to biological considerations (for instance, as mentioned before, one might discard the spliced alignments such that the sequences flanking exons are not compatible with splice-site consensus).

3. *Postprocessing*. The reader should be warned that an optimal data analysis will very likely require a good deal of postprocessing. In particular, pileup-level analysis as mentioned in Sect. 7.2.4.3 might be essential to achieve the goal of a deeper understanding of the data.

Once the above logical scheme is clear, it will not be difficult to estimate the interplay between the different planes given by each of the methods, and prefer the ones which are flexible and tunable enough to be adapted to the desired workflow.

7.4.3 What is Important, and What is Not

Obvious as it may seem, to obtain optimal results accuracy should always be preferred to speed. As of this writing, many mappers exist which allow the analysis of a whole HTS experiment on a single CPU core in a few days; hence, hard choices which imply a tradeoff between mapping quality and speed are likely to be relevant only to large sequencing centers, not to single biological laboratories.

In consequence, when selecting a mapping tool the user should first restrict his/her choice to the ones able to comply with the accuracy requirements dictated by the problem, as mentioned in Sect. 7.4.1; only after quality has been enforced shall the surviving candidates be sorted based on their speed. In particular, and whenever available for the specific problem being considered, exhaustive alignment programs as described in Sect. 7.3.3.1 should always be preferred.

7.5 Conclusions

In this chapter, we tried to present a fairly complete description of the state-of-the-art of short-read mapping, focusing more on the user's standpoint (how to setup an optimal analysis pipeline) rather than on the algorithm developer's. However, from time to time some knowledge of the internals and the implementation of each mapping method will also be needed to better understand the possibilities and the limitations of the available tools; and we tried to provide such information as well.

In general, our recommendations follow the usual guidelines which should be kept in mind when designing a sound setup for the quantitative analysis of any experiment: one should select a short-read mapper only *after* having clarified what the technical requirements imposed by the dataset in need of analysis are (mainly the number and the type of mismatches as determined by variants and sequencing errors), discarding all the alignment methods which are unable to comply with such technical specifications. If after this evaluation it turns out that several mappers would still be able to do the job, one should first and foremost give preference to the most accurate ones (and in particular, if possible, to those providing exhaustive mapping); only as a last criterion shall mapping speed influence a wise choice.

In spite of the claims of some mapper developers, the problem of easily setting up and routinely obtaining a fast high-quality analysis of short reads is still far from being solved, in particular when all possible biological scenarios/protocols are considered. We hope that the information provided in this chapter will help the reader to face the intricacies and fallacies of modern HTS short-read alignment.

References

M. L. Metzker. Sequencing technologies – the next generation. *Nature Reviews Genetics*, 11(1):31–46, January 2010.

J. M. Rothberg and J. H. Leamon. The development and impact of 454 sequencing. *Nature Biotechnologies*, 26(10):1117–1124, 2008.

S. F. Altschul, W. Gish, W. Miller, E. W. Myers, and D. J. Lipman. Basic local alignment search tool. *Journal of Molecular Biology*, 215(3):403–10, October 1990.

H. Li and R. Durbin. Fast and accurate short read alignment with Burrows-Wheeler transform. *Bioinformatics*, 25(14):1754–1760, 2009.

W. J. Kent. BLAT: The BLAST-like alignment tool. *Genome Research*, 12(4):656–664, 2002.

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

P. Ribeca and G. Valiente. Computational challenges of sequence classification in microbiomic data. *Briefings in Bioinformatics*, April 2011.

The RGASP: RNA-seq read alignment assessment. http://www.gencodegenes.org/rgasp/rgasp3.html.

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

P. Ribeca. GEM: GEnomic Multi-tool. http://gemlibrary.sourceforge.net, 2009.

A. McKenna, M. Hanna, E. Banks, A. Sivachenko, K. Cibulskis, *et al.* The Genome Analysis ToolKit: a MapReduce framework for analyzing next-generation DNA sequencing data. *Genome Research*, 20(9):1297–303, September 2010.

H. Li, B. Handsaker, A. Wysoker, T. Fennell, J. Ruan, *et al.* The Sequence Alignment/Map format and SAMtools. *Bioinformatics*, 25(16):2078–9, August 2009.

J. Rozowsky, G. Euskirchen, R. K. Auerbach, Z. D. Zhang, T. Gibson, *et al.* PeakSeq enables systematic scoring of ChIP-seq experiments relative to controls. *Nature Biotechnologies*, 27(1):66–75, January 2009.

D. Karolchik, A. S. Hinrichs, and W. J. Kent. The UCSC genome browser. *Current Protocols in Bioinformatics*, Chapter 1:Unit 1.4, March 2007.

Wikipedia: Sequence alignment software. http://en.wikipedia.org/wiki/Sequence_alignment_software.

H. Li and N. Homer. A survey of sequence alignment algorithms for next-generation sequencing. *Briefings in Bioinformatics*, 11(5):473–483, 2010.

H. Lin, Z. Zhang, M. Q. Zhang, B. Ma, and M. Li. ZOOM! Zillions of oligos mapped. *Bioinformatics*, 24(21):2431–7, November 2008.

R. Li, C. Yu, Y. Li, T.-W. Lam, S.-M. Yiu, *et al.* SOAP2: an improved ultrafast tool for short read alignment. *Bioinformatics*, 25(15):1966–7, August 2009.

N. Malhis, Y. S.-N. Butterfield, M. Ester, and S. J.-M. Jones. Slider – maximum use of probability information for alignment of short sequence reads and SNP detection. *Bioinformatics*, 25(1):6–13, January 2009.

F. Hach, F. Hormozdiari, C. Alkan, F. Hormozdiari, I. Birol, *et al.* mrsFAST: a cache-oblivious algorithm for short-read mapping. *Nature Methods*, 7(8):576–7, August 2010.

P. Jokinen and E. Ukkonen. Two algorithms for approxmate string matching in static texts. *Mathematical Foundations of Computer Science 1991*, pages 240–248, 1991.

P. Ferragina and G. Manzini. Opportunistic data structures with applications. In *Proceedings of the 41st Symposium on Foundations of Computer Science (FOCS 2000)*, pages 390–398, 2000.
D. Gusfield. *Algorithms on strings, trees, and sequences*. Cambridge University Press, 1997.
M. Burrows and D. J. Wheeler. A block-sorting lossless data compression algorithm. Technical Report 124, Digital Equipment Corporation, Palo Alto, CA, 1994.
J. Seward. Bzip2 and libbzip2: a program and library for data compression. http://sources.redhat.com/bzip2, 1998.
G. Navarro and R. Baeza-Yates. A hybrid indexing method for approximate string matching. *Journal of Discrete Algorithms*, 1(1):205–239, 2000.
J. Eid, A. Fehr, J. Gray, K. Luong, J. Lyle, *et al.* Real-time DNA sequencing from single polymerase molecules. *Science*, 323(5910):133–138, 2009.
R. Drmanac, A. B. Sparks, M. J. Callow, A. L. Halpern, N. L. Burns, *et al.* Human genome sequencing using unchained base reads on self-assembling dna nanoarrays. *Science*, 327(5961): 78–81, January 2010.

Chapter 8
DNA–Protein Interaction Analysis (ChIP-Seq)

Geetu Tuteja

Abstract ChIP-Seq, which combines chromatin immunoprecipitation (ChIP) with high throughput sequencing, is a powerful technology that allows for identification of genome-wide protein–DNA interactions. Interpretation of ChIP-Seq data has proven to be a complicated computational task, and multiple methods have been developed to address these challenges. This chapter begins by describing the protocol for ChIP-Seq library preparation and proper experimental design, without which computational tools would not be able to accurately capture in vivo interactions. Following a section on raw data pre-processing and data visualization, using Illumina Genome Analyzer output files as examples, general approaches taken by peak-calling tools are described. GLITR, a powerful peak-calling tool that utilizes a large set of control data to accurately identify regions that are bound in ChIP-Seq data, is then explained in detail. Finally, an approach for functional interpretation of ChIP-Seq peaks is discussed.

8.1 Introduction to ChIP-Seq Technology

The regulation of gene expression in mammals is a complex and highly orchestrated process that is crucial for allowing cell types to achieve their specialized functions. The mechanisms behind this regulation, particularly the transcription factor (TF) proteins that bind DNA targets to regulate gene expression, have been intensely studied over the last several decades. Until recently, TF/DNA complexes were typically studied in vivo on an individual target basis using Chromatin Immunoprecipitation (ChIP), which was first described in 1988 (Solomon et al. 1988). ChIP is a useful technique to determine if a protein is bound to tens of

G. Tuteja (✉)
Department of Developmental Biology, Stanford University, Stanford, CA, USA
e-mail: geetu@stanford.edu

N. Rodríguez-Ezpeleta et al. (eds.), *Bioinformatics for High Throughput Sequencing*,
DOI 10.1007/978-1-4614-0782-9_8,

potential target regions, but in order to determine where the protein is bound at a much larger scale, other approaches must be utilized.

ChIP-on-chip, which combines ChIP with microarray technology, allows for the large-scale identification of TF binding targets. The first ChIP-on-chip experiment was described around ten years ago, and was used to identify the binding sites of two TFs in yeast (Ren et al. 2000). This study and other early ChIP-on-chip studies were performed using microarrays with thousands of spotted PCR amplicons representing genomic regions, such as promoter regions, where a TF might bind (Ren et al. 2000; Simon et al. 2001; Wyrick et al. 2001; Friedman et al. 2004; Harbison et al. 2004; Le et al. 2005; Rubins et al. 2005). Typically, PCR amplicons are between 500 and 2,000 base pairs. This technology may be used to determine if the TF of interest is binding within or near the PCR amplicon, but does not determine the precise location of the TF binding site. More precise ChIP-on-chip platforms, tiling arrays, emerged soon after spotted arrays, and these increased the coverage of the genome and allowed for higher resolution in identifying TF binding sites. Tiling arrays contain short oligonucleotides, which can either overlap, are end-to-end, or have a regularly spaced gap. One company that manufactures these arrays is Agilent, whose mammalian tiling arrays contain over 200,000 60-mer oligonucleotide probes spaced 100–300 base pairs apart (www.agilent.com). While tiling arrays offer increased resolution compared to cDNA spotted arrays, the statistical treatment necessary to identify enriched tiles, and therefore the region where the TF is bound, is far more complicated (Buck and Lieb 2004; Mockler et al. 2005; Royce et al. 2005). All microarray technologies also share other drawbacks, including dye biases and cross-hybridization issues (Buck and Lieb 2004; Royce et al. 2005).

Recently, ChIP-Seq technology has started replacing ChIP-on-chip technology. ChIP-Seq has become possible with the introduction of high throughput sequencing technologies, which can rapidly sequence all of the DNA fragments in a sample. Different platforms available for performing ChIP-Seq experiments include the Illumina Genome Analyzer, the Applied Biosystems SOLiD system, the Roche 454 Life Sciences platform, the HeliScope instrument from Helicos, and the Illumina HiSeq. Because all fragments in a ChIP experiment can be sequenced using these platforms, TF binding targets are not limited to those oligonucleotides or PCR amplicons on a microarray. Additionally this technology allows more precise identification of TF binding sites, and eliminates issues caused by cross-hybridization. Another benefit of ChIP-Seq is that it can be used for any species with a sequenced genome, and it is not limited to only those species for which a microarray has been produced. Because most published ChIP-Seq experiments have been performed using the Illumina Genome Analyzer, this platform will be the focus of this chapter. However, the details discussed can be applied to many of the other platforms.

The first ChIP-Seq experiments carried out on the Illumina Genome Analyzer were performed only 4 years ago (Johnson et al. 2007; Robertson et al. 2007). These studies demonstrated that ChIP-Seq is indeed a powerful technique, and that it can be used to accurately identify TF bindings sites genome-wide, as well as to identify TF binding motifs. While the first ChIP-Seq experiments using the Illumina platform provided an impressive amount of data, great improvements to the technology have been

implemented over the last 4 years. Sequencing a single sample originally provided between three million and five million sequence tags that were 25 nucleotides long. Today, a single sample on the Genome Analyzer provides between ten million and twenty million sequence tags that are between 35 and 150 nucleotides. Additionally, DNA fragments can be sequenced from both ends rather than just one.

In this chapter, I discuss aspects of ChIP-Seq library preparation and data analysis for the Illumina Genome Analyzer platform. Data interpretation and analysis methods are highly dependent on data quality. Therefore, while bioinformatics approaches necessary for processing raw data from the Genome Analyzer and computational methods for peak-calling are the focus of the chapter, it is also important to discuss ChIP-Seq library preparation protocols and experimental design, as these will determine the quality of data that is generated.

8.2 ChIP-Seq Library Preparation

In this section, I first describe the ChIP protocol. While this method has been used for over 20 years, there are certain steps that should be optimized when material will be used for next generation sequencing library preparation. I will then describe a detailed protocol for library preparation for the Illumina Genome Analyzer.

8.2.1 Chromatin Immunoprecipitation

Details of the ChIP protocol have been published previously, and protocols may vary depending on the number of cells being used, the type of protein being assayed, or the number of days required to complete the protocol (Collas 2010). Because ChIP has been previously described and reviewed extensively (Kuo and Allis 1999; Orlando 2000; Chaya and Zaret 2004; Ren and Dynlacht 2004; Collas 2010), here I provide an overview of each of the steps (Fig. 8.1), with emphasis on those steps that may differ when a ChIP-Seq library will be prepared. The first step in performing ChIP is to prepare chromatin. The DNA–protein complexes are cross-linked with formaldehyde and when using animal tissue, the tissue should be minced prior to cross-linking. Following the cross-linking step, it is important to add glycine in order to quench the reaction and prevent over-crosslinking, which can lead to difficulties with DNA shearing. Cells are then lysed, and nuclei are released. The next step, shearing the DNA, is important for library preparation. For ChIP-qPCR or ChIP-on-chip, DNA fragment sizes are considered appropriately sheared if they are between 500 and 1,000 base pairs. However for ChIP-Seq performed on the Genome Analyzer, it is important that fragment sizes are smaller, and on average should be between 100 and 200 base pairs. This is to ensure that DNA fragments are abundant for the size selection step, described in the next part of this section. Following sonication, a small amount of chromatin is uncrosslinked and purified. This DNA,

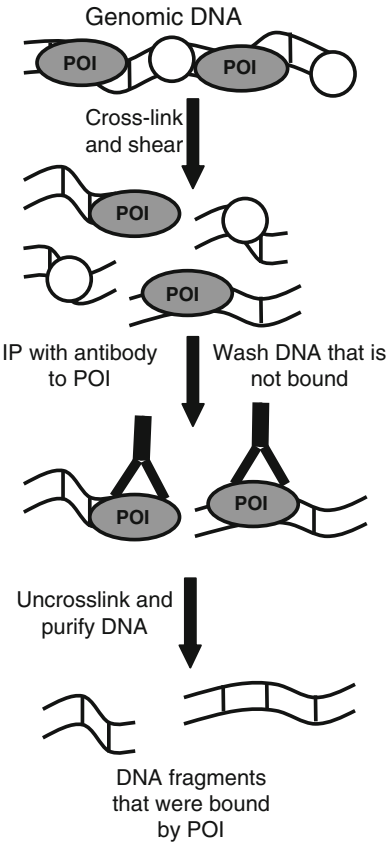

Fig. 8.1 Outline of Chromatin Immunoprecipitation (ChIP). ChIP captures the in vivo binding of a protein of interest (POI) to DNA. DNA is cross-linked, sheared, and incubated with an antibody that recognizes the POI. Unbound DNA is washed away, and bound DNA is uncrosslinked, purified, and analyzed with qPCR

termed input DNA, is quantified and used to determine if shearing was sufficient. Because DNA concentrations can be quite low and accurate visualization of sheared DNA on an agarose gel can be difficult, the Agilent Bioanalyzer can be used to determine the size distribution of sheared DNA. The Bioanalyzer only requires 1 μl of input DNA, and shows the size distribution of DNA fragments in an easy to interpret trace. Examples of optimally sheared DNA and poorly sheared DNA are shown in Fig. 8.2. If input DNA shows larger fragment sizes than desired, chromatin can be sonicated again, and input DNA can be re-purified and re-evaluated. Once appropriately fragmented chromatin is obtained, it can be immunoprecipitated with an antibody of interest. Prior to library preparation, the immunoprecipitated DNA should be tested for enrichment of target regions using qPCR. Enrichment can be calculated compared to input DNA, using the ΔΔCt method. While a fold change of 2 is technically considered enriched, a region that has such a low fold-change by qPCR will likely not show enrichment when ChIP-Seq data is analyzed, because it will not have enough signal compared to background sequence tags. If qPCR enrichments are less than tenfold, experiments should be optimized further to ensure that the signal is strong enough to be recovered in a ChIP-Seq experiment.

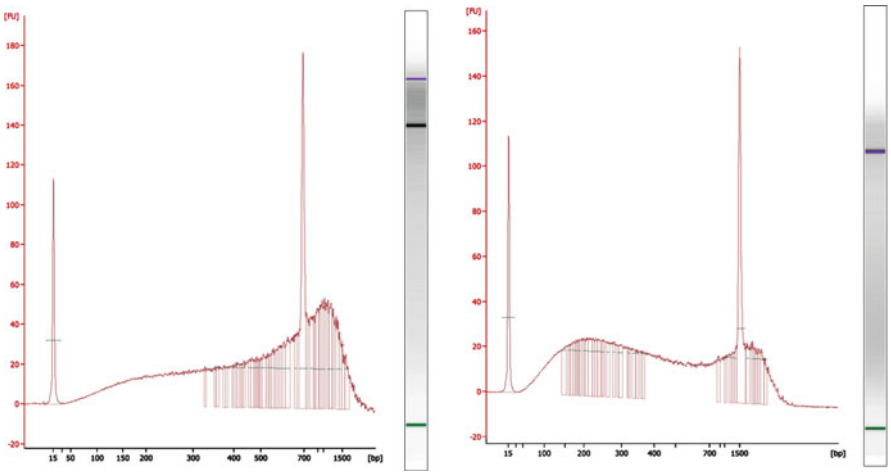

Fig. 8.2 Bioanalyzer traces of sheared chromatin. Chromatin that will be used for ChIP-Seq library preparation should be between 100 and 200 bp. *Left panel* is sheared sufficiently for ChIP-qPCR, but not ChIP-Seq. *Right panel* is sheared sufficiently for both ChIP-qPCR and ChIP-Seq

8.2.2 Library Preparation

Illumina provides a detailed protocol for library preparation that should be followed carefully (http://www.illumina.com). Here I summarize the protocol (Fig. 8.3) and emphasize steps that are important for optimal library preparation. When preparing a ChIP library, it is important to prepare an input library along side to ensure that the steps of library preparation were performed properly. Illumina recommends 10 ng of starting material for library preparation. For most ChIP assays that are performed for a TF, a single ChIP produces far less than 10 ng of DNA, and often there is not enough material to be accurately quantified. However, ChIP-Seq library preparation has been performed using material pulled down from single ChIP experiments (Le Lay et al. 2009; Tuteja et al. 2009), which can be estimated to contain 1–2 ng of DNA.

The first step in library preparation is to blunt the ends of fragmented DNA. The enzymes used in this reaction are T4 DNA polymerase, Klenow polymerase, and T4 polynucleotide kinase. These enzymes remove the 3′ overhangs of the DNA fragments and fill in the 5′ overhangs. The next step is to add an "A" base to the 3′ end of the blunt DNA fragments, using the Klenow exo (3′ to 5′ minus) enzyme. The "A" base allows the DNA fragments to be ligated to the Illumina adapters, which have a single 3′ "T" overhang, in the next step. The Illumina adapters prepare the DNA fragments for hybridization on a flow cell for the sequencing reaction. Once the adapters are ligated to the DNA fragments, the DNA must be size selected. DNA should be run on a 2% agarose gel, and it is recommended that size selection be performed using a dark reader transilluminator to avoid UV exposure, which can

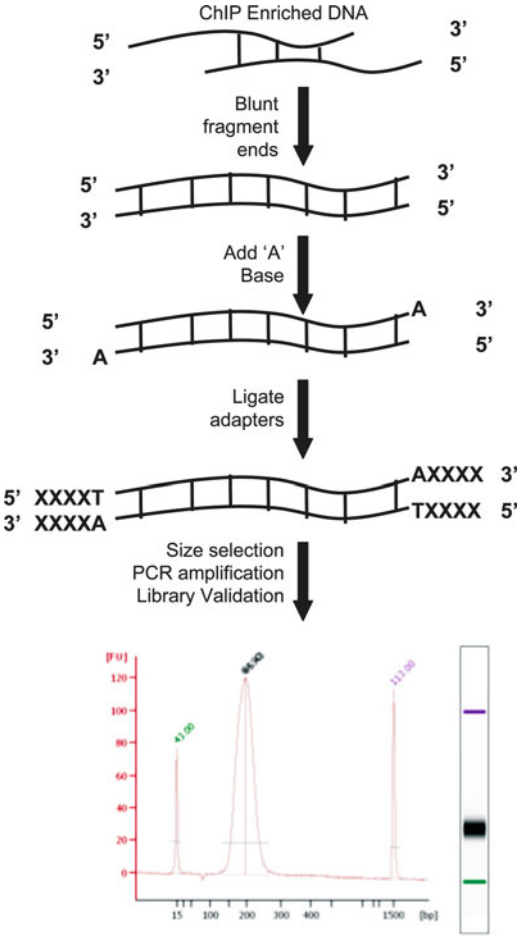

Fig. 8.3 ChIP-Seq Library Preparation. ChIP DNA fragment ends are repaired, followed by addition of an "A" base to the 3′ end. Platform-specific adapters are ligated, and DNA is size selected at 200 bp. Following PCR amplification, library size is validated using the Bioanalyzer

damage DNA. The slice that is excised should be a tight range of DNA, at 200 ± 25 base pairs, and should then be gel purified. This size-selected DNA is then amplified, using Phusion polymerase and PCR primers available through Illumina. Following amplification and purification, the quality of the ChIP-Seq library should be validated using the Bioanalyzer. The bioanalyzer trace should only have a peak around 200 ± 25 base pairs (Fig. 8.3). Any other peaks present in the bioanalyzer trace indicate improper library preparation, which could affect sequencing results. Bioanalyzer traces sometimes show a strong, sharp peak caused by adapter-dimers. If a sample with adapter-dimers is sequenced, the majority of the sequence reads will be adapters, and the number of reads that align to the genome will be significantly decreased. The amount of DNA produced after amplification is generally sufficient for multiple ChIP-Seq runs, and must be diluted prior to submitting samples for sequencing.

8.3 Experimental Design

In order to produce ChIP-Seq data that is accurately representing the biological system being investigated, it is important to consider certain elements of experimental design. Here I describe three important aspects of setting up a ChIP-Seq experiment: using appropriate controls, biological replicates, and using bar-coding to combine multiple experimental conditions or samples into one lane of sequencing.

8.3.1 Controls for ChIP-Seq Experiments

When performing a ChIP-Seq experiment, it is important to also sequence control DNA. Control DNA can either be sheared genomic DNA (input DNA), or a ChIP that was performed with a non-specific antibody, such as IgG. There are regions of the genome that are more likely to be sequenced, due to PCR amplification bias, sonication of DNA, or incorrect mapping of sequences derived from repetitive regions. These regions will show a pile-up of sequence reads in the control data as well as in the ChIP data, and it is therefore important to include control data to eliminate these false-positive peaks.

There have been reports that show that chromatin structure can affect signal distribution of sequence tags (Auerbach et al. 2009). It was demonstrated that chromatin fragmentation more often occurs around open chromatin of expressed genes, and if an input library is sequenced, there is a positive correlation of sequence tags with expressed genes (Auerbach et al. 2009). These results raise important issues in using input DNA as a control. While input DNA from one tissue may have more sequence reads near genes expressed in that tissue, a true ChIP-Seq peak will generally still have more sequence reads around a target region it is binding when compared to input DNA. Input DNA is used to remove non-specific regions that were sequenced, and these regions have very high signal. Additionally, if different chromatin tissue samples were processed under similar sonication conditions, the highly enriched false-positive peaks will be the same between all samples and conditions (Tuteja et al. 2009). While smaller peaks identified in input DNA may differ between tissues, the distribution of peak heights between inputs generated from different tissues is similar, and the distribution can ultimately be used to determine if a peak identified in ChIP-Seq data is higher than what is expected based on input data. The similar distribution of input sample peaks was demonstrated by showing that if GLITR, a ChIP-Seq peak-calling algorithm described later in this chapter, is run on two input samples, treating one as the ChIP DNA and one as the control DNA, no significant peaks are identified (Tuteja et al. 2009).

Sequencing lanes of control DNA can be costly, if it is done for every condition and replicate being assayed. To address this issue, and because a large number of sequence tags are necessary to estimate a background distribution, input sequence tag alignment data from the Genome Analyzer has been made available for MM8,

MM9, HG18, and HG19 (http://web.me.com/kaestnerlab1/GLITR/). These sequence tags were pooled from different tissues, and will remove false-positive peaks that arise in all sequencing reactions without diluting signal for specific tissues and conditions. Because some sequencing biases are specific to sonication conditions, or the environment the library was prepared in (sequencing adapters are sensitive and will stick to lingering DNA on the bench), it is recommended that at least one input sample is sequenced per individual preparing ChIP-Seq libraries, to filter additional false peaks. Additionally, it has been shown that certain cell lines have specific deletions or duplications in their genomes, and if these cell lines are being assayed, the genomic instabilities must be accounted for using experiment-specific control DNA (Blahnik et al. 2010).

8.3.2 Biological Replicates

To identify target regions in ChIP-Seq data, it is important to generate a sufficient number of sequence tags. When the Genome Analyzer was first released, this meant sequencing 3–4 lanes per experimental condition. Because one ChIP-Seq library preparation produces enough material for several lanes of sequencing, one way to produce sufficient sequence tags is to sequence the sample in multiple flow-cell lanes (technical replicates). Another option is to perform library preparation on multiple immunoprecipitation experiments that were done on different chromatin samples (biological replicates). While technical replicates are a faster option, it was shown that biological replicates are a more appropriate option to identifying true target regions (Tuteja et al. 2009). If a sample is re-sequenced in multiple flow cell lanes, it is likely that the majority of the fragments that are sequenced will be the same between lanes. Since all ChIP experiments contain background DNA that was pulled down non-specifically by the antibody of interest, in a technical replicate experiment these non-specific fragments pile-up and become false-positive peaks. Biological replicate sequencing would eliminate ChIP-specific background DNA while amplifying true peaks. Although sequencing technology has advanced since the first release of the Genome Analyzer, and it is now possible to obtain the same number of sequence tags through one lane of sequencing as four lanes in the original technology, biological replicates should still be performed and will provide the same benefits as they did previously.

8.3.3 Bar Coding

Because one lane of sequencing often produces an excess number of sequence tags for yeast and other species with small genomes, a bar-coding technique has been introduced to combine more than one ChIP-Seq library in one lane of sequencing

(Lefrancois et al. 2009). Combining multiple ChIP-Seq libraries into one lane is a cost-effective and efficient way to obtain sequencing data.

In a bar-coding experiment, specific nucleotides, which are different for each ChIP-Seq library preparation that will be sequenced, are added between the sequencing adapter and the DNA fragment. After the sequencing reaction, the first bases that are sequenced correspond to the nucleotides that were added, thus allowing the ability to distinguish one ChIP-Seq library preparation from another one. In experiments performed in yeast, four bar-codes that were four nucleotides each were used. Oligonucleotides were generated that were the same sequence as Illumina adapters, followed by the 4-nucleotide tag (Lefrancois et al. 2009). It is important that all bar-codes used in an experiment end in a "T" base, to allow ligation with the "A" overhang that is added to fragmented DNA during the library preparation. Additionally, the four bar-codes were generated such that even if there was a one or two base pair-sequencing error within the bar-coded region, the sequence read would still be assigned properly (Lefrancois et al. 2009).

Because sequencing technology is constantly advancing, and the number of sequence tags produced from a single experiment is always increasing, the bar-coding technique is now applicable for larger genomes. In addition, the improvements in technology could allow generation of longer bar-codes, and allow more experiments to be combined into one lane of sequencing.

8.4 Raw Data Processing

Following a sequencing reaction, hundreds of gigabytes worth of imaging data must be processed. Most images produced by the Genome Analyzer are processed using the Illumina pipeline, which first uses Firecrest to extract intensities for each cluster. Following image analysis, base calling is performed using Bustard, assigning a sequence to each cluster. Finally, these sequences are aligned to the genome using Eland. There are multiple programs available for these processing steps, and they are discussed in other chapters of this book. Here I discuss processing of alignment files produced by the Illumina pipeline, however, most alignment programs will provide equivalent information in different formats.

8.4.1 Alignment Data

The Illumina pipeline produces multiple output files containing cluster intensities, base quality values, and sequence information. For each lane, a file with the extension "export" is produced, which contains the data necessary to produce input files for peak-calling tools. Information contained in these files include the tile the cluster

was located on, the X and Y coordinates of the cluster on the image, the sequence of the cluster, the quality score of each of the bases in the sequence, the chromosome, start coordinate, and strand of the sequence on the genome, the location and base of a mismatch in the sequence read, a read alignment score, if the sequence read has passed quality control filtering, and an alignment code. The alignment code is either "NM", which indicates the alignment had no match to the genome, "QC", which indicates there was a quality control issue with the alignment, or is of the form "X:Y:Z". In this format, where three numbers are separated by colons, the first number (X) corresponds to the number of times the sequence read aligns to the genome with zero mismatches. The second number (Y) corresponds to the number of times the sequence read aligns the genome with one mismatch, and the third number (Z) corresponds to the number of times the sequence read aligns the genome with two mismatches. The reads that align uniquely to the genome with zero, one, or two mismatches, are generally the only sequence tags that are used for peak-calling. In other words, the sequence reads with alignment codes that are either 1:Y:Z, 0:1:Z, or 0:0:1 are used for peak-calling. This method removes sequence reads that align to multiple locations in the genome from analysis, but has its disadvantages. First, a sequence read with an alignment code of 1:156:178 would pass the filter, even though it aligns to the genome multiple times when a single base in the read is changed. It is possible that one base in the read was not sequenced properly, and the true read actually has multiple alignments to the genome. On the other hand, a sequence read with an alignment code of 2:0:0 would not pass the filter. It is possible that there are two copies of a gene in the genome, and the sequence read aligns in the promoter of both copies of the gene. This sequence tag, and other sequence tags near it that may also be lost, are valuable because it is important to know that there is a peak near a gene involved in a particular function, but if both genes have similar functions, it is not necessarily important to know if the sequence tags should have been assigned to one or both genes. While these examples demonstrate weaknesses in only choosing sequence tags with alignment codes of the form 1:Y:Z, 0:1:Z, or 0:0:1, the cases discussed likely only represent a small fraction of sequence tags and thus using only reads that align uniquely to the genome is sufficient.

8.4.2 Data Visualization

The most common way to visualize ChIP-Seq data is using the UCSC Genome Browser (http://genome.ucsc.edu/). ChIP-Seq data can be uploaded to the Genome Browser using multiple data formats to visualize either raw sequence tags or peaks identified by a ChIP-Seq peak-calling algorithm. Here, raw data visualization methods are discussed, which can be easily adapted to visualizing specific peaks that were determined to be significant.

The Browser Extensible Data (BED) format can be generated from sequence tag data available in the "export" file, described in the previous section. Lines in a BED file must have at least three fields: the chromosome of the sequence read, the start coordinate of the sequence read, and the end coordinate of the sequence read. For reads that were sequenced on the forward or reverse strand, the chromosome and start coordinate are provided in the "export" file, and the end coordinate is simply the start coordinate plus the number of bases that were sequenced. Other optional fields in the BED file include the name, which would be displayed to the left of the sequence read when the display mode in the browser is set to "pack"; the score, which can be displayed in different shades of gray depending on the strength of the score (this option is useful if uploading ChIP-Seq peaks with scores); the strand of the sequence read, written in the file as "+" or "−"; and the itemRgb field. The itemRgb field is an RGB value, written in the file as "R, G, B", which can be used to color sequence reads from individual experiments differently, or to distinguish forward reads from reverse reads. Because sequence reads are only a small portion of the DNA fragment that was sequenced, it is often preferable to visualize full DNA fragments rather than sequence tags. BED files can be used in this case as well, and the only additional data necessary is an estimate of the fragment size of DNA that was excised from the gel during ChIP-Seq library preparation. If X corresponds to the number of bases that were sequenced, and Y corresponds to the fragment size of DNA that was excised from the gel (minus the length of adapters that were ligated), then if the sequence read is on the forward strand, the end coordinate is simply the start coordinate plus Y. If the sequence read is on the reverse strand, then the end coordinate is the coordinate reported in the "export" file plus X, and the start coordinate is then the end coordinate minus Y.

Another format accepted by the UCSC Genome Browser that can be used to visualize ChIP-Seq data is the wiggle (Wig) format. This format can be used to plot the number of overlapping sequence tags (or stack height) at every point in the genome, and allows for continuous peak visualization. To handle the large amounts of data produced by ChIP-Seq analysis, bigBed and bigWig formats have also been introduced. These files are indexed in a binary format, which makes them load faster than their corresponding BED and Wig files. The UCSC Genome Browser offers utilities to convert BED files to bigBed and Wig files to bigWig files. An example of the visualization formats described in this section is shown in Fig. 8.4.

If a custom track is uploaded to the Genome Browser, it can only be viewed on the machine that was used to upload it, and it is automatically deleted 48 h after the last time it was accessed. Users also have the option of creating a "session," which will allow them to save different configurations of their browser with specific track combinations. Any of these configurations can be shared, and each is saved for four months after the last time it was accessed. In addition to the UCSC Genome Browser, other browsers are also available for ChIP-Seq data visualization, such as the Genome Environment Browser (GEB), and the Integrated Genome Browser (IGB) (Huntley et al. 2008; Ji et al. 2008; Nicol et al. 2009).

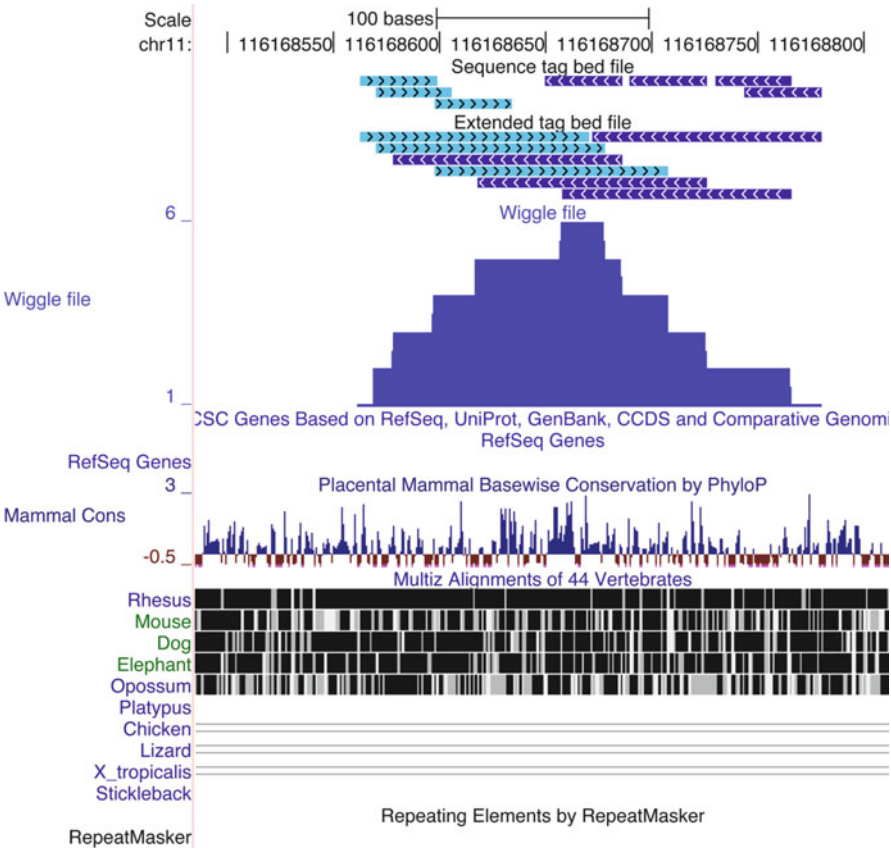

Fig. 8.4 Data visualization formats. Sequenced DNA reads can be visualized using the UCSC Genome Browser. Examples are shown using BED file format for raw sequence reads, sequence reads extended by the DNA fragment length, and using the Wiggle format

8.5 Peak-Calling

A single ChIP-Seq experiment produces millions of reads that align uniquely to the genome. While known targets can be visualized on the UCSC Genome Browser, identifying all of the enriched regions in the data is a challenging task. Many peak-calling tools have been developed over the last few years to address these challenges, and are summarized in Table 8.1 (Boyle et al. 2008; Fejes et al. 2008; Ji et al. 2008; Jothi et al. 2008; Kharchenko et al. 2008; Mortazavi et al. 2008; Nix et al. 2008; Valouev et al. 2008; Zhang et al. 2008, 2011; Lun et al. 2009; Rozowsky et al. 2009; Spyrou et al. 2009; Tuteja et al. 2009; Zang et al. 2009; Blahnik et al. 2010; Qin et al. 2010; Wu et al. 2010; Xu et al. 2010). In this section, different approaches taken by peak-calling programs are generalized, and then one method is discussed in detail.

Table 8.1 ChIP-Seq peak-calling software tools. A subset of peak-calling tools and the general approaches they take are listed

Program	Graphical user interface?	Control sequence tag consider-ation?	Window-based scan or binning	Tag clustering?	Sequence strand consideration (other than tags or peak shifting)	Peak height or fold enrichment reported?	FDR reported?	Availability	General approach	References
BayesPeak	No	Yes	Yes	No	No	No	No	http://WWW.compbio.group.cam.ac.uk/Resources/BayesPeak/csbayespeak.html	Uses a Bayesian hidden markov model.	Spyrou et al. (2009)
CCAT	No	Yes	Yes	No	No	No	Yes	http://cmb.G.edu.sg/CCAT.Htm	Estimates noise rate using an iterative algorithm. Multiple methods to score regions are supported.	Xu et al. (2010)
ChiP-Pam	No	No	Yes	No	Yes	No	No		Models background counts to the Gamma-Poisson model, used to evaluate peak tag counts, and uses pattern matching of forward and reverse strand tag count distributions.	Wu et al. (2010)
CisGenome	Yes	Yes	Yes	Yes	Yes	Yes	Yes	http://biogibbs.stanford.edu/~jihk./CisGenome/Index_files/download.Htm	Conditional binomial model is used to identify regions with significantly enriched ChIP reads relative to control reads.	Ji et al. (2008)

(continued)

Table 8.1 (continued)

Program	Graphical user interface?	Control sequence tag consider-ation?	Window-based scan or binning	Tag clustering?	Sequence strand consideration (other than tags or peak shifting)	Peak height or fold enrichment reported?	FDR reported?	Availability	General approach	References
CSDeconv	No	Yes	Yes	No	Yes	Yes	Yes	http://crab.Ritgers/edi/	Uses Gaussian kernel density estimation and an iterative blind deconvolution approach.	Lun et al. (2009)
ERANGE	No	Yes	No	Yes	No	Yes	Yes	http://Woldlab.Catlech.edu/rnaseq/	Identifies sequence tag location clusters and calculates fold change within the cluster relative to control reads.	Mortazavi et al. (2008)
F-Seq	No	No	Yes	No	Yes	No	No	http://www.genome.duke.edu/labs/Urey/software/fseq/	Generates a continuous tag sequence density using a Gaussian Kernel density estimator.	Boyle et al. (2008)
FindPeaks	No	No	No	Yes	Yes	Yes	Yes	http://www.bcgsc.ca/platform/bioino/software/findpeaks	Finds groups of overlapping tags and uses Monte Carlo simulation for FDR calculation.	Fejes et al. (2008)
GLITR	No	Yes	No	Yes	No	Yes	Yes	http://web.me.com/kaestnerlab1/GLITR	Overlapping extended sequence tags are grouped in ChIP and PseudoChIP data and fold change is calculated to multiply sampled control tags.	Tuteja et al. (2009)

Hpeak	No	Yes	Yes	No	No	Yes	No	http://www/sph	Hidden markov model approach to indentify enriched regions.	Qin et al. (2010)
MACS	No	Yes	Yes	No	No	Yes	Yes	http://liulab.Harvard.edu/MACS/	Uses Poisson distribution to indentify local biases in the genome.	Zhang et al. (2008)
PeakSeq	No	Yes	No	Yes	No	Yes	Yes	http://archive.gersteinlab.org/proj/PeakSeq/	Indentifies regions enriched compared to control data by first taking genome mappability into account.	Rozowsky et al. (2009)
PICS	No	Yes	Yes	No	Yes	Yes	Yes	http://www.bioconductor.org/peck-ages./2.6/bioc/html/PICS.html	Uses a Bayesian hierarchical truncated t-mixture model.	Zhang et al. (2011)
QuEST	No	Yes	No	Yes	No	Yes	Yes	http://mendel.stanford.edu/SidowLab/downloads/quest/	Uses kernel density estimation and splits control data into PseudoChIP and background to calculate FDR.	Valouev et al. (2008)
SICER	No	Yes	Yes	No	No	Yes	Yes	http://home.gwu.edu/~wpeng/Software.htm	Indentifies clusters of enriched windows (islands) and then incorporates control data to determine if the islands are significant.	Zang et al. (2009)

(continued)

Table 8.1 (continued)

Program	Graphical user interface?	Control sequence tag consider-ation?	Window-based scan or binning	Tag clustering?	Sequence strand consideration (other than tags or peak shifting)	Peak height or fold enrichment reported?	FDR reported?	Availability	General approach	References
SiSSRs	No	Yes	Yes	No	Yes	Yes	Yes	http://sissrs.rajajothi.com/	Indentifies candidate binding sites based on transition of sequence read strands and then determines true sites based on criteria including FDR and number of sequence reads in the window.	Jothi et al. (2008)
Sole-Search	Yes	Yes	Yes	No	No	Yes	Yes	http://chipseq.genomecenter.ucdavis.deu/cgi-bin/chipseq.cgi	First uses a background model based on sequenceable genomic regions and then each ChIP region indentified is compared to input using a one sample t-test.	Blahnik et al. (2010)
spp (wtd)	No	Yes	Yes	No	Yes	Yes	Yes	http://compbio.med.harvard.edu/Wupplements/ChIP-seq/	Scores are generated based on strand-specific sequence read numbers up and downstream of the current position (WTD method). Program also implements other peak-calling methods.	Kharchenko et al. (2008)
Useq	No	Yes	Yes	No	No	No	Yes	http://useq.sourceforge.net/	Package of algorithms including methods to calculate binomial *p*-value. Removes regions that have significant global passion *p*- value in control data.	Nix et al. (2008)

8.5.1 Overview of Methods

Multiple ChIP-Seq algorithms have been reviewed previously (Pepke et al. 2009; Wilbanks and Facciotti 2010). An important consideration when choosing a ChIP-Seq program is to ensure that the program can handle background sequence tags. As discussed in Sect. 8.3.1, input DNA sequence tags are not completely random. Therefore, an algorithm that estimates background by randomly assigning sequence tag coordinates to the genome will not be sufficient, even if the tags are modeled with a Poisson or negative binomial distribution. Control sequence tag data has been used by peak-calling programs in different ways. Sole-Search, for example, uniquely utilizes control data to identify and account for amplified and deleted regions present in ChIP-Seq data (Blahnik et al. 2010). Control data is also used directly when determining peak significance, discussed later in this section.

Following identification of tags that align uniquely to the genome, clusters of sequence reads must be identified. A first step that is often used to identify these clusters of reads in the genome is to extend sequence tags to the expected fragment length. While it is possible to estimate fragment length based on the distance between forward and reverse strands around the same peak, it is also known based on the narrow size of the fragment that was excised from the gel during library preparation. Following fragment extension, sequence tags that overlap with each other can be grouped together and considered part of the same peak. Many tools use this approach and define the maximum peak height as the maximum number of overlapping tags within the peak, and this information can be used to calculate significance of the peak (Robertson et al. 2007; Fejes et al. 2008; Rozowsky et al. 2009; Tuteja et al. 2009). Another common approach to grouping sequence reads is the sliding window method, in which a fixed window size is scanned across the genome, counting the number of sequence tags that fall within each window. MACS (*M*odel-based *A*nalysis of *C*hIP-*S*eq) utilizes a sliding window approach, and given the DNA fragment size, scans the genome using a window size that is twice the fragment size. Another strategy taken by peak-calling algorithms is to utilize strand information when identifying potential binding regions. For example, the SISSRS (*S*ite *I*dentification from *S*hort *S*equence *R*eads) algorithm assumes that truly enriched regions should contain roughly the same number of sequence tags from the forward and reverse strands of DNA, and identifies potential binding sites based on transitions between a group of forward tags to a group of reverse tags (Jothi et al. 2008). This method is generally appropriate for eliminating false positives, however, it may be too stringent for identifying weaker binding sites for some TFs, where only the strongest sites would contain sufficient sequence tags to fit the model. CisGenome uses strand information to provide a more precise binding site location, by separately identifying peaks in forward and reverse strands and then pinpointing the binding site to a location between them (Ji et al. 2008). It is also important to note that strand information is most useful when analyzing TF ChIP-Seq data, rather than histone modification, or RNA-polymerase ChIP-Seq data, which have much broader peaks composed of forward and reverse sequence reads throughout.

Often times, binding sites that are closely spaced together are merged into one ChIP-Seq peak. One program that has the ability detect closely spaced binding sites, that are within 100 base pairs or less of each other, is CSDeconv (Lun et al. 2009). CSDeconv uniquely utilizes an iterative blind deconvolution approach to estimate peak shapes and location of binding sites (Lun et al. 2009). Because of the computational intensity of the algorithm and its current implementation, CSDeconv is currently more suitable for analyzing microbial ChIP-Seq data sets rather than mammalian ones.

After peaks are identified in ChIP-Seq data, it is important to assess the significance of these peaks. At this step, control data can be incorporated to determine if the peak identified in ChIP-Seq data is also likely to occur in background control data. One value that can be used to determine statistical significance is the fold-change of the total number of tags in the ChIP-Seq region compared to the total number of tags in the control data in the same region (Johnson et al. 2007). Fold-change relative to control DNA can also be calculated base-by-base in a region, and the average value can be assigned to the region in order to eliminate the possibility of control sequence reads throughout the region that do not overlap with each other, which would falsely dilute the ChIP-Seq signal (Tuteja et al. 2009). Control tags can also be used to choose parameters for statistical models that can then be used to determine peak significance. For example, MACS first calculates a background distribution modeled according to the Poisson distribution using a control data set (Zhang et al. 2008). Candidate peaks that are significantly enriched over the background model are then identified, and local biases are removed (Zhang et al. 2008). MACS then estimates a false-discovery rate by performing a sample swap between ChIP and control data (Zhang et al. 2008). Other models that have been applied to identify significant peaks include the binomial distribution (Nix et al. 2008; Rozowsky et al. 2009), and a hidden markov model (Spyrou et al. 2009; Qin et al. 2010). Generally these models can be used to assign a *p*-value, or another statistical significance value to every peak or cluster of sequence tags that was identified in the data. Other methods, such as GLITR and QuEST, utilize large sets of control tags when identifying significant peaks (Valouev et al. 2008; Tuteja et al. 2009). This allows separation of a "PseudoChIP" sample to use to calculate fold-enrichment to remaining control tags. This allows for an accurate comparison of peak attributes in ChIP data relative to a control, and PseudoChIP data relative to the same control. In the next section, I describe details of the GLITR algorithm.

8.5.2 GLobal Identifier of Target Regions

The *GL*obal *I*dentifier of *T*arget *R*egions (GLITR) software and user manual are available online at http://web.me.com/kaestnerlab1/GLITR/. The input format is simply a file that contains one line for each uniquely aligning sequence read, with the chromosome, start coordinate, and strand, using a "+" to represent the forward strand and a "–" to represent the reverse strand. As described in Sect. 8.4, this information is easily extractable from the "export" file produced by the Illumina pipeline.

GLITR first filters each data set such that each start coordinate is represented once. This is an important step in ChIP-Seq analysis, and it reduces the effect of tags sequenced repeatedly in ChIP data that may not be present in input data. This step does not affect true peaks, as true peaks are covered by sequence tags starting at many different locations.

GLITR requires at least two times the number of ChIP tags in the control data set, but ideally the control set should be larger. As discussed in previous sections, the large control set can be used repeatedly for different experiments, and therefore large sets have been made available on the GLITR website for HG18, HG19, MM8, and MM9. The large set of input tags is necessary because GLITR creates a pseudoChIP set, by randomly sampling the same number of tags from the control set as are contained in the ChIP data set (Fig. 8.5). The remaining control tags (background tags) are used to calculate the fold-change of candidate regions that are identified in both ChIP and pseudoChIP sets, and then to estimate the FDR.

As previously discussed, GLITR extends sequence tags to the expected fragment length and then groups overlapping sequence tags into regions. This is done for the ChIP data as well as the pseudoChIP data, and each region is assigned a maximum peak height. The fold-change of each of these regions is then calculated relative to random samples of background tags, each with the same number of tags as the ChIP data set. GLITR calculates the fold-change to 100 different samples of background tags by default, but this number can be reduced to decrease total computation time. Randomly sampling background data for fold-change calculation prevents losing regions where one particular background sample contains many sequence tags in a region that is truly bound. The large amount of background data also allows for model-free analysis, which is robust through changing technology and identification of biological or experimental factors that affect ChIP-Seq data (Valouev et al. 2008; Tuteja et al. 2009).

To calculate the false discovery rate, GLITR uses both the region peak height, as well as the median fold-change of the region, calculated from all of the samples. This removes artificial peaks that were present in the control data, because they have high peak heights and low fold-changes. It also removes regions with a low peak height and acceptable fold-change, which often occur in background data. Regions are classified as bound using a k-nearest neighbors approach and a false discovery rate is calculated based on results from using the same approach on pseudoChIP regions (Tuteja et al. 2009).

It was demonstrated that GLITR more accurately identified binding sites in ChIP-Seq data for Foxa2 in adult mouse liver, when compared to MACS, SISSRS, PeakSeq, QuEST, and CisGenome (Tuteja et al. 2009). This was probably because of the model-free approach which used a large pool of background tags to estimate the FDR. While most ChIP-Seq programs will easily provide strong peaks that are present in data, weaker peaks are more difficult to identify. Other program comparisons have been carried out (Wilbanks and Facciotti 2010), however, to determine which software tool most often identifies weaker binding sites, it is important to thoroughly compare ChIP-Seq programs on a variety of experimentally generated data sets, including TFs with different ranges in the number of targets they bind to in the genome.

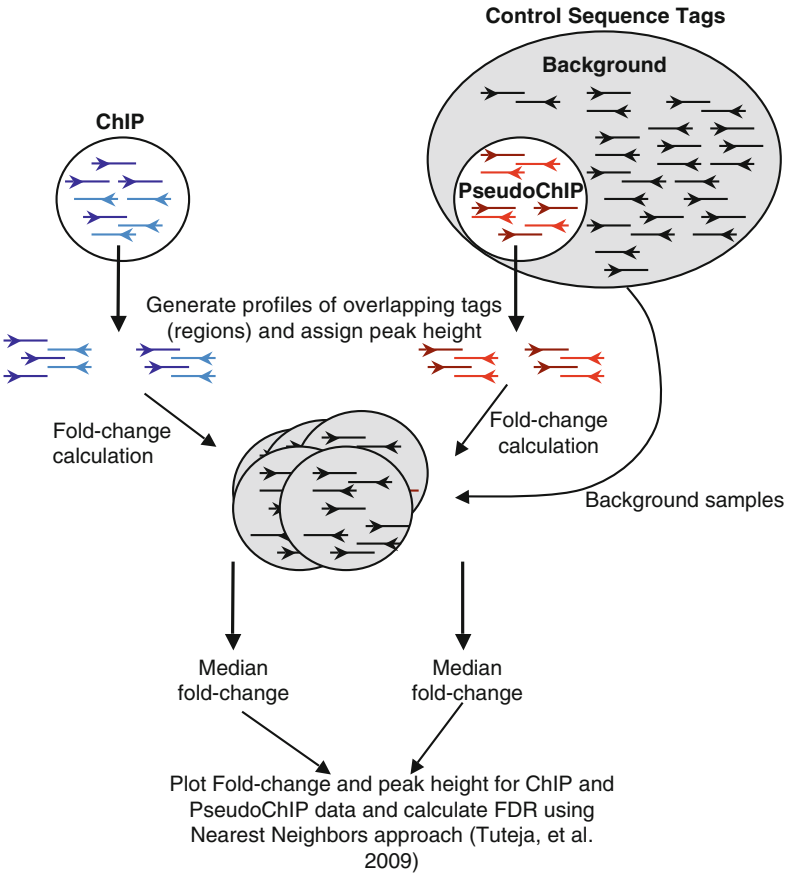

Fig. 8.5 GLITR algorithm outline. GLITR first generates a pseudoChIP sample, which contains the same number of tags as the ChIP-Seq sample, by randomly selecting the tags from a large number of control tags. Control tags are obtained from multiple sequencing runs of sheared input chromatin, and can be utilized for any ChIP-Seq experiment. Overlapping regions of tags are identified in the ChIP and pseudoChIP samples, and are assigned a maximum peak height. A median fold-change is then calculated for each of these regions, based on the fold-change to several random samplings of background tags. Significant peaks are determined by calculating an FDR based on a nearest neighbors approach that utilizes ChIP and pseudoChIP data

8.6 Functional Analysis of ChIP-Seq Peaks

Peak-calling often results in thousands of significant regions that must be further interpreted. A recently developed tool, GREAT (*G*enomic *R*egions *E*nrichment of *A*nnotations *T*ool), can be used to determine the functional significance of the ChIP-Seq regions (McLean et al. 2010). GREAT, available as a web application at http://great.stanford.edu/, is the first tool that appropriately assigns functional enrichments

for regions distal to transcription start sites, which are very commonly found in hIP- sequencing data. GREAT currently includes annotations for over 20 ontologies and supports g18, hg19, mm9, and danRer7. GREAT takes a list of peaks in BED format as input, and can also handle user-defined background sets. This tool can be used to facilitate interpretation of ChIP-Seq data by dividing peaks into categories of enriched biological functions.

8.7 Conclusions

ChIP-Seq was first introduced only 4 years ago, but great improvements have been made in both the sequencing technology and data analysis approaches. ChIP-Seq has revolutionized the study of transcriptional regulation by allowing rapid identification of all of the binding sites in the genome targeted by a TF. When planning a ChIP-Seq experiment, it is important to carefully think about experimental design, and the best way to perform experiments, in order to achieve results that best represent the system being assayed. Additionally it is important to contemplate the specific biological questions that will be answered with the ChIP-Seq data, in order to ensure the most appropriate analysis approaches are carried out.

References

Auerbach RK, Euskirchen G, Rozowsky J, et al. (2009) Mapping accessible chromatin regions using Sono-Seq. Proc Natl Acad Sci USA 106:14926–31.

Blahnik KR, Dou L, O'Geen H, et al. (2010) Sole-Search: an integrated analysis program for peak detection and functional annotation using ChIP-seq data. Nucleic Acids Res 38:e13.

Boyle AP, Guinney J, Crawford GE, et al. (2008) F-Seq: a feature density estimator for high-throughput sequence tags. Bioinformatics 24:2537–8.

Buck MJ and Lieb JD (2004) ChIP-chip: considerations for the design, analysis, and application of genome-wide chromatin immunoprecipitation experiments. Genomics 83:349–60.

Chaya D and Zaret KS (2004) Sequential chromatin immunoprecipitation from animal tissues. Methods Enzymol 376:361–72.

Collas P (2010) The current state of chromatin immunoprecipitation. Mol Biotechnol 45:87–100.

Fejes AP, Robertson G, Bilenky M, et al. (2008) FindPeaks 3.1: a tool for identifying areas of enrichment from massively parallel short-read sequencing technology. Bioinformatics 24: 1729–30.

Friedman JR, Larris B, Le PP, et al. (2004) Orthogonal analysis of C/EBPbeta targets in vivo during liver proliferation. Proc Natl Acad Sci USA 101:12986–91.

Harbison CT, Gordon DB, Lee TI, et al. (2004) Transcriptional regulatory code of a eukaryotic genome. Nature 431:99–104.

Huntley D, Tang YA, Nesterova TB, et al. (2008) Genome Environment Browser (GEB): a dynamic browser for visualising high-throughput experimental data in the context of genome features. BMC Bioinformatics 9:501.

Ji H, Jiang H, Ma W, et al. (2008) An integrated software system for analyzing ChIP-chip and ChIP-seq data. Nat Biotechnol 26:1293–300.

Johnson DS, Mortazavi A, Myers RM, et al. (2007) Genome-wide mapping of in vivo protein-DNA interactions. Science 316:1497–502.
Jothi R, Cuddapah S, Barski A, et al. (2008) Genome-wide identification of in vivo protein-DNA binding sites from ChIP-Seq data. Nucleic Acids Res 36:5221–31.
Kharchenko PV, Tolstorukov MY and Park PJ (2008) Design and analysis of ChIP-seq experiments for DNA-binding proteins. Nat Biotechnol 26:1351–9.
Kuo MH and Allis CD (1999) In vivo cross-linking and immunoprecipitation for studying dynamic Protein: DNA associations in a chromatin environment. Methods 19:425–33.
Le Lay J, Tuteja G, White P, et al. (2009) CRTC2 (TORC2) contributes to the transcriptional response to fasting in the liver but is not required for the maintenance of glucose homeostasis. Cell Metab 10:55–62.
Le PP, Friedman J, Schug J, et al. (2005) Glucocorticoid receptor-dependent gene regulatory networks. PLoS Genetics 2:159–170.
Lefrancois P, Euskirchen GM, Auerbach RK, et al. (2009) Efficient yeast ChIP-Seq using multiplex short-read DNA sequencing. BMC Genomics 10:37.
Lun DS, Sherrid A, Weiner B, et al. (2009) A blind deconvolution approach to high-resolution mapping of transcription factor binding sites from ChIP-seq data. Genome Biol 10:R142.
McLean CY, Bristor D, Hiller M, et al. (2010) GREAT improves functional interpretation of cis-regulatory regions. Nat Biotechnol 28:495–501.
Mockler TC, Chan S, Sundaresan A, et al. (2005) Applications of DNA tiling arrays for whole-genome analysis. Genomics 85:1–15.
Mortazavi A, Williams BA, McCue K, et al. (2008) Mapping and quantifying mammalian transcriptomes by RNA-Seq. Nat Methods 5:621–8.
Nicol JW, Helt GA, Blanchard SG, Jr., et al. (2009) The Integrated Genome Browser: free software for distribution and exploration of genome-scale datasets. Bioinformatics 25:2730–1.
Nix DA, Courdy SJ and Boucher KM (2008) Empirical methods for controlling false positives and estimating confidence in ChIP-Seq peaks. BMC Bioinformatics 9:523.
Orlando V (2000) Mapping chromosomal proteins in vivo by formaldehyde-crosslinked-chromatin immunoprecipitation. Trends Biochem Sci 25:99–104.
Pepke S, Wold B and Mortazavi A (2009) Computation for ChIP-seq and RNA-seq studies. Nat Methods 6:S22–32.
Qin ZS, Yu J, Shen J, et al. (2010) HPeak: an HMM-based algorithm for defining read-enriched regions in ChIP-Seq data. BMC Bioinformatics 11:369.
Ren B and Dynlacht BD (2004) Use of chromatin immunoprecipitation assays in genome-wide location analysis of mammalian transcription factors. Methods Enzymol 376:304–15.
Ren B, Robert F, Wyrick JJ, et al. (2000) Genome-wide location and function of DNA binding proteins. Science 290:2306–9.
Robertson G, Hirst M, Bainbridge M, et al. (2007) Genome-wide profiles of STAT1 DNA association using chromatin immunoprecipitation and massively parallel sequencing. Nat Methods 4:651–7.
Royce TE, Rozowsky JS, Bertone P, et al. (2005) Issues in the analysis of oligonucleotide tiling microarrays for transcript mapping. Trends Genet 21:466–75.
Rozowsky J, Euskirchen G, Auerbach RK, et al. (2009) PeakSeq enables systematic scoring of ChIP-seq experiments relative to controls. Nat Biotechnol 27:66–75.
Rubins N, Friedman J, Le P, et al. (2005) Transcriptional networks in the liver: hepatocyte nuclear factor 6 function is largely independent of Foxa2. Mol Cell Biol 25:7069–77.
Simon I, Barnett J, Hannett N, et al. (2001) Serial regulation of transcriptional regulators in the yeast cell cycle. Cell 106:697–708.
Solomon MJ, Larsen PL and Varshavsky A (1988) Mapping protein-DNA interactions in vivo with formaldehyde: evidence that histone H4 is retained on a highly transcribed gene. Cell 53: 937–47.
Spyrou C, Stark R, Lynch AG, et al. (2009) BayesPeak: Bayesian analysis of ChIP-seq data. BMC Bioinformatics 10:299.
Tuteja G, White P, Schug J, et al. (2009) Extracting transcription factor targets from ChIP-Seq data. Nucleic Acids Res 37:e113.

Valouev A, Johnson DS, Sundquist A, et al. (2008) Genome-wide analysis of transcription factor binding sites based on ChIP-Seq data. Nat Methods 5:829–34.
Wilbanks EG and Facciotti MT (2010) Evaluation of algorithm performance in ChIP-seq peak detection. PLoS One 5:e11471.
Wu S, Wang J, Zhao W, et al. (2010) ChIP-PaM: an algorithm to identify protein-DNA interaction using ChIP-Seq data. Theor Biol Med Model 7:18.
Wyrick JJ, Aparicio JG, Chen T, et al. (2001) Genome-wide distribution of ORC and MCM proteins in *S. cerevisiae*: high-resolution mapping of replication origins. Science 294:2357–60.
Xu H, Handoko L, Wei X, et al. (2010) A signal-noise model for significance analysis of ChIP-seq with negative control. Bioinformatics 26:1199–204.
Zang C, Schones DE, Zeng C, et al. (2009) A clustering approach for identification of enriched domains from histone modification ChIP-Seq data. Bioinformatics 25:1952–8.
Zhang X, Robertson G, Krzywinski M, et al. (2011) PICS: Probabilistic Inference for ChIP-seq. Biometrics 67(1):151–63.
Zhang Y, Liu T, Meyer CA, et al. (2008) Model-based analysis of ChIP-Seq (MACS). Genome Biol 9:R137.

Chapter 9
Generation and Analysis of Genome-Wide DNA Methylation Maps

Martin Kerick, Axel Fischer, and Michal-Ruth Schweiger

Abstract Cytosine methylations are common mechanisms of epigenetic modifications of DNA molecules which also influence gene expression and cell phenotypes. Thus, 5 methyl-cytosine is sometimes called the fifth base of the genome. The development of high throughput sequencing (HTS) technologies has – for the first time – brought about tools to investigate epigenetic alterations in a genome-wide approach. First methylation maps have already been created and it is only a question of time until complete epigenetic maps of healthy and diseased human tissues are available. Here, we summarize the use of HTS for diverse epigenetic technologies, give an overview of the status quo of methylation maps, touch bioinformatics software applications and problems and, finally, outline future perspectives for the application in oncology and basic research.

9.1 The Genome and the Epigenome Determine the Phenotype of Organisms

Despite identical genotypes the phenotypes between organisms can differ dramatically. For humans this effect is obvious for monozygotic twins, but can also be seen in intra-individual changes of gene expression during disease, in response to environmental stimuli and during the aging process. Underlying mechanisms need to be constant during cellular development, heritable to daughter cells and we need to be able to respond to outer environmental influences. These pre-requirements are met by epigenetic modifications, e.g. histone modifications and, most importantly, also DNA methylation alterations.

M. Kerick • A. Fischer • M.-R. Schweiger (✉)
Cancer Genomics Group, Department of Vertebrate Genomics,
Max Planck Institute for Molecular Genetics, Berlin, Germany
e-mail: mschweig@molgen.mpg.de

N. Rodríguez-Ezpeleta et al. (eds.), *Bioinformatics for High Throughput Sequencing*,
DOI 10.1007/978-1-4614-0782-9_9,

9.2 Altered Epigenetic Patterns are Found in Several Diseases, Especially in Cancer

The nucleotide sequence is the primary level of genetic information and the basic principle of genetic inheritance. Another level of complexity in genomic information arises from epigenetic variations of DNA segments which are also underlying the inheritance of phenotypes from generation to generation as well as from cell to cell during cell division (Laird 2010). Genome-wide studies on epigenetic changes are now termed "epigenomics". Epigenetic variations can be grouped into covalent DNA modifications, in particular methylation of nucleotides, or post-transcriptional modifications of histones (e.g. acetylation, ubiquitination or methylation). In humans, cytosine methylation was the first mark discovered. In the current paradigm it is required for the regulation of gene expression as well as for silencing transposons and other repetitive sequences (Walsh et al. 1998). The chemical modification occurs predominantly via a covalent attachment of a methyl group to the C5 position of the cytosine ring (5mC) in CpG dinucleotides. Thereby the structure of cytosine is altered without changing its base-pairing properties. Altered methylation patterns have been reported in a diverse array of complex human diseases such as cancer, systemic autoimmune and psychiatric diseases as well as in monogenic epigenetic diseases (Feinberg 2007). In this regard, the first molecular epigenetic change, a global reduction of methylation in cancer cells, has been described by Feinberg and Vogelstein (1983) and in the same year by Gama-Sosa et al. (1983). These changes were found in both pre-invasive and invasive cancers and implicate that alterations in the cytosine methylation patterns are among the earliest events in tumorigenesis. In addition, it has been shown that specific alterations in the methylation patterns of CpGs in promoter regions are associated with certain tumor entities or stages. Consequently, the first biomarkers have been developed on the basis of these modifications (Banerjee and Verma 2009).

9.3 Technologies for High-Throughput Epigenetic Analyses

Over the past years several epigenetic technologies have been developed either for profiling methylated genomic regions (indirect methods) or for typing the methylated base (direct methods). These approaches differ concerning the obtainable resolution with direct methods resulting in single-nucleotide patterns of methylated cytosines within genomes, while indirect methods measure average methylation levels across many molecules (Beck and Rakyan 2008; Laird 2010; Lister and Ecker 2009; Pomraning et al. 2009).

Using HTS technologies for the interrogation of methylation patterns, the classification into indirect and direct approaches can be maintained and extended (Tables 9.1 and 9.2): Indirect methods – Methyl-Seq, MCA-Seq, HELP-Seq, MSCC, MeDIP-Seq, MBP-Seq and MIRA-Seq – are based on enrichments of methylated

Table 9.1 Methods used for the detection of methylated cytosines

Measurement	Type	Abbreviation	Method	Publications
Direct	BS-Seq	BS-Seq	Shotgun-bisulfite-sequencing	Carr et al. (2007), Korshunova et al. (2008) and Wang et al. (1980)
		BC-Seq	Bisulfite capture sequencing	Hodges et al. (2009)
		BSPP	Bisulfite specific padlock probes	Ball et al. (2009), Berman et al. (2009), Deng et al. (2009) and Li et al. (2009)
		RRBS	Reduced representation bisulfite sequencing	Meissner et al. (2005, 2008)
Indirect	Enzyme-Seq	Methyl-Sec	Methylation sequencing	Brunner et al. (2009)
		MSCC	Methyl sensitive cut counting	Ball et al. (2009) and Berman et al. (2009)
		HELP-Sec	Hpa II fragment enrichment by ligation PCR	Oda et al. (2009)
		MCA-Seq	Methylated CpG island amplification	Toyota et al. (1999)
	AF-Seq	MeDIP-Seq	Methylation-dependent immunoprecipitation sequencing	Jacinto et al. (2008)
		MBP-Seq	Methyl-binding protein sequencing	Gebhard et al. (2006) and Jorgensen et al. (2006)
		MIRA-Seq	Methylated CpG island recovery assay	Rauch and Pfeifer (2005)

Table 9.2 Technological features of methods for DNA methylation analyses

Abbreviation	Methylation information	Enrichment	Covered regions	Nucleotide resolution	Biases	Key-features
BS-Seq	CpG-, Non-CpG-, cis co- and allele specific	None	Whole-genome	Single	Incomplete bisulfite conversion/PCR	Two-adapter-sets
BC-Seq		Array				Array-capture
BSPP		PCR				Padlock probe enrichment
RRBS						MspI-digest, size selection
Methyl-Seq	CoG-, Non-CpG- and allele specific	Size selection	Pre-selected regions		Fragment size	HpaII-, MspI-digest, in silico size selection
MSCC						HpaII-, MspI-digest Adapters with Mme site; ~20bp tags
HELP-Sec						HpaII-digest; ligation-mediated PCR; Array hybridization
MCA-Seq						SmaI-, XmaI-digest, size selection by PCR
MeDIP-Seq	Partly allele specific	Precipitation	Methylated regions	100 bp	GC content/CpG density/CNVs	Precipitation of methylated DNA with bead coupled antibody
MBP-Seq						Precipitation of methylated DNA with bead coupled MBP
MIRA-Seq						Precipitation of methylated DNA with bead coupled MBD3L1

regions. Methylation profiles are then inferred by subsequent sequencing, read alignment and counting of reads per genomic interval. Direct methods – BS-Seq, BC-Seq, BSPP and RRBS – in contrast rely on bisulfite conversion of unmethylated cytosines and consecutive sequencing, which allows methylation profiling with a resolution on single base level. Both, direct and indirect methods will be the focus of this review followed by a short outlook on future developments and their potential employment for medical applications.

9.3.1 Indirect Epigenetic Analyses

Indirect approaches provide information as a methylation score for regions of approximately 100–200 bp length. All methods are based on the enrichment of methylated DNA. The fragments captured by any of those methods can then be identified by either hybridization to known sequences or by sequencing.

The use of HTS instead of custom-designed hybridization-arrays to identify precipitated DNA fragments provides genome-wide information about methylated regions. This implies that all DNA fragments can be identified and not only preselected regions which are immobilized on an array. The completeness of the data is especially advantageous in generating methylation profiles outside of CpG-islands and promoter regions, for example in gene bodies where DNA methylation changes have recently been shown to occur (Ball et al. 2009; Rakyan et al. 2008).

MeDIP-Seq and MBP-Seq rely on precipitations of DNA fragments containing methylated cytosines (5mC) using an anti-5mC antibody or methyl-binding proteins (MBP) and are thus termed methylation-dependent immunoprecipitation (MeDIP) and MBP assays (Cross et al. 1994; Keshet et al. 2006; Rauch and Pfeifer 2005; Weber et al. 2005). Both methods belong to the class of affinity-enrichment sequencing approaches (AE-Seq).

The MeDIP-enrichment depends upon the 5mC content in a way that a threshold level of methylation, approximately 2–3%, is required for a successful enrichment. Regions with high CpG content are therefore more likely to be enriched than regions with low CpG content. First MeDIP-seq experiments indicate that approximately 30–40 million reads are required for a human genome-wide analysis (Beck and Rakyan 2008; Down et al. 2008). MeDIP-seq approaches have been performed so far using Illumina's Genome Analyzer technology (Down et al. 2008), but we recently established several methylation analysis methods for SOLiD sequencers, because of improved throughput (Boerno et al. 2011). MBPs preferentially bind double stranded DNA with symmetrically methylated CpG sequences and, in contrast to MeDIP-protocols, where the DNA is denatured and single stranded, the adapter ligation step is less critical and can be performed after the affinity purification. A challenge of both AE-Seq methods is that "no signal" can be explained either by very low methylation levels or experimental failure and hypomethylation patterns are therefore very difficult to assess.

Protocols that use endonucleases (Enzyme-Seq technologies) like Methyl-Seq (Brunner et al. 2009), MCA-Seq (Toyota et al. 1999), HELP-Seq (Oda et al. 2009)

and MSCC (Ball et al. 2009; Berman et al. 2009) exploit the fact that restriction enzymes exist which target sequences that comprise CpG sites in a methylation-sensitive manner.

Analysis is done by counting the reads per genomic region and combined evaluation of treatment and control samples. If no control samples exist methylation-sensitive (e.g. HpaII, SmaI) and methylation-insensitive (e.g. MspI, XMaI) preparations can be compared, a step which is also advisable if copy number variants are expected to be present (Oda et al. 2009). In the same line restriction digests can also be compared to randomly sheared fragments as was shown in a study that used 3–10 million sequencing reads per sample and was able to interrogate 65% of all annotated CpG islands (Brunner et al. 2009). The commonly used size selection constraint to 100–200 bases limits the number of CpG sites that can be interrogated. This can be improved by the MSCC approach. The usage of an adapter with MmeI-recognition sites and performance of a MmeI-digest after the ligation step results in genomic DNA tags of approximately 20 bp length which is an ideal length for HTS. With this approach a maximum of 1.4 million CpG sites can be interrogated and with approximately 20 million sequencing reads, 66% of the CpG sites have been covered with at least one read. A drawback of the Enzyme-Seq methods is that any region showing at least one read in the methylation-sensitive digest is currently called "unmethylated". Thereby the quantitative methylation state of the individual region is lost, and partial methylation remains unidentified (Ball et al. 2009; Brunner et al. 2009).

For indirect epigenetic techniques the question of sequence resolution plays an important role since single-nucleotide methylation patterns have not yet been achieved. There is an ongoing discussion whether determining global changes in methylation – as observed by indirect assessment techniques – might be sufficient for epigenetic studies due to correlations between CpG island methylation within short regions (1,000 bp) and coordinated gene suppression across entire chromosome bands (Eckhardt et al. 2006; Frigola et al. 2006).

9.3.2 Direct Epigenetic Analyses

Direct assessment techniques like BS-Seq (Carr et al. 2007; Korshunova et al. 2008; Wang et al. 1980), BC-Seq (Hodges et al. 2009), BSPP (Ball et al. 2009; Berman et al. 2009; Deng et al. 2009; Li et al. 2009), or RRBS (Meissner et al. 2005, 2008) determine methylation profiles directly from the sequence enabling base pair resolution. Methylated DNA is marked through a "bisulfite (BS) conversion" reaction for which genomic DNA is treated with sodium bisulfite under denaturing conditions. Cytosine residues get deaminated and converted to uracil leaving methylated cytosine moieties unaffected (Frommer et al. 1992). Identification of the resulting DNA sequence leads to a detection of converted and unconverted cytosine residues and subsequent identification of the prior methylation status of the nucleotide. The analysis deduces that cytosine residues were methylated if they have not been converted by bisulfite. Common to all direct investigation techniques are pitfalls leading to false positive

methylation calls due to incomplete conversion reactions, degraded DNA caused by harsh conversion conditions and methylation in pseudogenes (Esteller 2002; Warnecke et al. 1997). Using conventional sequencing strategies like Sanger sequencing, genome-wide undertakings would be extremely time and cost-intensive. Only with the aid of HTS technologies do comprehensive m5C patterns become feasible (Eckhardt et al. 2006).

So far, genome-wide single-nucleotide resolution BS studies with HTS technologies have been mainly performed for small genomes like *Arabidopsis thaliana* (Cokus et al. 2008; Lister et al. 2008). In these studies approximately 85% of all cytosines in the 119 Mb *A. thaliana* genome have been addressed. In a pilot study the BS-seq approach has also been used for mouse genomic germ cell DNA where 66% of all sequencing reads could be mapped to the genome, demonstrating that these approaches can be extended to larger genomes such as those in mice or humans (Popp et al. 2010). First human genome-wide epigenetic maps after bisulfite treatment have been constructed and show that more than 93% of all CpGs can be targeted (Li et al. 2010; Lister and Ecker 2009).

Major challenges for whole genome BS sequencing are the sequencing capacities and costs required, which are still relatively high. Thus, it is more practical to investigate only parts of the genome to gain insight into methylation patterns of mammals, especially if large numbers of samples need to be analyzed. First approaches to reduce the genome complexity for bisulfite sequencing have been performed by PCR-amplification of target regions (BC-Seq) (Korshunova et al. 2008; Taylor et al. 2007). Another approach, termed BSPP, combines targeted enrichment of specific DNA regions by padlock probes and rolling circle PCR, BS conversion and HTS. As a proof of principle 10,000 independent regions were queried (Ball et al. 2009). A disadvantage of targeted enrichment BS-Seq is the bias introduced by selecting a subset of "interesting" sites. One protocol, which does not rely on sequence specific pre-selections of DNA regions but does select for regions with high CpG density is reduced representation BS Sequencing (RRBS). It uses the digestion of the genomic DNA at CCGG sites with a methylation-insensitive restriction enzyme followed by size selection, BS conversion and sequencing (Meissner et al. 2008).

9.3.3 Comparison of Epigenetic Analysis Protocols

DNA methylation analysis methods cannot easily be compared as many approaches have competing strengths and weaknesses. The number of samples which can be analyzed in parallel, the quantity of DNA and the desired resolution are the central decision points.

Methods based on endonuclease treatment (Enzyme-Seq) tend to require DNA of high quality and quantity (Oda et al. 2009). Affinity-enrichment techniques (AE-Seq) can tolerate a certain amount of DNA impurity but result in a coarse resolution of DNA methylation (Laird 2010). Bisulfite treatment (BS-Seq) not only requires DNA denaturation before treatment but also can cause substantial DNA degradation. Even more challenging, an overtreatment with bisulfite

can lead to a conversion of methylated cytosines to thymine (Wang et al. 1980). On the other side these protocols give detailed and nucleotide-specific information on methylation patterns at the price of high sequencing costs and low throughput in the case of genome-wide analyses.

The focus of most approaches is towards methylated cytosines within the CpG context. While other types of methylation, such as methylation at CpHpG (H = A, T, C) sites, exist (Kriaucionis and Heintz 2009; Lister et al. 2008), they are less frequently investigated and fewer studies are published (Huang et al. 2010; Jin et al. 2010; Nestor et al. 2010).

Enzyme recognition sites limit the detectable CpG context for Enzyme-Seq methods and it is hard to measure DNA methylation quantitatively. While Enzyme-Seq methods are able to resolve methylation differences in low-CpG-density regions, affinity-based methods perform well for CpG-rich regions (Irizarry et al. 2008). On the other hand genome-scale techniques like BS-Seq are not particularly well suited for the detection of low-frequency methylation states in a large cohort of samples as sensitivity is generally a function of sequencing depth and therefore also of costs.

The read count methods (AE-Seq and Enzyme-Seq) are prone to sources of bias, such as GC content, fragment size and copy-number variations in the source DNA, that affect the likelihood that a particular region is included in the sequenced fragments (Aird et al. 2011; Dohm et al. 2008; Schweiger et al. 2009). The bisulfite-based methods are also subject to these effects, but they do not influence the DNA methylation measurement itself, as this information is extracted from the sequence.

Taken together, the number of different HTS epigenetic technologies is large, and each has its own advantages and disadvantages. The selection of the right technology for the research question investigated is crucial in making the most out of the enormous power HTS has to offer for basic and clinical directions of research.

9.4 Bioinformatic Analyses

Along with the development of high-throughput sequencing technologies the need for adequate data handling emerged. The large amount of data, the statistical analyses with massive multiple-testing approaches, requires advances in both data storage and software. To investigate methylation patterns analysis tools comprise software for the mapping of short reads to DNA, peak calling algorithms in the case of enrichment-based technologies and downstream functional annotation techniques.

9.4.1 Alignment

The key bioinformatics step of methylation analysis is the fast but accurate mapping of short read sequences to the reference genome. For the basic alignment of short reads

several algorithms have been developed and are still in the process of improvement. Most frequently used softwares comprise: Bowtie (Langmead et al. 2009), BWA (Li and Durbin 2009), Eland, Maq (Li et al. 2008) and Novoalign. In the case of bisulfite converted DNA the task is somewhat more complicated because the conversion leads to an overrepresentations of uridines/thymidines and thus to a reduction of the genetic complexity. Here, either the alignment parameters should be optimized to tolerate more mismatches than usual (bowtie: –v, bwa: –M) or special software for BS-data should be used [Bismark, BSMAP (Xi and Li 2009), BS seeker (Chen et al. 2010), Novoalign, GSNAP (Wu and Nacu 2010)].

9.4.2 Interpretation of Data from Bisulfite Treated DNA

After bisulfite conversion, the majority of DNA being sequenced is effectively composed of just three bases and one encounters a high error rate when base-calling is performed. For calibration purposes it is therefore necessary to sequence a control library which contains all four bases and it is advisable to optimize the base callers (Cokus et al. 2008; Lister et al. 2008).

Analysis of sequences from bisulfite treated DNA is based on single-nucleotide variant detection methods that identify variants which result from bisulfite conversion. Cytosine residues which have not been converted are assumed to be methylated. However, CpG dinucleotides are common sites of polymorphisms and one has to distinguish polymorphisms from bisulfite-induced deaminations. One error prone method is to exclude known polymorphisms. However, this approach might miss potential methylations within known polymorphisms. Another method is to sequence the non-bisulfite-converted genome of interest and to use it for comparison.

A more efficient method takes advantage of the information gathered by high-throughput sequencing. A SNV caused by long-term (e.g. evolutionary) deamination of C to T will have been propagated on the opposing DNA strand as an A, whereas bisulfite deamination (spontaneous deamination) of an unmethylated cytosine will leave the G on the opposing strand unaffected (Weisenberger et al. 2005). Sequencing of both DNA strands of bisulfite-converted DNA can therefore discriminate between a CpG SNV and an unmethylated CpG without the need to sequence the non-bisulfite converted genome.

9.4.3 Peak Detection

Enrichment-based approaches (AE-Seq and Enzyme-Seq) to determine methylation profiles only exploit the position of the mapped reads as opposed to bisulfite sequencing. In detail, the data can be summarized as counts in a genomic interval (bin). Peak finding algorithms are then needed to identify regions with significantly increased coverage of reads. Numerous algorithms have already been developed and evaluated (Pepke et al. 2009; Wilbanks and Facciotti 2010) for chromatin

immunoprecipitation datasets (ChIP-Seq), but they have to be used with caution for methylation analysis as there is only limited knowledge of defined peak shapes (Robinson et al. 2010a). However, two main studies have adapted the ChiP-Seq analysis pipeline especially to DNA methylation analyses. Down et al. combined MeDIP with high-throughput sequencing for whole-genome methylation studies and developed the "Batman" algorithm for analyzing MeDIP profiles (Down et al. 2008). With similar performance, but with a more time-efficient computational method Chavez et al. established the MEDIPs package which in addition contains quality control metrics and identifies differential methylation (Chavez et al. 2010). Both algorithms have been developed to take local CpG densities into consideration and calculate differential methylation levels.

9.4.4 Sources of Bias

While many studies proclaim that "their" technique used is unbiased and covers the whole methylome, first reports (Hodges et al. 2009; Robinson et al. 2010b) and our own data suggest that methylation profiling is not straightforward. In fact, the detection of biases, its accounting and normalization form a significant part of the bioinformatic analysis. Next to the limits of the protocol used that determines how many CpG sites can actually be observed, four sources of bias have been described: sequencing bias, mapping bias, CpG density bias and copy number bias (Robinson et al. 2010a).

High throughput sequencing in itself is positively correlated with the GC content of a region (Down et al. 2008; Schweiger et al. 2009; Timmermann et al. 2010). In addition, enrichment methods also tend to be affected by the local CpG distribution. For analyses like SNV detection this bias only has an indirect effect as the coverage levels will vary and some positions might be missed entirely. However, for analysis strategies that generate information based on coverage levels it is rather important to correct for GC content as is implemented for instance in the Batman and MEDIPs packages (Chavez et al. 2010; Down et al. 2008) or background models need to be calculated (Zhang et al. 2008).

Mapability poses the second challenge to methylation analyses. Depending on the read length a certain proportion of the genome cannot be covered by unambiguously placed reads due to repetitive sequences; this affects approximately 13% of the genome with 36 bp reads and 8% with 50 bp reads. Although some reads will extend into, and thereby recover repetitive sequences, the region will have a lower coverage than regions with unique sequence. Longer reads and paired-end (mate-pair) reads improve the mapability problem significantly.

For bisulfite-treated DNA the situation is even more complicated. The strategy how the reference genome is degenerated to enable the sequence alignment step – if "C" is replaced by "Y" or "T" – has an impact on mapability. Even more dangerous, the substitution of "C" by "Y" in the reference sequence renders the mapability

dependent on methylation state while the substitution of "C" to "T" decreases the amount of uniquely mapable sequences (Robinson et al. 2010a).

The distribution of CpG densities is important for comparisons of data across platforms. None of the protocols available today is able to produce data which exactly mimics the genome-wide CpG density distribution and, in addition, a few protocols seem to have a strong bias towards high CpG density (RRBS and MBP-Seq) (Robinson et al. 2010a).

In our enrichment-based cancer methylome analyses we find copy number alterations as an important cause for bias, and therefore perform experiments for copy number alterations in parallel (unpublished data). One possibility used so far is to omit known regions of amplification (Ruike et al. 2010). Instead of excluding "no-analysis" areas it is desirable to determine copy number levels and correct the coverage data accordingly. Unfortunately no algorithm exists to date to accomplish this task.

A strategy to minimize potential bias is to assess relative rather than absolute differences between samples (Li et al. 2010). In this way the observed variance gets reduced to technical reproducibility and biological variability while systemic bias becomes eliminated.

Faced with the increasing amount of software and the work needed to compare and evaluate the analysis strategies one has to carefully weigh the need for normalization against the scientific question at hand – a few endeavors, as for instance tumor classification, might work quite well using only raw data (Boerno et al. 2011).

9.4.5 Tertiary Analyses

A typical analysis yields hundreds of genes with disease specific methylation profiles in the promoter and/or gene body. Several bioinformatics methods to identify common patterns among these genes can be applied like over-representation analysis of Gene Ontology terms, sequence motive discovery or genomic clustering analyses.

Another common approach is to compare the methylation profiles to complementary data, like e.g. expression profiles or protein interaction networks which will help to reverse engineer epigenetic regulation by methylation. An exciting approach combining methylation profiles with expression data would be to correlate all sites of differential methylation with all differentially regulated transcripts for a number of samples as has been exemplified for copy number data by Yuan et al. (2010). This approach could help to disentangle cis and trans effects of differential methylation. Hidden Markov models become even more important as they might allow to infer epigenetic states from methylation profiles, an analysis strategy that has already proven valuable for histone modifications (Filion et al. 2010). As DNA methylation bioinformatics, biostatistics and computational biology are under rapid development many tools have yet to be developed – with exciting times ahead!

9.5 Conclusion

Until quite recently the immense progress in the field of cancer genomics has been inconceivable because of the lack of adequate technologies and the limitations in performing genome-wide studies. Recent advances in molecular biology, in particular the development of high-throughput sequencing technologies, have made it possible to gain profound insight into complex biological systems and to analyze the underlying networks responsible for the functionality in healthy and diseased states. These HTS technologies have been proven to be not only extremely useful for genetic, but also for large-scale epigenetic studies. With regard to epigenetic changes in cancer it has been shown that DNA methylation and histone modifications play essential roles in tumor initiation and progression. A number of tumor biomarkers based on aberrant methylation profiles have been developed so far and are tested as potential markers for early diagnosis and risk assessment (Lopez et al. 2009). However, the complex interplay between different aberrant methylation sites or the influence of mutations and epigenetic alterations on gene expression has just started to be addressed.

For the examination of epigenetic alterations using HTS technologies mainly established epigenetic methods have been adapted to HTS protocols. This is done by either an indirect assessment of methylated DNA regions by enrichment or by using a direct "labeling" of genomic DNA with bisulfite treatment followed by HTS. Both approaches can be used for genome-wide investigations; they mainly differ in the amount of sequencing capacities required and the depth of genomic resolution. Regardless of which approach is used, the beauty of the combination of epigenetic technology and HTS is a genome-wide readout of methylation patterns. Unless a reduction of genome complexity is explicitly desired no pre-selection of investigated regions is required as is the case with array or PCR-based technologies. This opens up a large field of new questions to be addressed and it is without doubt that new insights will be achieved with regard to tumor markers as well as molecular-biological mechanisms underlying tumor development.

9.6 Future Perspectives

With the combination of advanced epigenetic techniques and HTS, additional novel genes or DNA regions that contribute to tumorigenesis are certain to be identified. Since epigenetic marks are chemically stable and relatively easy to detect, they are attractive biomarkers in oncology. In addition, specialized protocols permit the extraction and conversion of DNA from formalin-fixed and paraffin-embedded (FFPE) tissue samples (Bian et al. 2001). Patients' samples at pathology departments are routinely stored as FFPE samples and their use would open up access to a variety of clinical trials and would enable routine diagnostic work-ups of patients. However, an FFPE preparation is incompatible with many downstream molecular biology

techniques such as PCR-based amplification methods and gene expression studies. Using HTS technologies we were able to show that small samples of over 20-year-old FFPE material can be used for HTS (Schweiger et al. 2009). With these experiments it is highly likely that BS-converted FFPE DNA can also be used for HTS analyses. We and others have already performed MeDIP-seq experiments which indicate the usability of FFPE tissues. Furthermore, DNA methylation assays can be performed on small numbers of cells obtained by laser capture micro-dissection as well as on DNA extracted from diverse body fluids such as blood, urine or sputum (Kerjean et al. 2001). The combinations of all these methods open up a broad field of clinically or molecular biologically relevant questions including the problem of tumor evolution from single tumor stem cells.

Besides these future developments in oncology we will also experience powerful advancements in HTS technologies: As indicated by the name the throughput has increased enormously, but several enrichment, amplification and labeling steps still cause the performance to be relatively time and cost-intensive. In comparison, future nanopore and scanning probe sequencing approaches, the so-called "third generation sequencers", are directed towards sequencing of single DNA molecules without any prior amplification or labeling (Branton et al. 2008; Lund and Parviz 2009; Pushkarev et al. 2009) and, most importantly, they can detect all "five" nucleotides (A, T, C, G, 5mC) during one sequencing step. However, since these technologies are still under development, it will indeed take some time until they are used for methylation studies, and in the meantime "conventional" approaches such as those described in this review will be emphasized.

References

Aird, D., Ross, M.G., Chen, W.S., Danielsson, M., Fennell, T., Russ, C., Jaffe, D.B., Nusbaum, C., and Gnirke, A. (2011). Analyzing and minimizing PCR amplification bias in Illumina sequencing libraries. Genome Biol *12*, R18.

Ball, M.P., Li, J.B., Gao, Y., Lee, J.H., LeProust, E.M., Park, I.H., Xie, B., Daley, G.Q., and Church, G.M. (2009). Targeted and genome-scale strategies reveal gene-body methylation signatures in human cells. Nat Biotechnol *27*, 361–368.

Banerjee, H.N., and Verma, M. (2009). Epigenetic mechanisms in cancer. Biomark Med *3*, 397–410.

Beck, S., and Rakyan, V.K. (2008). The methylome: approaches for global DNA methylation profiling. Trends Genet *24*, 231–237.

Berman, B.P., Weisenberger, D.J., and Laird, P.W. (2009). Locking in on the human methylome. Nat Biotechnol *27*, 341–342.

Bian, Y.S., Yan, P., Osterheld, M.C., Fontolliet, C., and Benhattar, J. (2001). Promoter methylation analysis on microdissected paraffin-embedded tissues using bisulfite treatment and PCR-SSCP. Biotechniques *30*, 66–72.

Boerno, S.T., Fischer, A., Kerick, M., Muench, P.C., Tusche, C., McHardy, A.C., Faelth, M., Wirth, H., Binder, H., Brase, J.-C., *et al.* (2011). Genome-wide catalogue of DNA-methylation in human prostate cancer. Under review.

Branton, D., Deamer, D.W., Marziali, A., Bayley, H., Benner, S.A., Butler, T., Di Ventra, M., Garaj, S., Hibbs, A., Huang, X., *et al.* (2008). The potential and challenges of nanopore sequencing. Nat Biotechnol *26*, 1146–1153.

Brunner, A.L., Johnson, D.S., Kim, S.W., Valouev, A., Reddy, T.E., Neff, N.F., Anton, E., Medina, C., Nguyen, L., Chiao, E., *et al.* (2009). Distinct DNA methylation patterns characterize differentiated human embryonic stem cells and developing human fetal liver. Genome Res *19*, 1044–1056.

Carr, I.M., Valleley, E.M., Cordery, S.F., Markham, A.F., and Bonthron, D.T. (2007). Sequence analysis and editing for bisulphite genomic sequencing projects. Nucleic Acids Res *35*, e79.

Chavez, L., Jozefczuk, J., Grimm, C., Dietrich, J., Timmermann, B., Lehrach, H., Herwig, R., and Adjaye, J. (2010). Computational analysis of genome-wide DNA methylation during the differentiation of human embryonic stem cells along the endodermal lineage. Genome Res *20*, 1441–1450.

Chen, P.Y., Cokus, S.J., and Pellegrini, M. (2010). BS Seeker: precise mapping for bisulfite sequencing. BMC Bioinformatics *11*, 203.

Cokus, S.J., Feng, S., Zhang, X., Chen, Z., Merriman, B., Haudenschild, C.D., Pradhan, S., Nelson, S.F., Pellegrini, M., and Jacobsen, S.E. (2008). Shotgun bisulphite sequencing of the Arabidopsis genome reveals DNA methylation patterning. Nature *452*, 215–219.

Cross, S.H., Charlton, J.A., Nan, X., and Bird, A.P. (1994). Purification of CpG islands using a methylated DNA binding column. Nat Genet *6*, 236–244.

Deng, J., Shoemaker, R., Xie, B., Gore, A., LeProust, E.M., Antosiewicz-Bourget, J., Egli, D., Maherali, N., Park, I.H., Yu, J., *et al.* (2009). Targeted bisulfite sequencing reveals changes in DNA methylation associated with nuclear reprogramming. Nat Biotechnol *27*, 353–360.

Dohm, J.C., Lottaz, C., Borodina, T., and Himmelbauer, H. (2008). Substantial biases in ultra-short read data sets from high-throughput DNA sequencing. Nucleic Acids Res *36*, e105.

Down, T.A., Rakyan, V.K., Turner, D.J., Flicek, P., Li, H., Kulesha, E., Graf, S., Johnson, N., Herrero, J., Tomazou, E.M., *et al.* (2008). A Bayesian deconvolution strategy for immunoprecipitation-based DNA methylome analysis. Nat Biotechnol *26*, 779–785.

Eckhardt, F., Lewin, J., Cortese, R., Rakyan, V.K., Attwood, J., Burger, M., Burton, J., Cox, T.V., Davies, R., Down, T.A., *et al.* (2006). DNA methylation profiling of human chromosomes 6, 20 and 22. Nat Genet *38*, 1378–1385.

Esteller, M. (2002). CpG island hypermethylation and tumor suppressor genes: a booming present, a brighter future. Oncogene *21*, 5427–5440.

Feinberg, A.P. (2007). Phenotypic plasticity and the epigenetics of human disease. Nature *447*, 433–440.

Feinberg, A.P., and Vogelstein, B. (1983). Hypomethylation distinguishes genes of some human cancers from their normal counterparts. Nature *301*, 89–92.

Filion, G.J., van Bemmel, J.G., Braunschweig, U., Talhout, W., Kind, J., Ward, L.D., Brugman, W., de Castro, I.J., Kerkhoven, R.M., Bussemaker, H.J., *et al.* (2010). Systematic protein location mapping reveals five principal chromatin types in Drosophila cells. Cell *143*, 212–224.

Frigola, J., Song, J., Stirzaker, C., Hinshelwood, R.A., Peinado, M.A., and Clark, S.J. (2006). Epigenetic remodeling in colorectal cancer results in coordinate gene suppression across an entire chromosome band. Nat Genet *38*, 540–549.

Frommer, M., McDonald, L.E., Millar, D.S., Collis, C.M., Watt, F., Grigg, G.W., Molloy, P.L., and Paul, C.L. (1992). A genomic sequencing protocol that yields a positive display of 5-methylcytosine residues in individual DNA strands. Proc Natl Acad Sci USA *89*, 1827–1831.

Gama-Sosa, M.A., Slagel, V.A., Trewyn, R.W., Oxenhandler, R., Kuo, K.C., Gehrke, C.W., and Ehrlich, M. (1983). The 5-methylcytosine content of DNA from human tumors. Nucleic Acids Res *11*, 6883–6894.

Gebhard, C., Schwarzfischer, L., Pham, T.H., Andreesen, R., Mackensen, A., and Rehli, M. (2006). Rapid and sensitive detection of CpG-methylation using methyl-binding (MB)-PCR. Nucleic Acids Res *34*(11), e82.

Hodges, E., Smith, A.D., Kendall, J., Xuan, Z., Ravi, K., Rooks, M., Zhang, M.Q., Ye, K., Bhattacharjee, A., Brizuela, L., *et al.* (2009). High definition profiling of mammalian DNA methylation by array capture and single molecule bisulfite sequencing. Genome Res *19*, 1593–1605.

Huang, Y., Pastor, W.A., Shen, Y., Tahiliani, M., Liu, D.R., and Rao, A. (2010). The behaviour of 5-hydroxymethylcytosine in bisulfite sequencing. PLoS One *5*, e8888.

Irizarry, R.A., Ladd-Acosta, C., Carvalho, B., Wu, H., Brandenburg, S.A., Jeddeloh, J.A., Wen, B., and Feinberg, A.P. (2008). Comprehensive high-throughput arrays for relative methylation (CHARM). Genome Res *18*, 780–790.

Jacinto, F.V., Ballestar, E., and Esteller, M. (2008). Methyl-DNA immunoprecipitation (MeDIP): hunting down the DNA methylome. Biotechniques *44*(1), 35, 37, 39 passim.

Jin, S.G., Kadam, S., and Pfeifer, G.P. (2010). Examination of the specificity of DNA methylation profiling techniques towards 5-methylcytosine and 5-hydroxymethylcytosine. Nucleic Acids Res *38*, e125.

Jorgensen, H.F., Adle, K., Chaubert, P., and Bird, A.P. (2006). Engineering a high-affinity methyl-CpG-binding protein. Nucleic Acids Res *34*(13), e96.

Kerjean, A., Vieillefond, A., Thiounn, N., Sibony, M., Jeanpierre, M., and Jouannet, P. (2001). Bisulfite genomic sequencing of microdissected cells. Nucleic Acids Res *29*, E106–106.

Keshet, I., Schlesinger, Y., Farkash, S., Rand, E., Hecht, M., Segal, E., Pikarski, E., Young, R.A., Niveleau, A., Cedar, H., *et al.* (2006). Evidence for an instructive mechanism of de novo methylation in cancer cells. Nat Genet *38*, 149–153.

Korshunova, Y., Maloney, R.K., Lakey, N., Citek, R.W., Bacher, B., Budiman, A., Ordway, J.M., McCombie, W.R., Leon, J., Jeddeloh, J.A., *et al.* (2008). Massively parallel bisulphite pyrosequencing reveals the molecular complexity of breast cancer-associated cytosine-methylation patterns obtained from tissue and serum DNA. Genome Res *18*, 19–29.

Kriaucionis, S., and Heintz, N. (2009). The nuclear DNA base 5-hydroxymethylcytosine is present in Purkinje neurons and the brain. Science *324*, 929–930.

Laird, P.W. (2010). Principles and challenges of genome-wide DNA methylation analysis. Nat Rev Genet *11*, 191–203.

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

Li, H., and Durbin, R. (2009). Fast and accurate short read alignment with Burrows-Wheeler transform. Bioinformatics *25*, 1754–1760.

Li, H., Ruan, J., and Durbin, R. (2008). Mapping short DNA sequencing reads and calling variants using mapping quality scores. Genome Res *18*, 1851–1858.

Li, J.B., Gao, Y., Aach, J., Zhang, K., Kryukov, G.V., Xie, B., Ahlford, A., Yoon, J.K., Rosenbaum, A.M., Zaranek, A.W., *et al.* (2009). Multiplex padlock targeted sequencing reveals human hypermutable CpG variations. Genome Res *19*, 1606–1615.

Li, N., Ye, M., Li, Y., Yan, Z., Butcher, L.M., Sun, J., Han, X., Chen, Q., Zhang, X., and Wang, J. (2010). Whole genome DNA methylation analysis based on high throughput sequencing technology. Methods *52*, 203–212.

Lister, R., and Ecker, J.R. (2009). Finding the fifth base: genome-wide sequencing of cytosine methylation. Genome Res *19*, 959–966.

Lister, R., O'Malley, R.C., Tonti-Filippini, J., Gregory, B.D., Berry, C.C., Millar, A.H., and Ecker, J.R. (2008). Highly integrated single-base resolution maps of the epigenome in Arabidopsis. Cell *133*, 523–536.

Lopez, J., Percharde, M., Coley, H.M., Webb, A., and Crook, T. (2009). The context and potential of epigenetics in oncology. Br J Cancer *100*, 571–577.

Lund, J., and Parviz, B.A. (2009). Scanning probe and nanopore DNA sequencing: core techniques and possibilities. Methods Mol Biol *578*, 113–122.

Meissner, A., Gnirke, A., Bell, G.W., Ramsahoye, B., Lander, E.S., and Jaenisch, R. (2005). Reduced representation bisulfite sequencing for comparative high-resolution DNA methylation analysis. Nucleic Acids Res *33*, 5868–5877.

Meissner, A., Mikkelsen, T.S., Gu, H., Wernig, M., Hanna, J., Sivachenko, A., Zhang, X., Bernstein, B.E., Nusbaum, C., Jaffe, D.B., *et al.* (2008). Genome-scale DNA methylation maps of pluripotent and differentiated cells. Nature *454*, 766–770.

Nestor, C., Ruzov, A., Meehan, R., and Dunican, D. (2010). Enzymatic approaches and bisulfite sequencing cannot distinguish between 5-methylcytosine and 5-hydroxymethylcytosine in DNA. Biotechniques *48*, 317–319.

Oda, M., Glass, J.L., Thompson, R.F., Mo, Y., Olivier, E.N., Figueroa, M.E., Selzer, R.R., Richmond, T.A., Zhang, X., Dannenberg, L., *et al.* (2009). High-resolution genome-wide

cytosine methylation profiling with simultaneous copy number analysis and optimization for limited cell numbers. Nucleic Acids Res *37*, 3829–3839.

Pepke, S., Wold, B., and Mortazavi, A. (2009). Computation for ChIP-seq and RNA-seq studies. Nat Methods *6*, S22–32.

Pomraning, K.R., Smith, K.M., and Freitag, M. (2009). Genome-wide high throughput analysis of DNA methylation in eukaryotes. Methods *47*, 142–150.

Popp, C., Dean, W., Feng, S., Cokus, S.J., Andrews, S., Pellegrini, M., Jacobsen, S.E., and Reik, W. (2010). Genome-wide erasure of DNA methylation in mouse primordial germ cells is affected by AID deficiency. Nature *463*, 1101–1105.

Pushkarev, D., Neff, N.F., and Quake, S.R. (2009). Single-molecule sequencing of an individual human genome. Nat Biotechnol *27*, 847–850.

Rakyan, V.K., Down, T.A., Thorne, N.P., Flicek, P., Kulesha, E., Graf, S., Tomazou, E.M., Backdahl, L., Johnson, N., Herberth, M., *et al.* (2008). An integrated resource for genome-wide identification and analysis of human tissue-specific differentially methylated regions (tDMRs). Genome Res *18*, 1518–1529.

Rauch, T., and Pfeifer, G.P. (2005). Methylated-CpG island recovery assay: a new technique for the rapid detection of methylated-CpG islands in cancer. Lab Invest *85*, 1172–1180.

Robinson, M.D., Statham, A.L., Speed, T.P., and Clark, S.J. (2010a). Protocol matters: which methylome are you actually studying? Epigenomics *2*, 587–598.

Robinson, M.D., Stirzaker, C., Statham, A.L., Coolen, M.W., Song, J.Z., Nair, S.S., Strbenac, D., Speed, T.P., and Clark, S.J. (2010b). Evaluation of affinity-based genome-wide DNA methylation data: effects of CpG density, amplification bias, and copy number variation. Genome Res *20*, 1719–1729.

Ruike, Y., Imanaka, Y., Sato, F., Shimizu, K., and Tsujimoto, G. (2010). Genome-wide analysis of aberrant methylation in human breast cancer cells using methyl-DNA immunoprecipitation combined with high-throughput sequencing. BMC Genomics *11*, 137.

Schweiger, M.R., Kerick, M., Timmermann, B., Albrecht, M.W., Borodina, T., Parkhomchuk, D., Zatloukal, K., and Lehrach, H. (2009). Genome-wide massively parallel sequencing of formaldehyde fixed-paraffin embedded (FFPE) tumor tissues for copy-number- and mutation-analysis. PLoS One *4*, e5548.

Taylor, K.H., Kramer, R.S., Davis, J.W., Guo, J., Duff, D.J., Xu, D., Caldwell, C.W., and Shi, H. (2007). Ultradeep bisulfite sequencing analysis of DNA methylation patterns in multiple gene promoters by 454 sequencing. Cancer Res *67*, 8511–8518.

Timmermann, B., Kerick, M., Roehr, C., Fischer, A., Isau, M., Boerno, S.T., Wunderlich, A., Barmeyer, C., Seemann, P., Koenig, J., *et al.* (2010). Somatic mutation profiles of MSI and MSS colorectal cancer identified by whole exome next generation sequencing and bioinformatics analysis. PLoS One *5*, e15661.

Toyota, M., Ho, C., Ahuja, N., Jair, K.W., Li, Q., Ohe-Toyota, M., Baylin, S.B., and Issa, J.P. (1999). Identification of differentially methylated sequences in colorectal cancer by methylated CpG island amplification. Cancer Res *59*, 2307–2312.

Walsh, C.P., Chaillet, J.R., and Bestor, T.H. (1998). Transcription of IAP endogenous retroviruses is constrained by cytosine methylation. Nat Genet *20*, 116–117.

Wang, R.Y., Gehrke, C.W., and Ehrlich, M. (1980). Comparison of bisulfite modification of 5-methyldeoxycytidine and deoxycytidine residues. Nucleic Acids Res *8*, 4777–4790.

Warnecke, P.M., Stirzaker, C., Melki, J.R., Millar, D.S., Paul, C.L., and Clark, S.J. (1997). Detection and measurement of PCR bias in quantitative methylation analysis of bisulphite-treated DNA. Nucleic Acids Res *25*, 4422–4426.

Weber, M., Davies, J.J., Wittig, D., Oakeley, E.J., Haase, M., Lam, W.L., and Schubeler, D. (2005). Chromosome-wide and promoter-specific analyses identify sites of differential DNA methylation in normal and transformed human cells. Nat Genet *37*, 853–862.

Weisenberger, D.J., Campan, M., Long, T.I., Kim, M., Woods, C., Fiala, E., Ehrlich, M., and Laird, P.W. (2005). Analysis of repetitive element DNA methylation by MethyLight. Nucleic Acids Res *33*, 6823–6836.

Wilbanks, E.G., and Facciotti, M.T. (2010). Evaluation of algorithm performance in ChIP-seq peak detection. PLoS One *5*, e11471.

Wu, T.D., and Nacu, S. (2010). Fast and SNP-tolerant detection of complex variants and splicing in short reads. Bioinformatics *26*, 873–881.

Xi, Y., and Li, W. (2009). BSMAP: whole genome bisulfite sequence MAPping program. BMC Bioinformatics *10*, 232.

Yuan, Y., Curtis, C., Caldas, C., and Markowetz, F. (2010). A sparse regulatory network of copy-number driven expression reveals putative breast cancer oncogenes. In ArXiv e-prints, pp. 1409.

Zhang, Y., Liu, T., Meyer, C.A., Eeckhoute, J., Johnson, D.S., Bernstein, B.E., Nusbaum, C., Myers, R.M., Brown, M., Li, W.*, et al.* (2008). Model-based analysis of ChIP-Seq (MACS). Genome Biol *9*, R137.

Chapter 10
Differential Expression for RNA Sequencing (RNA-Seq) Data: Mapping, Summarization, Statistical Analysis, and Experimental Design

Matthew D. Young, Davis J. McCarthy, Matthew J. Wakefield, Gordon K. Smyth, Alicia Oshlack, and Mark D. Robinson

Abstract RNA sequencing (RNA-seq) is an exciting technique that gives experimenters unprecedented access to information on transcriptome complexity. The costs are decreasing, data analysis methods are maturing, and the flexibility that RNA-seq affords will allow it to become the platform of choice for gene expression analysis. Here, we focus on differential expression (DE) analysis using RNA-seq, highlighting aspects of mapping reads to a reference transcriptome, quantification

M.D. Young • D.J. McCarthy
Bioinformatics Division, Walter and Eliza Hall Institute, Melbourne, VIC 3050, Australia
e-mail: myworkemailisnow@gmail.com; davis.mccarthy@balliol.ox.ac.uk

M.J. Wakefield
Bioinformatics Division, Walter and Eliza Hall Institute, Melbourne, VIC 3050, Australia

Department of Zoology, University of Melbourne, Melbourne, VIC 3010, Australia
e-mail: wakefield@wehi.EDU.AU

G.K. Smyth
Bioinformatics Division, Walter and Eliza Hall Institute, Melbourne, VIC 3050, Australia

Department of Mathematics and Statistics, University of Melbourne, Melbourne, VIC 3010, Australia
e-mail: smyth@wehi.edu.au

A. Oshlack
Bioinformatics Division, Walter and Eliza Hall Institute, Melbourne, VIC 3050, Australia

School of Physics, University of Melbourne, Melbourne, VIC 3010, Australia

Murdoch Childrens Research Institute, Parkville, VIC 3052, Australia
e-mail: alicia.oshlack@mcri.edu.au

M.D. Robinson (✉)
Bioinformatics Division, Walter and Eliza Hall Institute, Melbourne, VIC 3050, Australia

Department of Medical Biology, University of Melbourne, Melbourne, VIC 3010, Australia

Epigenetics Laboratory, Cancer Research Program, Garvan Institute of Medical Research, Darlinghurst, NSW 2010, Australia
e-mail: mark.robinson@imls.uzh.ch

N. Rodríguez-Ezpeleta et al. (eds.), *Bioinformatics for High Throughput Sequencing*,
DOI 10.1007/978-1-4614-0782-9_10, © Springer Science+Business Media, LLC 2012

of expression levels, normalization for composition biases, statistical modeling to account for biological variability and experimental design considerations. We also comment on recent developments beyond the analysis of DE using RNA-seq.

10.1 Introduction

RNA-seq is a rich family of methods that quantitatively measures genome-wide expression at single base resolution. It differs from tag-based methods such as serial analysis of gene expression (SAGE) and cap analysis of gene expression (CAGE) in that it produces multiple distinct sequences from each transcript. RNA-seq differs from traditional transcriptome sequencing by aiming to produce a quantitative sample rather than a normalized comprehensive description of the sequence. RNA-seq utilizes the ability of high-throughput sequencing (HTS) platforms (such as the Illumina/Solexa HiSeq or LifeTech/ABI SOLiD) to sequence a large number of short DNA fragments at a cost and throughput that allows each transcript to be observed a sufficient number of times for the measurement to be quantitative.

Constructing the library of fragments to be sequenced requires isolation of RNA, random fragmentation of the transcripts into smaller pieces, conversion of the RNA into DNA by reverse transcription, ligation of adapter sequences for amplification, fragment size selection, and priming the sequencing reaction. Each of these steps can vary by the implementation of the protocol and will introduce specific technical biases in the resulting data.

As most applications of RNA-seq aim to measure the abundance of protein-coding genes, polyA-purified mRNA is usually used for library construction to minimize the sequencing of uninformative and abundant ribosomal transcripts (rRNA). In cases where long non-coding RNAs need to be measured, the alternative protocol of ribosomal depletion can be employed. For small non-coding RNAs, size selection is usually employed to remove both mRNA and rRNA, or selective ligation of adapters to 3′-hydroxyl ends.

For the study of protein-coding genes, the optimal size of DNA fragment for reverse transcription and sequencing on second-generation sequencers is 200–350 base pairs. RNA-seq uses physical, chemical (NaOH), or enzymatic (RNAseIII) fragmentation to reduce transcripts (averaging 2.2 kb) to a distribution of random sizes that includes the efficiently sequenced lengths. The optimal size range is then selected by gel purification or SPRI beads, usually after cDNA synthesis and adaptor ligation.

Synthesis of cDNA requires priming with a short sequence. This is achieved by using randomly synthesized oligonucleotides or ligating known sequences onto the RNA. Sequences complementary to the ligated oligonucleotides are used for priming cDNA synthesis, and the oligos include the sequences necessary for amplifying and priming the sequencing reaction. When random priming is performed, the necessary sequences are added by ligating double-stranded adaptors to the double-stranded cDNA. This results in the loss of any information about strand specificity unless specific steps (such as the inclusion of dUTP) have been utilized to mark first

or second strand synthesis and actinomycin D is included to prevent DNA-dependent polymerase action of the reverse transcriptase (Levin et al. 2010).

The end result of all of the variations in RNA-seq protocols is, one hopes, sequences from one or both ends of a representative sample of fragments from the RNA population of interest. This measurement will be affected by protocol-specific technical influences such as fragmentation site bias, random priming, or ligase bias. Additional biases such as fragment length and abundance bias that are inherent to all protocols will also be present.

10.2 Microarrays and Sequencing for Gene Expression

Several comparisons of RNA-seq and microarray data have emerged, demonstrating that there is a strong concordance between platforms, especially for detecting DE (Bradford et al. 2010; Cloonan et al. 2008; Fu et al. 2009; Mortazavi et al. 2008; Sultan et al. 2008). These comparisons have also established that sequencing-based approaches are more sensitive and have a larger dynamic range with minimal levels of background and technical variation (‘t Hoen et al. 2008; Liu et al. 2011; Marioni et al. 2008).

No genome-scale assay is without its biases, and platform-specific nuances can affect an assay’s overall performance. For example, cross-hybridization on microarrays can have significant effects on probe intensities (Naef and Magnasco 2003; Wu and Irizarry 2005), as can probe sequence content (Binder et al. 2004). As a result, microarrays are not generally used for comparing expression levels between genes, but work well for comparisons of the same gene across multiple experimental conditions. On the other hand, one of the purported benefits of RNA-seq is the ability to compare expression levels between genes in a single sample. However, there are still limitations. For example, GC bias is present in RNA-seq data (Bullard et al. 2010) and ambiguity in mapping can affect some regions more than others. Similarly, comparing gene expression levels across experimental conditions using RNA-seq has its own biases. For example, statistical power to detect changes is greater at higher counts, thus introducing a clear association between DE and gene length, an effect not present in microarray data (Oshlack and Wakefield 2009; Young et al. 2010). Furthermore, due to the high sensitivity of sequencing, the protocols used to extract RNA, enrich for subpopulations, fragment and convert RNA to cDNA all have a large potential to introduce prominent biases. For example, studies thus far have identified biases in sequence composition due to hexamer priming and biases in the distribution of observed reads along a transcript (Hansen et al. 2010; Quail et al. 2008; Wang et al. 2009). Furthermore, the method by which small RNAs are captured has been found to strongly affect the set of observable sequences (Linsen et al. 2009). Similarly, de novo transcriptome assembly approaches are necessarily biased by expression level, since more information is available for highly expressed genes (Robertson et al. 2010; Trapnell et al. 2010).

Beyond dynamic range and sensitivity, there are several further reasons that explain the swift transition from microarrays to sequencing for many research

groups. First, sequencing can be used for any organism, even where no reference genome or transcriptome exists. While custom microarray design is now widely available, many commercial providers only make the platform available for a small number of model organisms. In contrast, genomes are available for thousands of species (NCBI 2011), and genomes for many more organisms will continue to appear rapidly as sequencing costs decrease. Furthermore, sequencing can more readily reveal information regarding features such as novel transcribed regions and alternative isoforms, as well as provide information that arrays simply cannot, such as allele-specific expression, RNA-editing, and the discovery of fusion genes.

Despite all these advantages, RNA-seq data is complex and the data analysis can be a bottleneck. Furthermore, the cost of the platform may be limiting for some studies. The cost per sample to generate sufficient sequence depth (see Sect. 10.7) will soon be comparable to microarrays (if not already), especially with the expanding capacities of current instruments (e.g. Illumina HiSeq 2000) and the ability to multiplex, whereby multiple samples are sequenced simultaneously in a single experiment. The required informatics infrastructure for processing even moderately sized datasets is non-trivial and the cost of storing and processing the large amounts of data is easily underestimated (Schadt et al. 2010). On the other hand, microarray data analysis procedures are relatively mature and often can be run on desktop computers, so researchers with basic gene expression needs in model organisms may still choose to use microarrays.

10.3 Mapping

The first step towards quantifying gene expression levels using RNA-seq is to "map" the millions of short reads to a suitable reference genome or transcriptome. The goal of mapping is to find the unique location where a short read is identical, or as close as possible, to the reference sequence. The reference is used as a guide, but the mapping procedure must be flexible enough to accommodate sample-specific attributes, such as single nucleotide polymorphisms (SNPs), insertions/deletion (indels), and sequencing errors that will inevitably be present in some sequences. Furthermore, unlike mapping genomic DNA, the transcriptome is "built from" the genome by splicing out intronic regions, so the mapping procedure must adjust for this additional complexity. There is also the problem of multimapping, whereby reads can align equally well to multiple locations, requiring some solution to this ambiguity.

All mapping solutions by necessity involve some compromise between the flexibility of mismatches between the read and the reference and the computational demands of the algorithm. Generally, short read alignment algorithms utilize a "heuristic" first pass step to rapidly find likely candidates, followed by a computationally demanding "local alignment". Local alignment strategies are simply too inefficient to be applied from the outset to even moderately sized genomes.

There are many different aligners currently in use, but almost all of them use either hash-tables or the Burrows Wheeler Transform (BWT) to enable fast heuristic

matching, which reduces the number of local alignments performed. A detailed explanation of how these techniques achieve rapid matching to the genome is beyond the scope of this chapter, but the details can be found in (Ferragina and Manzini 2000; Flicek and Birney 2009; Li and Durbin 2009; Li et al. 2008). The hash-table-based approach has the advantage of being guaranteed to find all structural variants up to the level of variation specified by the user. However, the memory requirements for increasing the level of detectable variation from the reference rise rapidly and hashed references need to be remade for each experiment. On the other hand, the BWT approach needs to store only one copy of the (transformed) reference in memory, regardless of the level of variation that is detectable. This technique is orders of magnitude faster than the hash-table approach when only a small number (<4) of mismatches from the reference are allowed. The downside to the BWT approach is that the method for determining mismatches involves trying to align a large number of variants of each read, each of which has a comparable computational cost to performing a full alignment of a perfectly matching read. To mitigate this computational cost, many aligners do not explore all possible mismatched alignments, meaning that some valid alignments may be missed. Furthermore, the rapid scaling of complexity places a hard ceiling on the number of mismatches that can be detected when aligning reads using the BWT, with few aligners allowing more than three possible mismatches in the portion of the read used for the first pass and limited ability to find insertions or deletions.

In order to increase the computational efficiency and minimize the loss in sensitivity as a result of the heuristic alignment algorithms, the heuristic is usually applied not to the entire read, but to an n base substring (usually, the first n bases), called the seed. Once the heuristic algorithm has identified all potential mapping locations using only the seed, the seed is extended back to the full read and each of these locations is evaluated using a local alignment routine. Therefore, the choice of seed length involves a tradeoff between speed and sensitivity. A longer seed will mean fewer putative matching sites that need to be ranked with local alignment algorithms. However, as the heuristic matching algorithm is always less sensitive than a local alignment, a longer seed can cause more valid alignments to be missed. On the other hand, a short seed minimizes the loss in sensitivity due to the heuristic alignment, but at the cost of having to sort through many more potential mapping locations with the slower local alignment algorithm.

Aligners also differ in how they handle reads that map equally well to several locations. Many algorithms discard them (Langmead et al. 2009), randomly allocate them (Li et al. 2008), or are guided by local coverage (Cloonan et al. 2008; Mortazavi et al. 2008). Recently, a statistical method that uses alignment scores has also been proposed (Taub and Speed 2010). Paired-end reads reduce the problem of multimapping, since both ends of the cDNA fragment should map nearby on the transcriptome, allowing the ambiguity of multimaps to be resolved in many cases.

The most common choice of reference to map reads against, at least initially, is the genome itself. This has the benefit of being independent of annotation. However, reads that cross exon–exon junctions will not map to the reference genome.

Specifically, under this framework, lower coverage will be observed on average (given the same expression level) on transcripts with shorter exons, since they will contain more junctions. This situation will be exacerbated by longer reads, since more junctions will be covered (Sultan et al. 2008). In order to account for reads that span junctions, it is common practice to supplement the genomic reference with exon-junction libraries, i.e., small sections of sequence that incorporate the junctions, given by an annotation database (Marioni et al. 2008; Mortazavi et al. 2008; Pickrell et al. 2010; Sultan et al. 2008). To map reads that cross exon boundaries without relying on existing annotations, it is possible to use the data itself to detect splice junctions de novo (Ameur et al. 2010; De Bona et al. 2008). Another option, if the depth of reads is sufficient, is de novo assembly of the transcriptome, for use as a reference, using assembly tools (Robertson et al. 2010; Simpson et al. 2009; Zerbino and Birney 2008). All de novo methods have the ability to identify novel transcripts and may be the only option for organisms for which no genomic reference or annotation is available. However, de novo methods are computationally intensive and may require long, paired-end reads and high levels of coverage to work reliably.

10.4 Summarization

In order to estimate expression levels for some biological entity of interest (e.g. exons, transcripts or genes), the next step is to summarize reads into a "table of counts," which records the number of reads associated with each entity. The simplest such approach records the number of reads overlapping the exons in a gene (e.g. Bullard et al. 2010; Marioni et al. 2008; Mortazavi et al. 2008). This simple and popular metric to summarize gene expression levels is a crude summary of the total gene output as it pays little regard to the genuine complexity (e.g. alternative splice forms) present.

Reads often align to genomic regions outside annotated transcripts, even in well-annotated organisms. An alternative to exonic summarization is to include reads that map anywhere between the start and end of a gene, perhaps with the option to include regions adjacent to the start and end to account for poorly annotated start and termination sites. This measure will include unannotated exons in the count, while accounting for poorly annotated or variable UTRs and exon boundaries. In some cases, including intronic regions will include overlapping transcripts that share a genomic location but in reality originate from different genes. The downside to this approach is that if the genomic annotation is accurate, including intronic reads adds noise to the summarized counts in unpredictable ways.

There are many other possible variations that could be used for summarization: for instance, one could include only reads that map to coding sequence or summarize based on de novo predicted exons (Trapnell et al. 2009). Junction reads can also be added into the gene summary count or be used to model the abundance of splicing isoforms (Griffith et al. 2010; Trapnell et al. 2010). These different possibilities are illustrated schematically in Fig. 10.1.

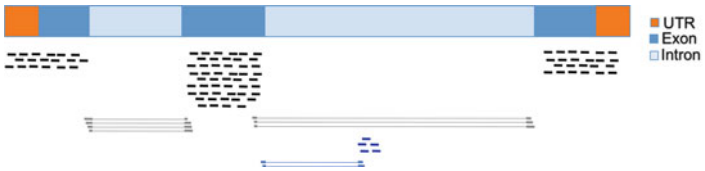

Fig. 10.1 Reads mapping to transcripts. The large colored bars represent a canonical transcript, including untranslated regions (UTRs), exons, and introns. Below the transcript, a schematic of possible mapped reads is shown, highlighting that reads generally map to exons (*black bars*) and exon–exon junctions (*grey bars*), but may be supplemented with novel exons (*blue bars*) and novel exon–exon junctions (*light blue bars*)

RNA-seq offers the further promise of being able not just to identify alternative splicing, but also to quantify the expression level of each isoform in a sample. However, as different transcripts of a gene share most of their sequence, quantifying the expression level of each transcript is a considerable challenge. There have been a number of approaches to solving this problem, all of which utilize sequence regions unique to each transcript (such as transcript-specific exons or splice junctions) to estimate the expression level of each transcript. Reads that map to multiple transcripts can then either be randomly assigned to an isoform (Li et al. 2008), ignored (Langmead et al. 2009) or used to generate a non-count probabilistic measure of transcript expression (such as FPKM) (Trapnell et al. 2010). The problem of assigning these reads to transcripts is, in many ways, analogous to the issue of assigning multimapped reads to the genome discussed in the previous section. As with multimapped reads, each of the proposed solutions has advantages and disadvantages. For example, although the probabilistic FPKM measure may be a more realistic measure of transcript expression, it has complex statistical properties that make it difficult to input to downstream statistical procedures. Many of the most promising statistical models for assessing DE using RNA-seq data require raw counts as input.

The choice of summarization method has the potential to have a large impact on the results of a DE analysis. The tradeoffs between different summarization methods still need to be explored and their relative merits investigated using real datasets. For the purposes of the discussion below, we require that a table of counts be generated by one of the methods mentioned above.

10.5 Normalization

Normalization can be important to ensure that expression measurements are directly comparable. For example, if gene-to-gene comparisons of expression in a single sample are of interest, compensation needs to be made for the length of the genes, since at similar expression levels, longer genes will collect more reads

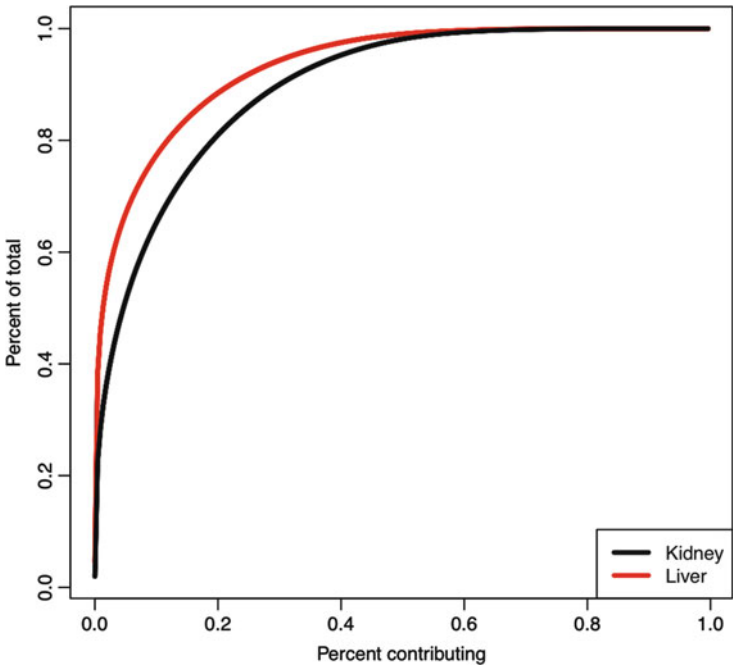

Fig. 10.2 Composition. Genes are ordered by the number of reads mapped to them and plotted here as the percent of total reads by the percent of genes contributing, highlighting that library composition can be very different from sample to sample and can directly affect the depth that each gene gets

(Cloonan et al. 2008; Mortazavi et al. 2008; Oshlack and Wakefield 2009). Our focus here is on comparing expression of the same gene across multiple conditions, where we expect that length-associated and other biases will largely cancel out. In the comparison of expression levels between samples, it is necessary to compensate for the depth of sequencing, which may vary from sample to sample. In addition, cDNA libraries can have very different composition and therefore we should not always expect read densities, even after compensating for total read depth, to be directly comparable. This is illustrated by an example shown in Fig. 10.2. This figure highlights that the percentage of the transcriptome accounting for the observed reads can vary drastically across experimental conditions (Fig. 10.2). That is, a reasonably small number of genes can consume a significant fraction of the sequencing resources, causing an undersampling of the remaining genes (Fig. 10.3). If not explicitly accounted for, this undersampling induces a bias to the detection of DE (Robinson and Oshlack 2010). To compensate for such composition effects, scaling factors in addition to the adjustment made for depth can be calculated. Several strategies have been proposed to determine these factors: first, Bullard et al. proposed computing the 75th percentile of the read counts across each sample (Bullard et al. 2010); second, proposed the trimmed mean of M-values

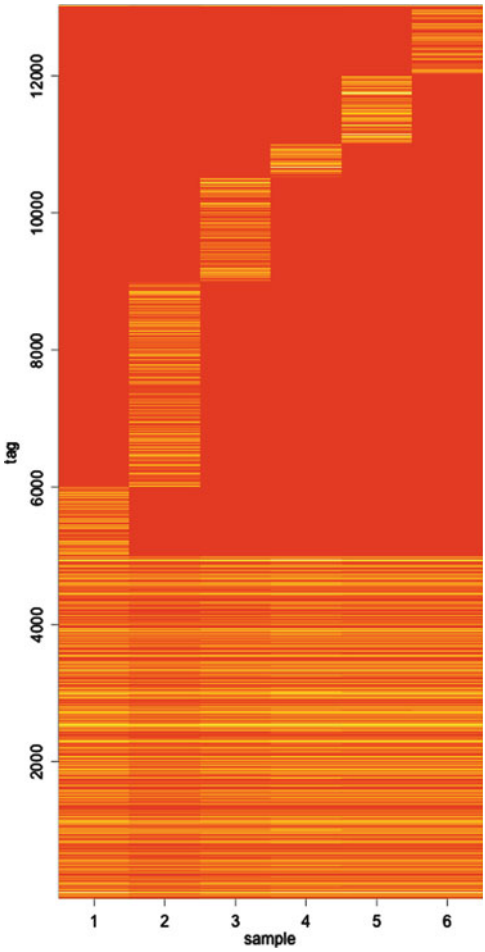

Fig. 10.3 Hypothetical setting of composition biases. Six libraries are sampled (according to an empirical distribution of real RNA-seq counts) to the same overall depth, but with varying composition, i.e. levels of unique-to-sample counts. About 5,000 genes are observed at similar relative levels across all six samples. The heatmap colors represent the sampled expression levels (*red* = 0, *brighter yellow* represents high counts). Each column sums to the same total. These differences in composition can induce artificial (and statistically significant) differences in counts

(TMM), which estimates the additional bias not accounted for in depth-normalized log-ratios; third, (Anders and Huber 2010) calculate a median sample and compute a robust relative expression ratio of every sample to the median sample to account explicitly for both depth and composition.

It is also important to point out that normalization for DE analysis of RNA-seq data need not involve changing the raw count data, unlike the many background correction and normalization procedures that have been suggested for microarray data (Quackenbush 2002). Transforming count data can be problematic, since it puts the data on an arbitrary scale, may not effectively stabilize the variance across the spectrum of expression levels, and may alter the mean–variance relationship. Furthermore, since the calculation of *p*-values is dependent on the raw level of the count, modifications can have substantial effects on these calculations. Further study of the transformation approach is required.

10.6 Statistical Models for Differential Expression

A summarized table of raw counts, augmented with information regarding additional scaling factors, forms the starting point for the statistical analysis of DE. It should also be noted that the methods for DE in RNA-seq data represent a general statistical framework for count data, and may have uses in other genome-scale datasets that can be represented as counts, such as the search for differentially methylated promoters using methylated DNA immunoprecipitation sequencing (MeDIP-seq) data (Bock et al. 2010; Robinson et al. 2010b), finding differentially enriched regions using chromatin immunoprecipitation sequencing (ChIP-seq), spectral counts in tandem mass spectrometry data (Carvalho et al. 2008) or in the analysis of counts from metagenomic data (White et al. 2009).

In terms of statistical models for count-based gene expression data, it is important to note that RNA-seq gives a discrete measurement for each gene, whereas microarray data have a continuous distribution. Since microarrays are typically scanned as 16-bit images (i.e. 65,536 possible values), they are technically discrete as well. But, since the resolution is high, microarray expression data can be assumed to follow a continuous distribution with negligible loss of information. Furthermore, microarray intensities are generally log-transformed to constrain the scale between 0 and 16 (log base 2) and to better stabilize the variance, allowing them to be more appropriately approximated by Gaussian or similar continuously distributed random variables. With very few dedicated tools available, some early adopters of RNA-seq elected to transform their count data (e.g. logarithm or square root) (‘t Hoen et al. 2008; Cloonan et al. 2008), but such transformations cannot be well approximated by continuous distributions in small samples or at low counts. Therefore, we favor statistical models for DE of RNA-seq that are specific to count data; simulations suggest that count-based models are better powered for DE analysis (Robinson and Oshlack 2010).

Sequencing a population of cDNA fragments can be thought of as random multinomial sampling. That is, from a large sample of reads (e.g. tens of millions), each read can be identified to be from one of a number of different genes, ignoring for the time being that some reads cannot be mapped and others can map to multiple locations. Under this framework, the vector of counts for a single sample is a multinomial random variable, with parameters representing the proportion of reads mapping to each gene. For simplicity, we often represent modeling assumptions in the context of a single gene. In the single gene case, the observed count is a binomial random variable, akin to a large-scale coin-tossing experiment. Furthermore, with a large number of reads where each gene represents a small proportion of the reads, the Poisson distribution provides a very good approximation (large number of trials, N and small proportion of “successes”, p). In addition, the mathematical simplicity of the Poisson distribution lends itself to form the basis for modeling RNA-seq count data. For the purposes of a DE analysis, we are modeling the vector of counts for a single gene across experimental conditions. The Poisson model assumes that the mean equals the variance and has been validated in one of the early RNA-seq

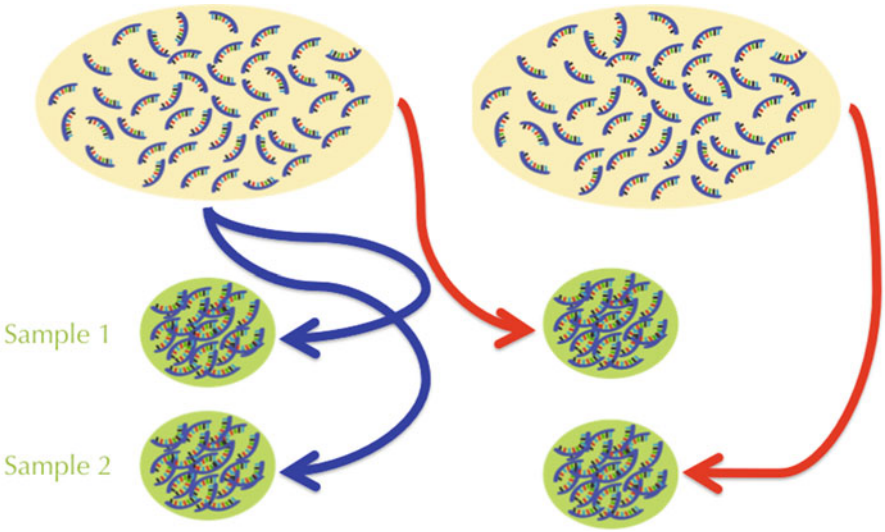

Fig. 10.4 Technical and Biological Replication. If two independent DNA populations from the same experimental conditions are available, one can make technical replicates (*blue* lines, repeat sampling of the same population) or biological replicates (*red* lines). The latter is expected to have higher variability, but conclusions based on this data should be more generalizable to the population

studies using the same initial source of RNA split into multiple lanes of an Illumina GA sequencer (Marioni et al. 2008). However, it is important to note that binomial sampling through the Poisson assumption only accounts for technical variation, the variation in read counts that one can expect by sampling the same DNA library on multiple occasions (see Fig. 10.4).

Genuine biological investigations (e.g. comparing multiple patients) will exhibit higher levels of variation, and will require extensions to the Poisson model. Analyzing biologically replicated data with the Poisson model will likely be prone to high false-positive rates due to the underestimation of the true variability (Anders and Huber 2010; Langmead et al. 2010; Robinson and Smyth 2008). Figure 10.5 shows the expected (Poisson) variation observed in the Marioni dataset and Fig. 10.6 shows the extent of extra-Poisson variation in biologically replicated datasets.

Various strategies have been proposed for modeling biological variability in RNA-seq count data. Methods originally designed for SAGE data have recently been applied to HTS-based digital gene expression data using the negative binomial (NB) distribution, as implemented in the edgeR package (Robinson et al. 2010a). The NB distribution, which arises as a mixture of Poisson distributions where the mixing distribution is a gamma distribution, requires an additional dispersion parameter to be estimated. The Poisson distribution represents the technical variation inherent in sampling fragments, while the gamma distribution models the

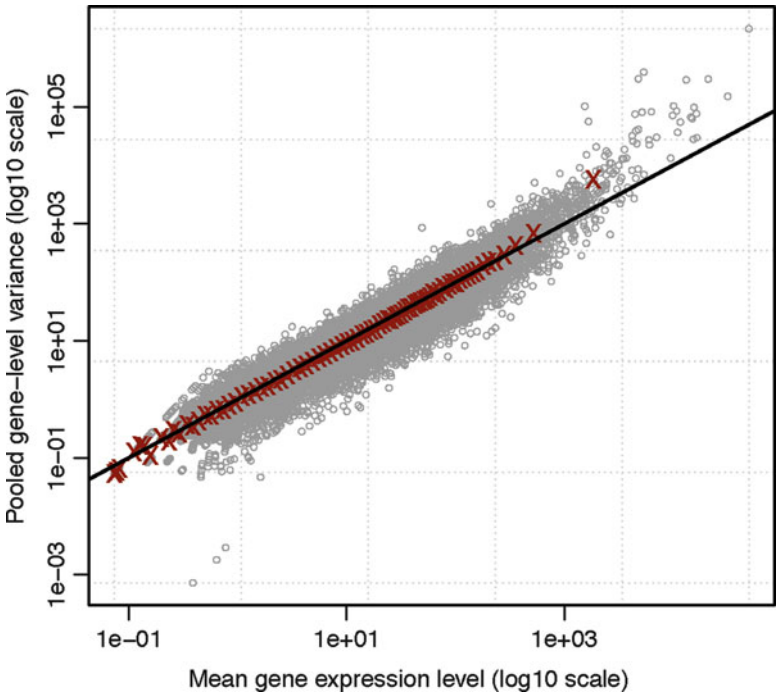

Fig. 10.5 Mean–variance plot for Marioni et al. dataset (Marioni et al. 2008). The variability in technically replicated RNA-seq data can be adequately captured using a Poisson model. The *grey* points in this plot shows the mean and pooled variance for each gene, scaled to account for differences in library size between samples. The *black* line displays the theoretical variance under the Poisson model where the variance is equal to the mean. The *red* crosses show binned variance, where genes are grouped by mean level

biological variation in true expression levels between replicate libraries. Naïve estimation of the dispersion parameter (e.g. maximum likelihood) can be improved by sharing information across the entire dataset, a strategy that has been successful in the analysis of DE using microarray data. The simplest and most extreme example of this, in the RNA-seq context, is to assume that all genes have the same dispersion (Robinson and Smyth 2008). If this assumption is true, small pieces of information for every gene accumulate to give a very accurate estimate, even for small samples. A relaxation of this approach is a moderated estimate of dispersion, whereby each gene-wise dispersion estimate is smoothed towards the common dispersion (Robinson and Smyth 2007), providing a stabilization while avoiding the inherent difficulty in dispersion estimation from very small samples (Lu et al. 2005; Robinson and Smyth 2008). These methods are implemented in the edgeR package (Robinson et al. 2010a). Statistical testing in a multiple group framework can be carried out with conditional exact tests (Anders and Huber 2010; Robinson and Smyth 2008), therefore not relying on large-sample theory for its justification. Variations on the moderated estimation strategy have recently emerged, such as modeling dispersion as a

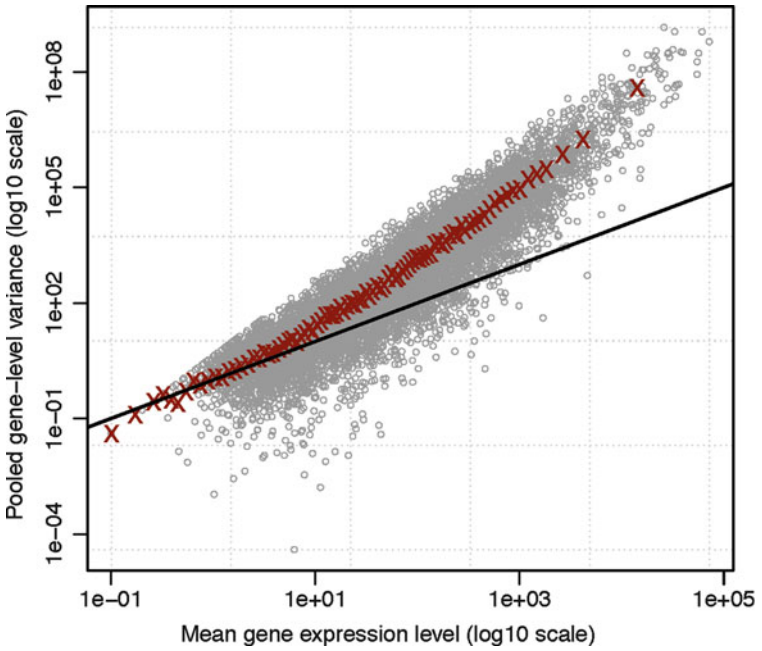

Fig. 10.6 Mean–variance plot for the Parikh et al. Dictyostelium dataset (Parikh et al. 2010). The variability in this biologically replicated RNA-seq dataset exhibits prominent extra-Poisson variability. The *grey* points show the mean and pooled variance for each gene. The *black* line displays the theoretical variance under the Poisson model where the variance is equal to the mean. The *red* crosses show binned variance, where genes are grouped by mean level

non-linear function of the mean (Anders and Huber 2010) or a two-stage Poisson model that only permits overdispersion for a selected subset of genes (Auer 2010). One slight disadvantage of the fixed dispersion–mean relationship is that no consideration is made for gene-specific dispersion, should larger sample sizes be available to provide information for it. Furthermore, a computationally intensive full Bayesian implementation of the NB-based DE analyses has been recently proposed (Hardcastle and Kelly 2010).

Unfortunately, the exact tests (with moderated dispersion estimates) are limited to simple experimental designs, such as two-group or multiple-group comparisons. For more complicated experimental designs, generalized linear models (GLMs) provide a logical extension for assessing DE from count data (McCullagh and Nelder 1989). GLM methods can handle time course experiments, paired samples, etc., while accounting for confounding variables, such as batch and lane effects. Any experimental design that can be expressed in terms of linear combinations of predictor variables can, in principle, be analyzed using GLM methods.

Thus far, three GLM approaches have been suggested for analyzing RNA-seq data – normal, Poisson, and NB models. All of these models can be made to fall under

the umbrella of GLMs, although the NB model is substantially more challenging to use, both mathematically and computationally. Normal models have been applied after applying some form of transformation to the count data (Cloonan et al. 2008; Langmead et al. 2010). Such approaches are computationally fast and convenient, but are unlikely to capture the true mean–variance relationship in RNA-seq data. Bullard et al. (2010) introduce GLMs to the RNA-seq context with a Poisson model. Their Poisson GLM approach is geared towards dealing with count data directly, but does not compensate for the extra-Poisson variation expected in most biologically interesting contexts.

Auer and Doerge (2010) propose a GLM with extra-Poisson variation, but offer no stabilization over all genes in the dataset. Srivastava and Chen (2010) also discuss GLM methods using Poisson and NB models, and introduce a position-level "generalized Poisson" model to account for overdispersion and purported underdispersion in RNA-seq data. Srivastava and Chen dismiss the NB model for RNA-seq data, but in their implementation of the NB model there is no sharing of information between genes. The edgeR package has shown how effective the NB model is at modeling RNA-seq data when information is shared between genes to improve inference. Methods recently introduced into the edgeR package (Robinson et al. 2010a) implement GLM methods that combine the NB model with stabilization over all genes in the dataset. This approach seems to be the first to allow appropriate modeling of the mean–variance relationship in the GLM context. All of the GLM approaches proposed thus far assess evidence for DE using a likelihood-ratio test for each gene.

10.7 Experimental Design for RNA-Seq

Soon after microarray platforms became a commonly used tool for molecular biologists, articles from statisticians appeared highlighting the need for appropriate experimental design (Churchill 2002; Yang and Speed 2002). The foundations of experimental design date back to Sir R.A. Fisher and rely on the fundamental concepts of replication, randomization, and blocking. Inadequate study design and the potential biases introduced from confounding factors cannot generally be corrected by clever data analysis. These design considerations are just as, if not more, important for sequencing-based studies (Auer and Doerge 2010). A common misconception of sequencing is that since the platform has low background and basically unlimited dynamic range, there is little need for replication. While it has been established that technical variation is indeed low, lack of biological replication prevents inferences regarding DE to be generalized to a sampled population. For example, if we compare expression levels from a single tumor sample to a single normal sample, any conclusions we make cannot possibly generalize to the *populations* of tumor and normal samples, since the analysis does not take into account the potentially large tumor-to-tumor or normal-to-normal variation. The conclusions apply only to the individuals under study.

In human studies, this biological variability is often enormous compared to genetically identical laboratory mice.

Another consideration is how to assign the samples to the available sequencing "lanes" within a flow cell (e.g. eight lanes within an Illumina Genome Analyzer). Most studies to date have assigned a single library in a single lane, with libraries spread over multiple flow cells (if the number of samples exceeds the number of lanes of a single flow cell). In designing an experiment, there are considerations to be made regarding how the sample-to-lane assignment is done. Experimenters should avoid confounded designs, such as assigning all replicates of one treatment condition (say, A) to one flow cell and all replicates of experimental condition B to another flow cell. Similarly, systematic differences may exist between lanes; random assignment of samples to lanes will reduce these effects. Cautionary tales in the analysis of proteomic data highlight the value in paying close attention to experimental design at the early stages of a study (Hu et al. 2005).

Another consideration for experimental design is the concept of blocking, whereby experimental units are grouped according to similarity, ensuring that samples are subjected to the same technical biases. One possibility is multiplexing, whereby each sample receives a barcode (incorporated into the adapter sequence). Auer and Doerge (2010) demonstrate the statistical justification of multiplexing to block on lane and batch. Their simulations illustrate that in the presence of lane and/or batch effects, the blocked design (i.e. multiplexed) exhibits improved sensitivity and specificity. Provided that the lane effects are present and the barcode itself does not introduce a bias (an assumption that has yet to be rigorously tested), such a design will improve statistical power. Designs that include blocking can be analyzed using the generalized linear model framework discussed above (cf. Auer and Doerge 2010; Bullard et al. 2010; Robinson et al. 2010a).

Another subtle design consideration is the "ideal" length of reads. Longer reads cost more, but will have better ability to map to the transcriptome, improved ability to discern exon–exon junctions, and more coverage to be able to partition expression by allele, or discover new genetic variants. However, since inferences of DE are made from read density, the statistical power increases with the total number of mapped reads, not by the total amount of sequence coverage. If a sequencing provider can guarantee a certain amount of total *sequence*, researchers may choose shorter reads to maximize the number of total reads. The gain in "mappability" from longer reads is generally small in comparison to the gain in statistical power from having more mapped reads.

Researchers may also wish to consider the use of paired-end reads, where both ends of a DNA fragment are sequenced. For total gene expression profiling, paired-end reads benefit users primarily from the increase in reads mapping to the transcriptome or genome, since reads mapping to multiple locations can often be reconciled by their matching pair. This gives unique access to structural variation (Maher et al. 2009), and additional information that helps deconvolve isoform expression (Trapnell et al. 2010). However, paired-end reads incur additional cost and do not necessarily increase the statistical power to detect DE.

10.8 Saturation Analysis

As mentioned above, statistical power is partially dictated by the total number of reads, which is ultimately decided by the total depth of sequencing that researchers can devote to their study. Therefore, it is important to recognize that DE results are dependent on this implicitly chosen depth. In general, deeper sequencing will allow increased detection of DE, but the amount of detected DE (at a given false discovery rate) should plateau at a sufficient depth. For many studies, it will be of interest to conduct a down-sampling analysis to determine the degree of saturation that has been reached, possibly revealing whether to dedicate more sequencing effort to the project. A down-sampling analysis works as follows: a proportion of the (mapped) reads are discarded and the data is re-analyzed with the same statistical procedure. This process is repeated for several levels of down-sampling (and multiple subsamples at a given level are taken). By fitting a functional form to the saturation curve, prediction of the total amount of DE can be made, as well as estimation of additional detections from a specified amount of further sequencing. An example of this is shown in Fig. 10.7, in a similar context of detecting differential methylation (see

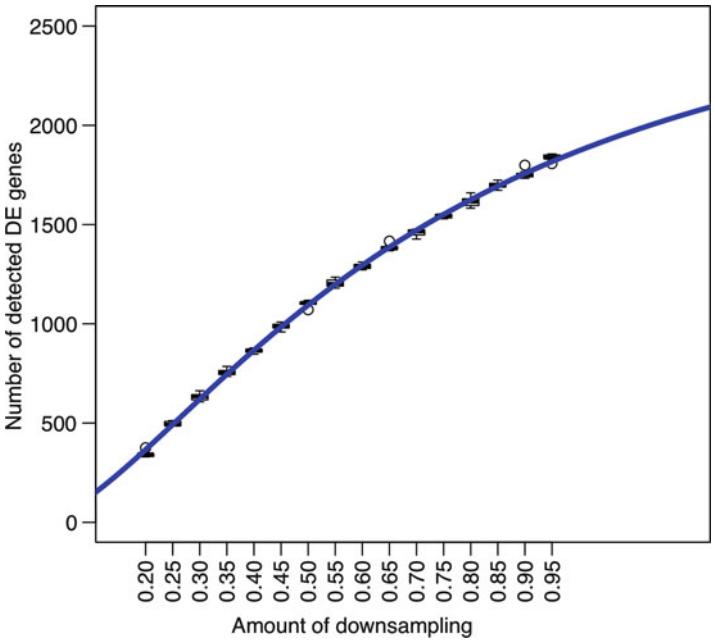

Fig. 10.7 Down-sampling analysis to determine saturation. The *X*-axis represents the level of downsampling where 1.00 is the full dataset. The *Y*-axis shows the number of DE genes that were determined at a given false discovery rate cutoff. The *blue* line represents a non-linear least squares fit according to a certain functional form. Estimates of the asymptote (i.e. estimated total number of DE genes) can be determined from the fit

Robinson et al. 2010b); similar saturation analyses are carried out in the assembly of transcripts using RNA-seq data. It should be noted that the extrapolated estimates of detected DE may be very sensitive to the chosen functional form.

10.9 Functional Category and Integrative Analyses

Often, biological insight into a system can be gained by placing a set of differentially expressed genes in the context of known functional information. A standard approach is so-called "functional category analysis," whereby each functional category is assessed for over-representation amongst differentially expressed genes. Such over-represented categories would be suggestive of disregulated pathways between the experimental conditions being studied. Similarly, biological insight into an experimental system can be gained by looking at the expression changes of predetermined *sets* of genes (e.g. Wu et al. 2010). Tools for mapping genes onto knowledge databases and inferring set-wise changes in gene expression are readily available (Dennis et al. 2003; Kanehisa and Goto 2000; Subramanian et al. 2005).

Careful analysis is required in order to apply the standard tools that have been widely used for microarray analysis to sequencing data. As already mentioned, RNA-seq is affected by biases not present in microarray data and these biases can have a substantial impact on functional category analysis (Young et al. 2010). Specifically, longer genes are expected to have higher counts compared to short genes at the same expression level, resulting in greater statistical power to detect changes. Thus, lists of differentially expressed genes are ultimately biased toward genes with high counts, which tend to be longer and more highly expressed (Oshlack and Wakefield 2009). Modifications to standard analyses have been suggested to account for this, including a DE t-statistic that has been divided by the square root of gene length (Bullard et al. 2010), or a sampling-based approach that generates a length-adjusted null distribution (Young et al. 2010). It is worth noting that gene length bias can affect all analyses that look for patterns of groups of genes. Furthermore, other biases in RNA-seq data, such as GC content bias and as yet unidentified biases, will also affect downstream systems biology approaches.

RNA-seq may be considered as the downstream output of an experimental system. Techniques for interrogating upstream gene regulatory mechanisms, such as transcription factor binding, histone modifications, and DNA methylation, are now becoming widely used through the use of immunoprecipitation-based techniques. Thus, there is wide scope and interest in integrating expression data from RNA-seq to the vast array of available regulatory, genetic, and epigenetic datasets (Hawkins et al. 2010). A few reports of these "integrative" analyses have emerged recently (Lister et al. 2009; Ouyang et al. 2009; Raha et al. 2010). For example, Lister and coauthors highlighted a striking difference in the correlations of RNA-seq expression with CG and non-CG methylation levels in gene bodies (Lister et al. 2009). Similarly, combinations of sequencing-based datasets are beginning to provide insights into the mono-allelic associations between expression, histone modifications, and DNA

methylation (Harris et al. 2010). In other work, RNA-seq has been used in conjunction with genotyping data to identify genetic loci responsible for variation in gene expression between individuals (eQTLs) (Montgomery et al. 2010; Pickrell et al. 2010). The integration of expression data with transcription factor binding, RNA interference, histone modification, and DNA methylation information has the potential to provide greater understanding of a variety of regulatory mechanisms.

10.10 Going Beyond Differential Expression with RNA-Seq

While our focus in this chapter has been on the typical use-case of RNA-seq (i.e. discovering changes in gene expression), much of the excitement around sequencing-based approaches lies in the ability to access biological features that were not readily available using previous technologies. RNA-seq data sheds light on many complexities within the landscape of gene expression, such as splicing events, differential isoforms between experimental conditions, and the presence of SNPs, insertions, and deletions. Harnessing this information can be used to further understand biological phenomena such as alternative splicing, allele-specific regulation, and RNA editing, as well as mechanisms that contribute to aberrations observed in disease.

One active area of research for RNA-seq studies is novel-transcript identification and the characterization of alternative splicing. Early reports suggest that approximately 95% of all multi-exon human genes exhibit alternative forms (Pan et al. 2008; Sultan et al. 2008; Wang et al. 2008), partially accounting for the vast human protein diversity. Several tools are available to estimate transcript abundance (Jiang and Wong 2009; Li et al. 2010), detect alternative splice forms based on existing annotation (Griffith et al. 2010; Trapnell et al. 2010; Wang et al. 2010) or discover novel transcripts independent of annotation (Robertson et al. 2010; Trapnell et al. 2010). This is an active area of research and, at time of writing, many new methods and tools are being proposed. Interestingly, aspects of alternative splicing can be tackled by representing the mapped read data as counts at exons and exon–exon junctions and making comparisons across experimental conditions using statistical models similar to those described above. For example, Blekhman et al. used a Poisson model with random effects to highlight differential isoform usage between primates between gender within species (Blekhman et al. 2010). Similarly, exon-level counts were used to find changes in isoform expression between mouse subspecies with standard analysis-of-variance (Harr and Turner 2010).

RNA-seq data also offer the potential to identify structural aberrations that create fusion transcripts. Using a hybrid short- and long-read approach, Maher et al. (2009) provided a proof-of-principle experiment, which recaptured known gene fusions from chronic myeloid leukemia and prostate cancer, while identifying and validating several new chimeric transcripts.

RNA-seq data can be used to detect SNPs, albeit with an ability that is biased towards highly expressed genes. Despite some potential biases in mapping, RNA-seq can detect SNPs in transcripts that are different from the reference base or

heterozygous in a sample (Degner et al. 2009). Heterozygously transcribed SNPs allow researchers to partition observed transcription by allele, a phenomenon that occurs with tissue specificity and in approximately 10–20% of RNAs (Zhang et al. 2009) and can be associated with monoallelic regulation (Harris et al. 2010). Similarly, by further increasing the complexity of the signal (e.g. sampling pools of individuals sampled from a population), there is the potential to search for allele-specific antisense transcription and alternative splicing variants (Babak et al. 2010).

A further use of SNP detection is to study the phenomenon of RNA editing, whereby primary transcripts undergo individual base substitutions that result in a mature transcript with sequence different from the DNA sequence. The landscape of such transitions has been explored recently in mitochondrial RNA, revealing hundreds of C-to-U conversions (Picardi et al. 2010). There is also potential to modify library preparation protocols to discover and characterize small RNAs and non-coding RNAs, further investigating the complex process of transcription.

10.11 Conclusions

RNA-seq is now a mainstream tool for the analysis of transcriptomes and is well on its way to replacing DNA microarrays as the platform of choice. The data from RNA-seq is rich in information but complex to analyze and sensitive to technical biases. The focus in this chapter was perhaps the most straightforward use-case: searching for differentially expressed genes. We have highlighted the issues surrounding mapping short reads to a reference transcriptome, summarizing mapped reads into a metric of expression level, normalization for depth and composition, statistical models to assess changes in count data, experimental design for RNA-seq data, and the impact of biases on downstream analyses. There is wide scope for integration of RNA-seq data with other types of high-throughput data, such as genetic and epigenetic variation.

References

't Hoen PA, Ariyurek Y, Thygesen HH, et al. (2008) Deep sequencing-based expression analysis shows major advances in robustness, resolution and inter-lab portability over five microarray platforms. Nucleic Acids Res 36:e141

Ameur A, Wetterbom A, Feuk L, et al. (2010) Global and unbiased detection of splice junctions from RNA-seq data. Genome Biol 11:R34

Anders S and Huber W (2010) Differential expression analysis for sequence count data. Genome Biol 11:R106

Auer PL (2010) Statistical Design And Analysis Of Next-Generation Sequencing Data. Doctor of Philosophy, Purdue University

Auer PL and Doerge RW (2010) Statistical design and analysis of RNA sequencing data. Genetics 185:405–16

Babak T, Garrett-Engele P, Armour CD, et al. (2010) Genetic validation of whole-transcriptome sequencing for mapping expression affected by cis-regulatory variation. BMC Genomics 11:473

Binder H, Kirsten T, Loeffler M, et al. (2004) Sensitivity of Microarray Oligonucleotide Probes: Variability and Effect of Base Composition. The Journal of Physical Chemistry B 108:18003–14

Blekhman R, Marioni JC, Zumbo P, et al. (2010) Sex-specific and lineage-specific alternative splicing in primates. Genome Res 20:180–9

Bock C, Tomazou EM, Brinkman AB, et al. (2010) Quantitative comparison of genome-wide DNA methylation mapping technologies. Nat Biotechnol 28:1106–14

Bradford JR, Hey Y, Yates T, et al. (2010) A comparison of massively parallel nucleotide sequencing with oligonucleotide microarrays for global transcription profiling. BMC Genomics 11:282

Bullard JH, Purdom E, Hansen KD, et al. (2010) Evaluation of statistical methods for normalization and differential expression in mRNA-Seq experiments. BMC Bioinformatics 11:94

Carvalho PC, Hewel J, Barbosa VC, et al. (2008) Identifying differences in protein expression levels by spectral counting and feature selection. Genet Mol Res 7:342–56

Churchill GA (2002) Fundamentals of experimental design for cDNA microarrays. Nat Genet 32 Suppl:490–5

Cloonan N, Forrest AR, Kolle G, et al. (2008) Stem cell transcriptome profiling via massive-scale mRNA sequencing. Nat Methods 5:613–9

De Bona F, Ossowski S, Schneeberger K, et al. (2008) Optimal spliced alignments of short sequence reads. Bioinformatics 24:i174–80

Degner JF, Marioni JC, Pai AA, et al. (2009) Effect of read-mapping biases on detecting allele-specific expression from RNA-sequencing data. Bioinformatics 25:3207–12

Dennis G, Jr., Sherman BT, Hosack DA, et al. (2003) DAVID: Database for Annotation, Visualization, and Integrated Discovery. Genome Biol 4:P3

Ferragina P and Manzini G (2000) Opportunistic data structures with applications. Annu Symp Found Comput Sci Proc 2000:390–398

Flicek P and Birney E (2009) Sense from sequence reads: methods for alignment and assembly. Nat Methods 6:S6–S12

Fu X, Fu N, Guo S, et al. (2009) Estimating accuracy of RNA-Seq and microarrays with proteomics. BMC Genomics 10:161

Griffith M, Griffith OL, Mwenifumbo J, et al. (2010) Alternative expression analysis by RNA sequencing. Nat Methods 7:843–7

Hansen KD, Brenner SE and Dudoit S (2010) Biases in Illumina transcriptome sequencing caused by random hexamer priming. Nucleic Acids Res 38:e131

Hardcastle TJ and Kelly KA (2010) baySeq: empirical Bayesian methods for identifying differential expression in sequence count data. BMC Bioinformatics 11:422

Harr B and Turner LM (2010) Genome-wide analysis of alternative splicing evolution among Mus subspecies. Mol Ecol 19 Suppl 1:228–39

Harris RA, Wang T, Coarfa C, et al. (2010) Comparison of sequencing-based methods to profile DNA methylation and identification of monoallelic epigenetic modifications. Nat Biotechnol 28:1097–1105

Hawkins RD, Hon GC and Ren B (2010) Next-generation genomics: an integrative approach. Nat Rev Genet 11:476–86

Hu J, Coombes KR, Morris JS, et al. (2005) The importance of experimental design in proteomic mass spectrometry experiments: some cautionary tales. Brief Funct Genomic Proteomic 3:322–31

Jiang H and Wong WH (2009) Statistical inferences for isoform expression in RNA-Seq. Bioinformatics 25:1026–32

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

Langmead B, Hansen KD and Leek JT (2010) Cloud-scale RNA-sequencing differential expression analysis with Myrna. Genome Biol 11:R83

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

Levin JZ, Yassour M, Adiconis X, et al. (2010) Comprehensive comparative analysis of strand-specific RNA sequencing methods. Nat Methods 7:709–15
Li B, Ruotti V, Stewart RM, et al. (2010) RNA-Seq gene expression estimation with read mapping uncertainty. Bioinformatics 26:493–500
Li H and Durbin R (2009) Fast and accurate short read alignment with Burrows-Wheeler transform. Bioinformatics 25:1754–60
Li H, Ruan J and Durbin R (2008) Mapping short DNA sequencing reads and calling variants using mapping quality scores. Genome Res 18:1851–8
Linsen SE, de Wit E, Janssens G, et al. (2009) Limitations and possibilities of small RNA digital gene expression profiling. Nat Methods 6:474–6
Lister R, Pelizzola M, Dowen RH, et al. (2009) Human DNA methylomes at base resolution show widespread epigenomic differences. Nature 462:315–22
Liu S, Lin L, Jiang P, et al. (2011) A comparison of RNA-Seq and high-density exon array for detecting differential gene expression between closely related species. Nucleic Acids Res 39:578–88
Lu J, Tomfohr JK and Kepler TB (2005) Identifying differential expression in multiple SAGE libraries: an overdispersed log-linear model approach. BMC Bioinformatics 6:165
Maher CA, Kumar-Sinha C, Cao X, et al. (2009) Transcriptome sequencing to detect gene fusions in cancer. Nature 458:97–101
Marioni JC, Mason CE, Mane SM, et al. (2008) RNA-seq: An assessment of technical reproducibility and comparison with gene expression arrays. Genome Res 18:1509–17
McCullagh P and Nelder JA (1989) Generalized linear models, 2nd. Chapman and Hall, London ; New York
Montgomery SB, Sammeth M, Gutierrez-Arcelus M, et al. (2010) Transcriptome genetics using second generation sequencing in a Caucasian population. Nature 464:773–7
Mortazavi A, Williams BA, McCue K, et al. (2008) Mapping and quantifying mammalian transcriptomes by RNA-Seq. Nat Methods 5:621–8
Naef F and Magnasco MO (2003) Solving the riddle of the bright mismatches: labeling and effective binding in oligonucleotide arrays. Phys Rev E Stat Nonlin Soft Matter Phys 68:011906
NCBI (2011) NCBI – Entrez Genome. http://www.ncbi.nlm.nih.gov/sites/genome Accessed October 14
Oshlack A and Wakefield MJ (2009) Transcript length bias in RNA-seq data confounds systems biology. Biol Direct 4:14
Ouyang Z, Zhou Q and Wong WH (2009) ChIP-Seq of transcription factors predicts absolute and differential gene expression in embryonic stem cells. Proc Natl Acad Sci USA 106:21521–6
Pan Q, Shai O, Lee LJ, et al. (2008) Deep surveying of alternative splicing complexity in the human transcriptome by high-throughput sequencing. Nat Genet 40:1413–5
Parikh A, Miranda ER, Katoh-Kurasawa M, et al. (2010) Conserved developmental transcriptomes in evolutionarily divergent species. Genome Biol 11:R35
Picardi E, Horner DS, Chiara M, et al. (2010) Large-scale detection and analysis of RNA editing in grape mtDNA by RNA deep-sequencing. Nucleic Acids Res 38:4755–67
Pickrell JK, Marioni JC, Pai AA, et al. (2010) Understanding mechanisms underlying human gene expression variation with RNA sequencing. Nature 464:768–72
Quackenbush J (2002) Microarray data normalization and transformation. Nat Genet 32 Suppl: 496–501
Quail MA, Kozarewa I, Smith F, et al. (2008) A large genome center's improvements to the Illumina sequencing system. Nat Methods 5:1005–10
Raha D, Wang Z, Moqtaderi Z, et al. (2010) Close association of RNA polymerase II and many transcription factors with Pol III genes. Proc Natl Acad Sci USA 107:3639–44
Robertson G, Schein J, Chiu R, et al. (2010) De novo assembly and analysis of RNA-seq data. Nat Methods 7:909–12
Robinson MD, McCarthy DJ and Smyth GK (2010) edgeR: a Bioconductor package for differential expression analysis of digital gene expression data. Bioinformatics 26:139–40

Robinson MD and Oshlack A (2010) A scaling normalization method for differential expression analysis of RNA-seq data. Genome Biol 11:R25
Robinson MD and Smyth GK (2007) Moderated statistical tests for assessing differences in tag abundance. Bioinformatics 23:2881–7
Robinson MD and Smyth GK (2008) Small-sample estimation of negative binomial dispersion, with applications to SAGE data. Biostatistics 9:321–32
Robinson MD, Stirzaker C, Statham AL, et al. (2010) Evaluation of affinity-based genome-wide DNA methylation data: effects of CpG density, amplification bias, and copy number variation. Genome Res 20:1719–29
Schadt EE, Linderman MD, Sorenson J, et al. (2010) Computational solutions to large-scale data management and analysis. Nat Rev Genet 11:647–57
Simpson JT, Wong K, Jackman SD, et al. (2009) ABySS: a parallel assembler for short read sequence data. Genome Res 19:1117–23
Srivastava S and Chen L (2010) A two-parameter generalized Poisson model to improve the analysis of RNA-seq data. Nucleic Acids Res 38:e170
Subramanian A, Tamayo P, Mootha VK, et al. (2005) Gene set enrichment analysis: a knowledge-based approach for interpreting genome-wide expression profiles. Proc Natl Acad Sci USA 102:15545–50
Sultan M, Schulz MH, Richard H, et al. (2008) A global view of gene activity and alternative splicing by deep sequencing of the human transcriptome. Science 321:956–60
Taub M and Speed TP (2010) Methods for allocating ambiguous short-reads. Communications in information and systems 10:69–82
Trapnell C, Pachter L and Salzberg SL (2009) TopHat: discovering splice junctions with RNA-Seq. Bioinformatics 25:1105–11
Trapnell C, Williams BA, Pertea G, et al. (2010) Transcript assembly and quantification by RNA-Seq reveals unannotated transcripts and isoform switching during cell differentiation. Nature biotechnology 28:511–515
Wang ET, Sandberg R, Luo S, et al. (2008) Alternative isoform regulation in human tissue transcriptomes. Nature 456:470–6
Wang L, Xi Y, Yu J, et al. (2010) A statistical method for the detection of alternative splicing using RNA-seq. PLoS One 5:e8529
Wang Z, Gerstein M and Snyder M (2009) RNA-Seq: a revolutionary tool for transcriptomics. Nat Rev Genet 10:57–63
White JR, Nagarajan N and Pop M (2009) Statistical methods for detecting differentially abundant features in clinical metagenomic samples. PLoS Comput Biol 5:e1000352
Wu D, Lim E, Vaillant F, et al. (2010) ROAST: rotation gene set tests for complex microarray experiments. Bioinformatics 26:2176–82
Wu Z and Irizarry RA (2005) Stochastic models inspired by hybridization theory for short oligonucleotide arrays. J Comput Biol 12:882–93
Yang YH and Speed T (2002) Design issues for cDNA microarray experiments. Nat Rev Genet 3:579–88
Young MD, Wakefield MJ, Smyth GK, et al. (2010) Gene ontology analysis for RNA-seq: accounting for selection bias. Genome Biol 11:R14
Zerbino DR and Birney E (2008) Velvet: algorithms for de novo short read assembly using de Bruijn graphs. Genome Res 18:821–9
Zhang K, Li JB, Gao Y, et al. (2009) Digital RNA allelotyping reveals tissue-specific and allele-specific gene expression in human. Nat Methods 6:613–8

Chapter 11
MicroRNA Expression Profiling and Discovery

Michael Hackenberg

Abstract MicroRNAs are key players of numerous fundamental pathways whose dysregulation is involved in the development of many diseases. With the advent of High-Throughput Sequencing (HTS) technologies, the expression levels of known microRNAs can now be fast and inexpensively profiled. In this chapter, we review the basic steps of HTS small RNA data analysis including the preprocessing of the reads, expression profiling, isomiR analysis, differential expression, prediction of novel microRNAs, and downstream analyses. We will discuss eight different applications, including web server tools and software packages, developed for the analysis of microRNA data. We will specially emphasize the comparison of the different approaches and their putative effects on the results.

11.1 Introduction

MicroRNAs were discovered almost 20 years ago while studying the lin-14 gene in *Caenorhabditis elegans* (Lee et al. 1993). The abundance of the protein encoded by this gene was found to depend on a short non-coding RNA transcribed from another gene, the lin-4. The 3′ UTR sequence of lin-14 has partial complementary regions to the mature, 22 nt-long RNA sequence of lin-4 causing translation inhibition of lin-14 due to antisense RNA–RNA interaction. It took almost a decade before the second microRNA, let-7, was described in *C. elegans* (Reinhart et al. 2000). This microRNA was furthermore found to be conserved in a broad range of species, including *Drosophila melanogaster* and human, suggesting that post-transcriptional regulation of gene expression by means of short non-coding RNAs might be a

M. Hackenberg (✉)
Computational Genomics and Bioinformatics Group, Genetics Department, University of Granada, Campus de Fuentenueva s/n, 18071 Granada, Spain
e-mail: mlhack@gmail.com

N. Rodríguez-Ezpeleta et al. (eds.), *Bioinformatics for High Throughput Sequencing*,
DOI 10.1007/978-1-4614-0782-9_11,

widespread phenomenon and by no means nematode specific (Pasquinelli et al. 2000). Over the last years, microRNAs have been found in virtually all eukaryotic organisms except fungi and some plants, and the number of known microRNAs has grown strongly. The most recent version of miRBase (Release 16) (Kozomara and Griffiths-Jones 2011) contains up to 15,172 microRNAs. Many of them are phylogenetically conserved, suggesting a role in the regulation of important and very basic cellular functions such as apoptosis and cell differentiation (Wienholds and Plasterk 2005). In addition, the dysregulation of microRNAs is involved in the appearance of cancer and other diseases (Alvarez-Garcia and Miska 2005; Esquela-Kerscher and Slack 2006). Given the importance of microRNAs and other small ncRNAs in the regulation of gene expression, several important goals exist in the analysis of microRNAs: (1) profiling the expression of known microRNAs, (2) detecting differentially expressed microRNAs, (3) predicting and detecting novel microRNA genes, (4) detecting the microRNA target genes, and (5) developing integrated methods to infer microRNA regulatory networks. Until recently, the detection of microRNA expression levels has been a daunting task because existing techniques as northern blotting (Lagos-Quintana et al. 2001) or cloning/sequencing approaches (Cummins et al. 2006; Landgraf et al. 2007) are slow and expensive, and microarrays are limited to predefined features. With the advent of HTS technologies this scenario has drastically changed. First, the expression levels of known microRNAs and other small ncRNA can be inexpensively profiled within a given sample, and second, novel microRNAs can be detected in a more reliable way (Bar et al. 2008; Creighton et al. 2009; Morin et al. 2008). Normally, both the sequence composition and the secondary structure are used to predict novel microRNAs [see Lim et al. (2003) or Li et al. (2010) for a review]. The existence of a hairpin fold-back structure formed by the pre-microRNA plays a fundamental role in the detection of microRNA genes. It has been reported however that the human genome contains approximately 11 million sequences that can form a hairpin secondary structure (Bentwich et al. 2005). This fact suggests that prediction algorithms will suffer from a high false positive rate and therefore cross-species comparisons are usually applied in order to filter for conserved hairpin structures or to analyze the conservation profile (Berezikov et al. 2005). This implies accepting the drawback that probably a large number of nonconserved, species-specific microRNA genes will not be detected (Bentwich et al. 2005). The arrival of HTS technologies added new layers of information to the prediction that were not available before. First, the knowledge of whether a given sequence is expressed or not is given, and second, the traces left by Dicer, a endoribonuclease that cleaves the pre-microRNA to double-stranded mature microRNA can be assessed. This new information together with previously developed, machine-learning based methods helped to improve the detection of microRNAs enormously (Li et al. 2010).

In this chapter, I will review under a bioinformatics viewpoint the most common steps necessary to convert the information of HTS small RNA data contained within the FASTQ files into biological knowledge. Special emphasis will be put on the profiling of microRNA expression, addressing the most common problems like adapter

removal, quality filtering, isomiR detection, and the multiple-mapping problem. I will also briefly review the different approaches to predict novel microRNAs and the downstream analyses that are available so far. I will conclude giving a brief outlook on the field mentioning some possible developments for the future.

11.2 Profiling the Expression of Known MicroRNAs

A notable number of tools to analyze HTS small RNA data have been developed. In this chapter, I will focus on eight of them (see Table 11.1 for a summary): DSAP (Huang et al. 2010), E-miR (Buermans et al. 2010), miRanalyzer (Hackenberg et al. 2009), miRExpress (Wang et al. 2009), miRNAkey (Ronen et al. 2010), mirTools (Zhu et al. 2010), SeqBuster (Pantano et al. 2010), and the UEA sRNA toolkit (Moxon et al. 2008). Among the first software packages to analyze HTS microRNA data has been miRDeep (Friedlander et al. 2008), which is used now by mirTools for the prediction of new microRNAs. Other important protocols, but not software applications, have been developed and described elsewhere (Creighton et al. 2009; Morin et al. 2008).

The tools share most analysis steps, but differ in the exact order of the work flow or in the concrete approach for a given task. I will therefore provide a "task centered" rather than a "program centered" review.

11.2.1 Input Formats and Scope

Although microRNA sequencing protocols are available for all three major sequencing platforms, Illumina, 454 and AB SOLiD (Shendure and Ji 2008), most analysis tools do not accept color-space sequences as used by SOLiD to encode their di-base sequencing chemistry. The main advantage of color-space encoding is that by means of the alignments to the reference, sequencing errors (one mismatch) and sequence variants (two mismatches) can be distinguished. However, this implies that all alignments must be performed in color-space, i.e. the reference sequence must be converted to color-space as well. Note that color-space should not be converted to nucleotide sequences before aligning. This is because from the position of a sequencing error to the end of the read, the conversion would be incorrect. The drawbacks of using incorrectly converted reads are obvious: (1) many of them will not align to the reference and the information carried by them is lost as well, and (2) the reads might map at arbitrary positions, causing therefore a false "signal". Though, the effect of the conversion before aligning is not quantified it is not advisable to proceed like this. Currently only the new version of miRanalyzer accepts color-space input sequences as it uses the Bowtie aligner (Langmead et al. 2009).

Table 11.1 Brief comparison of the most important programs

	DSAP	E-miR	miRanalyzer	miRExpress	miRNAkey	mirTools	SeqBuster	UEA sRNA toolkit
Availability	http://dsap.cgu.edu.tw/dsap.html	http://www.lgtc.nl/EmiR	http://bioinfo2.ugr.es/miRanalyzer/miRanalyzer.php	http://mirexpress.mbc.nctu.edu.tw/	http://ibis.tau.ac.il/miRNAkey/	http://centre.bioinformatics.zj.cn/mirtools/	http://estivill_lab.crg.es/seqbuster/	http://srna-tools.cmp.uea.ac.uk/animal/cgi-bin/srna-tools.cgi?rm=input_form&tool=mircat
Implementation	Web server	Software package	Web server and local program	Local pipeline	Local program	Web server	Web server and local program	Web server
Main features	Differential expression, Rfam filter, graphical output, cross-species comparison, optimized sequence alignment for isomiRs, 142 species, remove poly-A,C,G,T	Differential expression, isomiR analysis, annotations via Ensembl Perl API,	Color-space support, prediction of novel microRNAs for plants and animals, differential expression including novel microRNAs, standalone version	Detection of novel microRNAs based on homology	Graphical user interface, graphical output, additional information like multiple mapping levels and post-clipping read lengths	Detection of new microRNAs (miRDeep), ncRNA, coding genes, Rfam and RepeatMasker libraries, graphical and text output, limited to 10 M	Differential expression, isomiR analysis, graphical user interface and output, prefiltering of low complexity reads	Expression profiling of known microRNAs (miRProf), detection of new microRNAs (miRCat), versions for plant and animal
Input	Read/count	FASTQ, SCARF, read/count	Read/count, fasta	FASTQ, read/count	FASTQ, fasta	Fasta	Fasta, illumina "seq" files	Fasta

Differential expression	Yes	Yes (Limma)	Yes (DESeq)	No	Yes (chi-square)	Yes (Bayes)	Yes (Z-test)	Yes
Novel microRNAs	NA	NA	Animal, plant	Homology based	NA	Animal, plant	NA	Animal, plant
Quality	Not used	Not used	In preprocessing with a perl script	Not used	Not used	In preprocessing with a perl script	Not used	Not used
Adapter handling	Remove: 5 nt, 0 MM, Supermatcher	Remove: 8 nt, 1 MM	Seed alignment	Remove: 70% identity	Remove: NA	Remove: NA	Remove: 10 nt, 3 MM,	Remove adapter (preprocessing tool)
Aligner	BLAST	Eland	Bowtie	Smith-Waterman	Burrows-Wheeler Aligner	SOAP, mega BLAST	Mega BLAST	PatMaN
Color space	No	No	Yes	No	No	No	No	No
Additional libraries	Rfam	No	Rfam, mRNA, (RepBase in old version)	No	No	Rfam, mRNA, RepBase	No	No

11.2.2 Preprocessing

The preprocessing of the reads is an important step in the analysis of ncRNA data that compasses both mere technical but also biological aspects. For example, the original FASTQ files are usually too big to be sent over the web and therefore the web server tools use either tab-separated read/count or fasta files in order to reduce the input file size. However, some steps like the adapter handling, the read count threshold and the consideration of quality values might drastically influence the outcome of the analysis.

11.2.2.1 Adapter Handling

The sequences of short ncRNA libraries have typically lengths between 17–35 bp (like in the case of Illumina "Small RNA Discovery and Analysis"), but the range might vary between the different protocols. This implies that very often the read length will be longer than the sequenced molecule. In such cases, the adapter will also be sequenced appearing at some arbitrary position towards the 3′ end of the read. The adapter sequence will cause mismatches in the alignments to the reference and must therefore be taken into account. Currently, two different approaches have been proposed: (1) the adapter sequence is detected and removed generating 3′ trimmed, clean reads, and (2) the adapter-free part of the read is aligned first (seed alignment), extending the alignment afterwards. Among the tools discussed here, with the exception of miRanalyzer, all implement the removal of the adapter sequence. The detection of the adapter sequence is not trivial and basically every tool has implemented its own method or parameter set; or even worse, do not mention in detail how they perform this step. It is known that the frequency of sequencing errors increases towards the 3′ end of the reads in Illumina sequencing, which implies more sequencing errors in the adapter sequence than in the 5′ end of the read. Furthermore, the adapter can appear at a broad range of positions within the read or even only partially at the 3′ end. Given this scenario, mismatches must be considered and a minimum length must be established. A tradeoff between specificity and sensitivity of adapter detection must be found. If very stringent alignment parameters are used, many adapter sequences will be missed and the reads will very likely fail to align, thus losing the information carried by them. If to lax parameters are applied (many mismatches and short-detected adapter sequence), non-adapter sequences will be erroneously identified as adapters.

For example, SeqBuster allows up to three mismatches (corresponding roughly to 85% sequence identity) and no gaps in the alignment. The first ten bases of the adapter are searched for between position 15 and the 3′ end of the read applying a modified Needleman-Wunsch algorithm. These parameters can be manipulated in the standalone version of the tool. In the DSAP web server this step is less stringent and only

the first 5 nt of the adapter need to be detected (probably without mismatches) using the Supermatcher algorithm from the EMBOSS package. In the E-miR paper, the authors report that "*matches to the first 4–6 nt of the adapter sequence may occur by chance within the RNA insert and cause aberrant truncation of the sequence.*" A first consequence of this observation is that the use of a 26-cycle protocol in Illumina might be problematic as many adapters would be only partially sequenced causing either a high rate of missed adapters and/or a high number of erroneously trimmed reads. The E-miR authors recommend the detection of at least 8 nt of the adapter sequence allowing 1 MM. A comparison to SeqBuster yielded a 2–3 times increase in speed and a more accurate removing of the adapter sequence without, however, mentioning how the comparison was done exactly. In summary, rather huge differences among the different approaches exist and it would be important to quantify the impact by means of a systematic comparison between the methods.

On the other hand, miRanalyzer implements a method very similar to the one used by Friedlander et al. in the miRDeep package (Friedlander et al. 2008). Following this approach, a subsequence of the read starting at the 5′ end called "seed" is mapped first to the reference. In the miRanalyzer web server, the first 17 nt of the read are used as seed (–l option in Bowtie). Among all possible best alignments (those with less mismatches), the longest one(s) maintaining the number of mismatches observed in the seed region are retained. In theory, this approach allows to align all reads with adapter sequences located after the seed region. The advantage is that the adapter sequence does not need to be detected explicitly, thus avoiding problems derivated from missed adapters or erroneously removed regions. A possible disadvantage is that single-nucleotide 3′ extensions cannot be detected (Morin et al. 2008).

11.2.2.2 Quality Values

Probably, one of the most disregarded aspects in the analysis of ncRNA data is the consideration of the base call quality values. Each sequenced base has assigned a Phred quality score that indicates the probability of a sequencing error. While it is clear that for SNV (Single Nucleotide Variants) calling and methylation profiling, the quality of the base calls is crucial, its impact on the analysis of ncRNA data is less clear. Currently, none of the programs discussed here uses the quality values during the alignment process. Just two programs, miRanalyzer and miRtools provide scripts allowing the filtering of low-quality reads in the preprocessing step when converting FASTQ format to read/count or fasta format. As mentioned before, the potential of the quality values has not been systematically assessed and probably for the expression profiling it is less important. However, for the detection of RNA editing and probably also for the detection of isomiRs it might be important to take the quality values of the individual bases into account.

11.2.2.3 Read Length

Some programs impose minimum and maximum read lengths. In the UEA sRNA toolkit, the length range can be set within the "Filter" tool during the preprocessing step, while in the mirTools it is a parameter of the proper tool (between 16 and 32 nt). miRanalyzer sets a minimum length of 17 nt and maximum length of 26 nt. These parameters can be manipulated in the standalone version of the tool. miRanalyzer does not perform an explicit adapter removal step and therefore a seed alignment is performed using this minimum length as seed length (–l option in Bowtie). For the other tools, the reads will be trimmed automatically to the size of the sequenced molecule due to the adapter removal. Some programs apply length thresholds after this step like E-miR (removing all reads shorter than 15 nt and longer than 32 nt) or contain implicit length thresholds like SeqBuster. This tool checks for adapter sequences from the position 15 on, and therefore all adapters incorporated earlier cannot be removed. Those reads will most likely not align and be therefore lost.

11.2.2.4 Number of Unique Reads

A common step is the generation of unique sequence reads, i.e. grouping together all reads with the same nucleotide sequence counting the number of copies. Some programs perform this step during the preprocessing providing data in read/count format, others accept redundant fasta files doing the grouping and counting internally (UEA sRNA toolkit). Reads with low copy numbers have a higher probability to be caused by sequencing errors and therefore some studies have used minimum count thresholds. For example, Morin et al. remove all reads with less than four counts, while Creighton et al. regard all reads with less than ten as putative sequencing errors. SeqBuster, miRExpress, UEA toolkit, miRNAkey, miRtools, and E-miR do not allow the automatic filtering of low copy reads although the user could of course manipulate the input by manually removing the reads with low counts. DSAP and miRanalyzer provide scripts for the preprocessing of the data including the filtering of reads with low counts. However, in both scripts, the threshold is applied before adapter removal and therefore the number of unique reads will be much higher as sequencing errors in the adapter will lead to "new" unique read sequences. Therefore, the correct way would be to apply this threshold after the adapter trimming or after the alignment in case of miRanalzyer (no adapter detection).

In summary, just very much as in the other preprocessing steps, so far no perfect or advisable threshold for the minimum count has been reported. It is clear that when using all reads including those with single or extremely low counts, the analysis will be more sensible (more microRNAs can be detected); undoubtfully, however, it will be also less specific (higher number of false positives). This will be especially true for the prediction of novel microRNAs where single count reads might constitute an important source of noise. This is an issue that needs to be tackled in the future.

11.2.3 Profiling Expression of Known MicroRNAs

The first step in the profiling of microRNA expression is the alignment of the reads to a microRNA reference library. The absolute expression value of a given microRNA is the read count sum of all reads that mapped to the reference sequence of this microRNA. The values can be normalized dividing the individual absolute counts by the read count sum of all reads that mapped to any of the reference sequences of the library. Right now, most tools use unique sequence reads that have already assigned the read count (copy number of each unique read) for the alignment. In the future, probably some local standalone applications or software packages might take advantage of the quality values contained within the FASTQ files, aligning first all reads, grouping them to unique reads afterwards. The most used sequence libraries are the known microRNAs from miRBase (Kozomara and Griffiths-Jones 2011). In general, the mature microRNA sequences are used to detect the expression levels although when detecting isomiRs, the pre-microRNA sequences must be also taken into account (see below). Additionally, miRanalyzer includes the profiling of all theoretically possible mature* sequences which allows the detection of previously unobserved mature* microRNAs.

11.2.3.1 Aligners and Parameters

The crucial step in profiling the expression of known microRNAs is the alignment of the reads to the reference libraries. A huge number of algorithms have been proposed over the last two decades for this task, and recently methods have been developed to address the specific needs of HTS experiments yielding hundreds of millions of short reads. Therefore, it is not surprising that the tools discussed here are based on different alignment algorithms (see also Table 11.1): MegaBLAST or BLAST (Zhang et al. 2000) (DSAP, miRtools, SeqBuster, Morin protocol), Smith-Waterman (miRExpress, Creighton protocol), PatMaN (Prufer et al. 2008) (UEA sRNA toolkit), the commercial Illumina aligner Eland (E-miR), Bowtie (Langmead et al. 2009) (miRanalyzer), and Burrows-Wheeler Aligner (Li and Durbin 2009) (miRNAkey).

Moreover, the tools do not coincide in the way mismatches are considered. The DSAP web server and the Morin protocol do not allow mismatches to the reference sequence. Creighton proposes a two-step philosophy aligning first with 0 MM and in a second step with 3 MM to the library of pre-microRNAs (loose matches). All other tools allow to manipulate the maximum number of mismatches, thus letting the decision to the user (miRExpress uses the % of sequence identity). Some tools (miRExpress, DSAP) force full-length alignments discarding therefore implicitly all variants (isomiRs) of the microRNA (see next section). It is clear that a high number of allowed mismatches might cause erroneous alignments to the wrong reference sequence, however, on the other hand, too stringent thresholds will discard valuable information. Therefore, the most adequate threshold for the maximum number of mismatches is another issue that should be assessed systematically in the future.

11.2.3.2 IsomiRs

IsomiRs are defined as variations of a mature microRNA sequence from its reference (predominant) sequence and have been commonly observed in cloning studies (Cummins et al. 2006; Landgraf et al. 2007). These variations include length variants produced by "incorrect" or alternative Dicer cleavage, nucleotide additions, and sequence variants due to RNA editing. In order to detect the length variants, the mapping must be done to a library of pre-microRNA sequences. If the position of the mature microRNA is known, the 5′ and 3′ base overhang can be determined. Creighton allows an overhang of three bases in 5′ and six bases in 3′. Currently, an isomiR analysis is possible in SeqBuster, DSAP, and E-miR. SeqBuster offers the most extensive analysis package. The variants are first classified into 5′ trimming, 3′ trimming, substitutions (RNA editing), and 3′ extensions. The results can be analyzed both, graphically and statistically. Furthermore, SeqBuster allows comparing the expression values of isomiRs over different samples, thus being able to detect differentially expressed variants. An analysis performed by these authors showed that generally the variants show differential expression if the predominant form is differentially expressed. In E-miR the differential expression of the variants is assessed in a different way. First the differentially expressed microRNAs are determined, detecting afterwards the isomiRs that significantly contribute to the observed difference. These authors found that the variants are uniformly expressed over the analyzed samples. Although, this finding might be specific for the used samples, it might also indicate the absence of functional alternative Dicer processing.

Regarding the analysis of RNA editing, it will be indispensable in the future to integrate quality values into the analysis of putative RNA editing as many substitutions might be sequencing errors and without quality values, the expected frequency of error-driven substitutions is difficult to assess.

In summary, the existence of isomiRs was shown rather recently and more research on this topic is needed. Right now, the tools allow quantifying the variation, but virtually nothing is known about the functional implications. However, these functional implications should be known in order to correctly classify the variants, i.e. whether to analyze them together with the predominant form or separately.

11.2.3.3 Multiple Mapping

Frequently, a given read maps to more than one reference sequence. The mapping to different positions in the genome might indicate the existence of more than one microRNA gene leading to the same mature sequence. However, it is less clear how to interpret multiple alignments to a non-redundant set of known microRNAs. Often microRNAs are members of families with closely related sequences. If mismatches are allowed in the mapping, sequencing errors can lead to multiple mapping. The miRNAkey authors report that up to 30% of all reads might have multiple or ambiguous alignments. They do however not specify the experimental setup where such high numbers were observed. Probably, the usage of very short reads (18 cycles)

might lead to a high number of ambiguous mappings. However, with a typical 36-cycle assay, the percentages should be much lower. We observed that when using miRanalyzer, typically less than 1% of all reads show multiple mappings. miRanalyzer reports those reads in a separate output instead of discarding them.

11.2.3.4 Filtering with ncRNA and Quantification of Contamination

During the preparation of the small RNA libraries it is unavoidable that other small or fragmented RNA molecules are sequenced as well. Some of the tools take this fact into account by mapping the reads also to Rfam (Gardner et al. 2009), RepBase (Kapitonov and Jurka 2008), and the coding regions of the genes or mRNA libraries. The mappings to these additional libraries can be used for two main purposes. First, they allow to quantify the degree of contamination with fragments from longer RNA molecules, and second, they can be used to filter out reads originating from other small RNA molecules prior to the mapping to known microRNAs or the genome (for prediction). The order of the mappings however varies between the different tools, which might have some impact on the results. For example, mirTools maps the reads to Rfam, miRBase, RepBase, and genes. In case of conflicts (a read maps to more than one library) they establish a hierarchy given priority to non-coding RNA from Rfam, followed by miRBase and afterwards repeat or gene-associated reads. DSAP eliminates first all reads that mapped to ncRNA (Rfam) and the remaining reads are then mapped to miRBase. miRanalyzer proceeds in a yet different way. The reads are first mapped to mature microRNAs, pre-microRNA, mRNA, Rfam, and finally to the genome in order to predict new microRNAs. After each step, the assigned reads are eliminated (for Rfam and mRNA certain parameters must hold to eliminate the reads, see (Hackenberg et al. 2009) for more details) so they cannot contribute again. The impact of the mapping order on the expression profiling has not been established so far. It might be however that the main importance of Rfam consideration lies in the prediction of novel microRNAs as reads from other small RNAs might lead to a higher number of false positive predictions.

11.2.4 Visualization

Some of the tools discussed here provide means to visualize the results. For example, miRtools offers graphically the length distribution of the unique reads and the total read count, pie-charts to summarize the mapping to the different reference libraries, and scatter plots to compare the expression values between two samples. SeqBuster provides a very complete visual analysis of microRNA variants. In this way, it allows to visualize the contribution of each variant to the microRNA (read count of the variant vs. count of all reads that map to the microRNA), analyzing the isomiRs by nucleotide position or depicting the differences between different samples.

11.2.5 Differential Expression

The detection of differentially expressed microRNAs is the final goal of many experimental assays. Frequently, the differentially expressed microRNAs will be the starting points for further, down-stream analyses. Therefore, most tools discussed here have incorporated this analysis feature now. Several statistical methods have been developed over the last years specially addressing the needs of digital expression profiling like RNA-seq (Marioni et al. 2008), DEGseq (Wang et al. 2010), edgeR (Robinson et al. 2010), or DESeq (Anders and Huber 2010). One of the first methods developed for digital expression data uses a Bayesian approach (Audic and Claverie 1997) and is applied by the miRtools server. SeqBuster implements a Z-test as proposed by Reinartz et al. (2002), E-miR uses a Limma (linear models)-based statistic, and miRNAkey is based on a chi-squared test. Recent methods are based either on the Poisson distribution (RNA-seq and DEGseq) or the negative binomial distribution (edgeR and DESeq). The latter method, DESeq, is used in miRanalyzer. Given that virtually all tools use a different method to detect differential expression, the way the data is processed also differs. Technical artifacts lead to fluctuations between the samples that need to be taken into account, otherwise, these fluctuations could be detected erroneously as differential expression. For example, miRtools normalizes the read counts to the total number of reads mapped to the microRNA reference before applying the statistical test. miRNAkey on the other side normalizes the data using a RPKM expression index (Mortazavi et al. 2008) which might be important for gene expression with a broad range of different transcript sizes; however, it is less clear if this normalization is meaningful for microRNA data. DESeq on the other hand takes raw counts as input doing the normalization internally and therefore miRanalyzer provides an expression matrix with absolute read counts.

miRtools has two different modules to detect differential expression. The first allows the comparison of two samples, and the other, which takes previously generated microRNA expression files as input, allows up to three samples per condition. miRanalyzer has no limitation on the number of samples. First, all samples are processed with miRanalyzer and second, by means of the unique ID assigned to each job, the groups can be defined without any limitations on the number of samples.

Finally, at present miRanalyzer seems to be the only tool able to detect differential expression among the predicted novel microRNAs. Only the UEA sRNA toolkit has a similar functionality being able to detect differentially expressed loci.

11.3 Prediction of Novel MicroRNAs

As mentioned in the introduction, the detection of new microRNAs has been facilitated enormously by means of HTS technologies. This is basically due to two reasons: first, the search for new microRNAs can be limited to expressed sequences, and second, the Dicer processing leaves characteristic pattern which sometimes can be

detected by means of these new technologies. In this way, virtually all prediction algorithms implemented into the discussed tools (or proposed in other protocols) are based on previous works adding the new expression layer information. We can distinguish two types of methods, those based on homology and those based on machine learning.

11.3.1 Homology-Based Approaches

The basic idea behind this approach is that microRNAs discovered in other (related) organisms might be conserved and therefore be also present in the species under analysis. Briefly, the rough procedure is the following: a set of exogeneous microRNAs is aligned to the species genome, a sequence around the mapping position is extracted, and the secondary structure of this sequence is calculated. If the secondary structure holds some parameters thresholds like minimum binding energy, the mapping is reported as a putative novel microRNA (Artzi et al. 2008; Dezulian et al. 2006). The miRExpress package implements a method following this philosophy. In their analysis of two human cell lines, they mapped all remaining reads (those that did not map to known human microRNAs) to all other mammalian microRNAs detecting 40 and 39 putatively novel microRNAs.

11.3.2 Machine-Learning Approaches

A significant number of methods have been developed to predict microRNA genes like miPred (Jiang et al. 2007), miRFinder (Huang et al. 2007), or ProMiR (Nam et al. 2006) recently reviewed by Li et al. (2010). Based on this knowledge new methods have been developed adding the new information granted by HTS experiments. Currently, the prediction of novel microRNAs is available in miRtools (through miRDeep), miRanalyzer, and the UEA sRNA toolkit. However, methods or protocols have been described also by Morin et al. and Creighton et al.

There are notable differences in the details of each method; however, some basic steps are shared by all of them:

- Map the reads against the genome sequence.
- Cluster together the reads which map to the same locus (isomiRs, sequencing errors, usage of non-adapter trimmed reads)
- Extract the genomic sequence plus some flanking regions in order to include the full pre-microRNA sequence
- Determine the secondary structure of the extracted sequence and reject non-hairpin structures
- Calculate properties based on structural and compositional features, expression values and/or traces left by Dicer processing (like existence of mature* sequence).
- Calculate the probability of the candidate to be a novel microRNA

A detailed comparison of the methods is beyond the scope of this chapter; I will, however, mention some important differences. Usually, just a single sequence is extracted out of the genome to check for secondary fold-back structures. miRanalyzer calculates up to 20 different secondary structures with different lengths taking into account that the microRNA can be located at either of the two arms of the pre-microRNA. This slows down the prediction of microRNA enormously; however, it reduces the probability that flanking regions change the secondary structure of the candidate pre-microRNA. A second important difference is given by the number of predicted microRNAs. It seems that miRanalyzer predicts much more novel microRNAs than, for example, miRDeep does or compared to the number reported by Morin et al. This might indicate that miRanalyzer is more sensitive than others but very likely less specific. However, the different methods have not yet been benchmarked against each other based on identical data sets. The prediction quality can only be compared by means of indicators like sensibility, specificity, and correlation given by the authors. All these values are obtained on different data sets and therefore no conclusions can be drawn so far. It would be an important task and very useful for the users of these tools to carry out a large-scale comparison in order to quantify the differences in prediction quality between these methods.

11.4 Downstream Analysis

Especially with a list of differentially expressed microRNAs at hand, usually several downstream analyses need to be carried out in order to obtain biological knowledge. Of outstanding interest is of course the detection of the microRNA target genes. Several techniques are coming up to detect microRNA/mRNA interactions experimentally like HITS-CLIP, high-throughput sequencing of RNAs isolated by cross-linking immunoprecipitation (Chi et al. 2009) or SILAC, and pSILAC to measure directly the impact of microRNAs on the protein abundance (Baek et al. 2008; Selbach et al. 2008). However, these data so far cannot be routinely generated together with the microRNA sequencing data and therefore the detection of microRNA/mRNA interactions is still strongly based on predictions [please see Li et al. (2010) for a review]. For example, miRanalyzer provides the target genes predicted by microCosm (Griffiths-Jones et al. 2008) for each of the detected known microRNAs. Furthermore, it allows the detection of putative target genes for novel microRNAs by means of the TargetSpy algorithm (Sturm et al. 2010). Once the target genes have been detected, enrichment/depletion analysis (Al-Shahrour et al. 2004) can be carried out in order to translate the gene list into biological knowledge.

If mRNA expression data is available for the same samples, a more directed search for regulated genes can be carried out. A common approach is to look for the enrichment of target sites from over-expressed microRNAs within the down-regulated transcripts. There are already some tools available for this kind of analysis like DIANA-mirExTra (Alexiou et al. 2010), GeneSet2miRNA (Antonov et al. 2009), or miRonTop (Le Brigand et al. 2010).

11.5 Outlook

The tools and protocols presented in this chapter concentrate principally on the profiling of microRNA expression data, prediction of novel microRNAs, and the detection of differentially expressed microRNAs. However, the ultimate goal of most experiments might be to detect those pathways that actively participate in the development of a given pathology or more generally being affected between two conditions. In order to archive those (microRNA) regulatory networks, not only microRNA expression data, but also mRNA, proteomics and probably also methylation data need to be analyzed together to obtain a complete understanding on the molecular background. Thus, the extension of the methods presented here towards an integral analysis of data from different experiments together with functional analyses will be one major goal in the future.

Acknowledgments This work was supported by the Ministry of Innovation and Science of the Spanish Government [BIO2010-20219], the Junta de Andalucia [P07FQM3163] and the "Juan de la Cierva" fellowship.

References

Al-Shahrour, F., Diaz-Uriarte, R. and Dopazo, J. (2004) FatiGO: a web tool for finding significant associations of Gene Ontology terms with groups of genes, *Bioinformatics*, 20, 578–580.

Alexiou, P., Maragkakis, M., Papadopoulos, G.L., Simmosis, V.A., Zhang, L. and Hatzigeorgiou, A.G. (2010) The DIANA-mirExTra web server: from gene expression data to microRNA function, *PLoS One*, 5, e9171.

Alvarez-Garcia, I. and Miska, E.A. (2005) MicroRNA functions in animal development and human disease, *Development*, 132, 4653–4662.

Anders, S. and Huber, W. (2010) Differential expression analysis for sequence count data, *Genome Biol*, 11, R106.

Antonov, A.V., Dietmann, S., Wong, P., Lutter, D. and Mewes, H.W. (2009) GeneSet2miRNA: finding the signature of cooperative miRNA activities in the gene lists, *Nucleic Acids Res*, 37, W323–328.

Artzi, S., Kiezun, A. and Shomron, N. (2008) miRNAminer: a tool for homologous microRNA gene search, *BMC Bioinformatics*, 9, 39.

Audic, S. and Claverie, J.M. (1997) The significance of digital gene expression profiles, *Genome Res*, 7, 986–995.

Baek, D., Villen, J., Shin, C., Camargo, F.D., Gygi, S.P. and Bartel, D.P. (2008) The impact of microRNAs on protein output, *Nature*, 455, 64–71.

Bar, M., Wyman, S.K., Fritz, B.R., Qi, J., Garg, K.S., Parkin, R.K., Kroh, E.M., Bendoraite, A., Mitchell, P.S., Nelson, A.M., Ruzzo, W.L., Ware, C., Radich, J.P., Gentleman, R., Ruohola-Baker, H. and Tewari, M. (2008) MicroRNA discovery and profiling in human embryonic stem cells by deep sequencing of small RNA libraries, *Stem Cells*, 26, 2496–2505.

Bentwich, I., Avniel, A., Karov, Y., Aharonov, R., Gilad, S., Barad, O., Barzilai, A., Einat, P., Einav, U., Meiri, E., Sharon, E., Spector, Y. and Bentwich, Z. (2005) Identification of hundreds of conserved and nonconserved human microRNAs, *Nat Genet*, 37, 766–770.

Berezikov, E., Guryev, V., van de Belt, J., Wienholds, E., Plasterk, R.H. and Cuppen, E. (2005) Phylogenetic shadowing and computational identification of human microRNA genes, *Cell*, 120, 21–24.

Buermans, H.P., Ariyurek, Y., van Ommen, G., den Dunnen, J.T. and t Hoen, P.A. (2010) New methods for next generation sequencing based microRNA expression profiling, *BMC Genomics*, 11, 716.

Creighton, C.J., Reid, J.G. and Gunaratne, P.H. (2009) Expression profiling of microRNAs by deep sequencing, *Brief Bioinform*, 10, 490–497.

Cummins, J.M., He, Y., Leary, R.J., Pagliarini, R., Diaz, L.A., Jr., Sjoblom, T., Barad, O., Bentwich, Z., Szafranska, A.E., Labourier, E., Raymond, C.K., Roberts, B.S., Juhl, H., Kinzler, K.W., Vogelstein, B. and Velculescu, V.E. (2006) The colorectal microRNAome, *Proc Natl Acad Sci USA*, 103, 3687–3692.

Chi, S.W., Zang, J.B., Mele, A. and Darnell, R.B. (2009) Argonaute HITS-CLIP decodes microRNA-mRNA interaction maps, *Nature*, 460, 479–486.

Dezulian, T., Remmert, M., Palatnik, J.F., Weigel, D. and Huson, D.H. (2006) Identification of plant microRNA homologs, *Bioinformatics*, 22, 359–360.

Esquela-Kerscher, A. and Slack, F.J. (2006) Oncomirs – microRNAs with a role in cancer, *Nat Rev Cancer*, 6, 259–269.

Friedlander, M.R., Chen, W., Adamidi, C., Maaskola, J., Einspanier, R., Knespel, S. and Rajewsky, N. (2008) Discovering microRNAs from deep sequencing data using miRDeep, *Nat Biotechnol*, 26, 407–415.

Gardner, P.P., Daub, J., Tate, J.G., Nawrocki, E.P., Kolbe, D.L., Lindgreen, S., Wilkinson, A.C., Finn, R.D., Griffiths-Jones, S., Eddy, S.R. and Bateman, A. (2009) Rfam: updates to the RNA families database, *Nucleic Acids Res*, 37, D136–140.

Griffiths-Jones, S., Saini, H.K., van Dongen, S. and Enright, A.J. (2008) miRBase: tools for microRNA genomics, *Nucleic Acids Res*, 36, D154–158.

Hackenberg, M., Sturm, M., Langenberger, D., Falcon-Perez, J.M. and Aransay, A.M. (2009) miRanalyzer: a microRNA detection and analysis tool for next-generation sequencing experiments, *Nucleic Acids Res*, 37, W68–76.

Huang, P.J., Liu, Y.C., Lee, C.C., Lin, W.C., Gan, R.R., Lyu, P.C. and Tang, P. (2010) DSAP: deep-sequencing small RNA analysis pipeline, *Nucleic Acids Res*, 38, W385–391.

Huang, T.H., Fan, B., Rothschild, M.F., Hu, Z.L., Li, K. and Zhao, S.H. (2007) MiRFinder: an improved approach and software implementation for genome-wide fast microRNA precursor scans, *BMC Bioinformatics*, 8, 341.

Jiang, P., Wu, H., Wang, W., Ma, W., Sun, X. and Lu, Z. (2007) MiPred: classification of real and pseudo microRNA precursors using random forest prediction model with combined features, *Nucleic Acids Res*, 35, W339–344.

Kapitonov, V.V. and Jurka, J. (2008) A universal classification of eukaryotic transposable elements implemented in Repbase, *Nat Rev Genet*, 9, 411–412; author reply 414.

Kozomara, A. and Griffiths-Jones, S. (2011) miRBase: integrating microRNA annotation and deep-sequencing data, *Nucleic Acids Res*, 39, D152–157.

Lagos-Quintana, M., Rauhut, R., Lendeckel, W. and Tuschl, T. (2001) Identification of novel genes coding for small expressed RNAs, *Science*, 294, 853–858.

Landgraf, P., Rusu, M., Sheridan, R., Sewer, A., Iovino, N., Aravin, A., Pfeffer, S., Rice, A., Kamphorst, A.O., Landthaler, M., Lin, C., Socci, N.D., Hermida, L., Fulci, V., Chiaretti, S., Foa, R., Schliwka, J., Fuchs, U., Novosel, A., Muller, R.U., Schermer, B., Bissels, U., Inman, J., Phan, Q., Chien, M., Weir, D.B., Choksi, R., De Vita, G., Frezzetti, D., Trompeter, H.I., Hornung, V., Teng, G., Hartmann, G., Palkovits, M., Di Lauro, R., Wernet, P., Macino, G., Rogler, C.E., Nagle, J.W., Ju, J., Papavasiliou, F.N., Benzing, T., Lichter, P., Tam, W., Brownstein, M.J., Bosio, A., Borkhardt, A., Russo, J.J., Sander, C., Zavolan, M. and Tuschl, T. (2007) A mammalian microRNA expression atlas based on small RNA library sequencing, *Cell*, 129, 1401–1414.

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

Le Brigand, K., Robbe-Sermesant, K., Mari, B. and Barbry, P. (2010) MiRonTop: mining microRNAs targets across large scale gene expression studies, *Bioinformatics*, 26, 3131–3132.

Lee, R.C., Feinbaum, R.L. and Ambros, V. (1993) The C. elegans heterochronic gene lin-4 encodes small RNAs with antisense complementarity to lin-14, *Cell*, 75, 843–854.

Li, H. and Durbin, R. (2009) Fast and accurate short read alignment with Burrows-Wheeler transform, *Bioinformatics*, 25, 1754–1760.

Li, L., Xu, J., Yang, D., Tan, X. and Wang, H. (2010) Computational approaches for microRNA studies: a review, *Mamm Genome*, 21, 1–12.

Lim, L.P., Lau, N.C., Weinstein, E.G., Abdelhakim, A., Yekta, S., Rhoades, M.W., Burge, C.B. and Bartel, D.P. (2003) The microRNAs of Caenorhabditis elegans, *Genes Dev*, 17, 991–1008.

Marioni, J.C., Mason, C.E., Mane, S.M., Stephens, M. and Gilad, Y. (2008) RNA-seq: an assessment of technical reproducibility and comparison with gene expression arrays, *Genome Res*, 18, 1509–1517.

Morin, R.D., O'Connor, M.D., Griffith, M., Kuchenbauer, F., Delaney, A., Prabhu, A.L., Zhao, Y., McDonald, H., Zeng, T., Hirst, M., Eaves, C.J. and Marra, M.A. (2008) Application of massively parallel sequencing to microRNA profiling and discovery in human embryonic stem cells, *Genome Res*, 18, 610–621.

Mortazavi, A., Williams, B.A., McCue, K., Schaeffer, L. and Wold, B. (2008) Mapping and quantifying mammalian transcriptomes by RNA-Seq, *Nat Methods*, 5, 621–628.

Moxon, S., Schwach, F., Dalmay, T., Maclean, D., Studholme, D.J. and Moulton, V. (2008) A toolkit for analysing large-scale plant small RNA datasets, *Bioinformatics*, 24, 2252–2253.

Nam, J.W., Kim, J., Kim, S.K. and Zhang, B.T. (2006) ProMiR II: a web server for the probabilistic prediction of clustered, nonclustered, conserved and nonconserved microRNAs, *Nucleic Acids Res*, 34, W455–458.

Pantano, L., Estivill, X. and Marti, E. (2010) SeqBuster, a bioinformatic tool for the processing and analysis of small RNAs datasets, reveals ubiquitous miRNA modifications in human embryonic cells, *Nucleic Acids Res*, 38, e34.

Pasquinelli, A.E., Reinhart, B.J., Slack, F., Martindale, M.Q., Kuroda, M.I., Maller, B., Hayward, D.C., Ball, E.E., Degnan, B., Muller, P., Spring, J., Srinivasan, A., Fishman, M., Finnerty, J., Corbo, J., Levine, M., Leahy, P., Davidson, E. and Ruvkun, G. (2000) Conservation of the sequence and temporal expression of let-7 heterochronic regulatory RNA, *Nature*, 408, 86–89.

Prufer, K., Stenzel, U., Dannemann, M., Green, R.E., Lachmann, M. and Kelso, J. (2008) PatMaN: rapid alignment of short sequences to large databases, *Bioinformatics*, 24, 1530–1531.

Reinartz, J., Bruyns, E., Lin, J.Z., Burcham, T., Brenner, S., Bowen, B., Kramer, M. and Woychik, R. (2002) Massively parallel signature sequencing (MPSS) as a tool for in-depth quantitative gene expression profiling in all organisms, *Brief Funct Genomic Proteomic*, 1, 95–104.

Reinhart, B.J., Slack, F.J., Basson, M., Pasquinelli, A.E., Bettinger, J.C., Rougvie, A.E., Horvitz, H.R. and Ruvkun, G. (2000) The 21-nucleotide let-7 RNA regulates developmental timing in Caenorhabditis elegans, *Nature*, 403, 901–906.

Robinson, M.D., McCarthy, D.J. and Smyth, G.K. (2010) edgeR: a Bioconductor package for differential expression analysis of digital gene expression data, *Bioinformatics*, 26, 139–140.

Ronen, R., Gan, I., Modai, S., Sukacheov, A., Dror, G., Halperin, E. and Shomron, N. (2010) miRNAkey: a software for microRNA deep sequencing analysis, *Bioinformatics*, 26, 2615–2616.

Selbach, M., Schwanhausser, B., Thierfelder, N., Fang, Z., Khanin, R. and Rajewsky, N. (2008) Widespread changes in protein synthesis induced by microRNAs, *Nature*, 455, 58–63.

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

Sturm, M., Hackenberg, M., Langenberger, D. and Frishman, D. (2010) TargetSpy: a supervised machine learning approach for microRNA target prediction, *BMC Bioinformatics*, 11, 292.

Wang, L., Feng, Z., Wang, X. and Zhang, X. (2010) DEGseq: an R package for identifying differentially expressed genes from RNA-seq data, *Bioinformatics*, 26, 136–138.

Wang, W.C., Lin, F.M., Chang, W.C., Lin, K.Y., Huang, H.D. and Lin, N.S. (2009) miRExpress: analyzing high-throughput sequencing data for profiling microRNA expression, *BMC Bioinformatics*, 10, 328.

Wienholds, E. and Plasterk, R.H. (2005) MicroRNA function in animal development, *FEBS Lett*, 579, 5911–5922.

Zhang, Z., Schwartz, S., Wagner, L. and Miller, W. (2000) A greedy algorithm for aligning DNA sequences, *J Comput Biol*, 7, 203–214.

Zhu, E., Zhao, F., Xu, G., Hou, H., Zhou, L., Li, X., Sun, Z. and Wu, J. (2010) mirTools: microRNA profiling and discovery based on high-throughput sequencing, *Nucleic Acids Res*, 38, W392–397.

Chapter 12
Dissecting Splicing Regulatory Network by Integrative Analysis of CLIP-Seq Data

Michael Q. Zhang

Abstract Detecting protein–RNA interaction and identifying RNA-binding protein (RBP) targets on a global scale have become increasingly important for understanding RNA regulatory mechanisms and reverse-engineering RNA regulation network at systems level. Significant progress was made in the past few years. Cross-linking and immunoprecipitation (CLIP) coupled with high-throughput sequencing (HITS) the method of choice for transcriptome-wide coverage as well as for high-resolution mapping of RBP sites and for in vivo target identification. In this chapter, we introduce the current status of the technology, with a focus on bioinformatic analysis of HITS-CLIP or CLIP-seq data, in understanding RNA splicing regulation network through building RNA splicing maps.

12.1 Experimental Background

Science is about studying interaction. Genome-wide RNA–RBP (such as SF, splicing factor) interaction biochemistry has always been lagging behind genome-wide DNA–DBP (DNA-binding protein) (such as TF, transcription factor) works, largely because of the difficulty associated with handling RNA (less stable, more complex structurally, etc.). There are excellent reviews, e.g., Licatalosi and Darnell (2010) on RNA processing and regulation, Darnell (2010) on HITS-CLIP technology, and Witten and Ule (2011) on RNA splicing maps. Here we try to briefly summarize relevant experimental breakthroughs and refer the readers to the above references for details.

M.Q. Zhang (✉)
Department of Molecular and Cell Biology, Center for Systems Biology, The University of Texas at Dallas, RL11, 800 West Campbell Road, Richardson, TX 75080, USA

Bioinformatics Division, TNLIST, Tsinghua University, Beijing 100084, China
e-mail: michael.zhang@utdallas.edu

N. Rodríguez-Ezpeleta et al. (eds.), *Bioinformatics for High Throughput Sequencing*,
DOI 10.1007/978-1-4614-0782-9_12, © Springer Science+Business Media, LLC 2012

The revolution of high-throughput sequencing (HITS) or next-generation sequencing (NGS) technologies brings powerful tools to study protein–DNA (ChIP-Seq) and protein–RNA (HITS-CLIP, or CLIP-Seq) interactions, histone modifications, DNase hypersensitive sites, and the transcriptome at a genome-wide scale (Wold and Myers 2008). In conjunction with mRNA-Seq, which has already emerged as an alternative to expression microarrays for detecting and quantifying alternative mRNA transcripts in specific conditions (Pepke et al. 2009; Wang et al. 2009; Zhang and Zhang 2011; Chap. 11), functional targets of transcription and splicing may be identified genome-wide with base-pair resolution. Especially with the CLIP-Seq technology, recent studies reveal unprecedented extensiveness of alternative pre-mRNA splicing and provide novel insights into global mechanisms of splicing regulation.

12.1.1 RNA–RBP Analysis by CLIP-Seq

In order to overcome some of the early problems in the RNA IP-based technology, UV cross-linking and immunoprecipitation (CLIP) technology was developed (Ule et al. 2003; Jensen and Darnell 2008). After treated with UV-B irradiation of the intact cells or tissues, covalent bonds between RNA and RBP in close contact can be formed. This physical link allows RNA–RBP complexes to be purified by using immunoprecipitation and denaturing gel electrophoresis. RNAs are sheared into fragments (from ~20 to 100 nt). The RBP is then digested by proteinase K and the RNA is purified. After ligated with RNA linkers, cDNA is synthesized with antisense primer and reverse transcriptase (RT) to generate templates for HITS. The advantage of UV cross-linking, compared with formaldehyde cross-linking widely used in ChIP-seq, is that it only cross-links direct protein–RNA contact and does not induce protein–protein cross-links, although it does induce RNA–RNA cross-links, which can help solving secondary RNA structures. After protein digestion, an amino acid (or a short peptide) remains at the RNA cross-linking site that leads to the frequent truncation of cDNAs previously exploited to map cross-linking sites using primer extension assays. A newly improved protocol iCLIP (individual nucleotide resolution CLIP: König et al. 2010), using cDNA circularization to prepare the cDNA library, allows for the HITS of cDNAs that truncate a peptide (because of the irreversibility of the UV cross-link) and mapping of the binding sites at higher resolution. Also the use of random barcodes to mark cDNAs during library preparation allows to determining if identical sequences arose from multiple independent cDNAs or PCR artifact. Because reverse transcriptase used in CLIP frequently skips the cross-linked amino acid–RNA adduct, this can result in a nucleotide deletion. Genome-wide analysis of these cross-linking-induced mutation sites (CIMS) in Nova and Argonaut (Ago) HITS-CLIP data demonstrated deletions in ~8–20% mRNA tags. The new CIMS analysis method (Zhang and Darnell 2011) can systematically analyze HITS-CLIP data to identify exact cross-linking sites, and thereby determine protein–RNA interactions at single-nucleotide resolution.

12.1.2 Available HITS Platforms

There are several HITS platforms commercially available, including 454 (Roche), Illumina Genome Analyzer (Illumina/Solexa), and AB SOLiD (Shendure and Ji 2008). Without going into details about the biochemistry underlying each sequencing platform, the major advantage of 454 pyrosequencing is its read length: approximately 400,000 reads, each of 300–400 nt, can be generated per instrument-run. In contrast, Illumina and SOLiD systems have a much higher throughput (>1 G base pairs in an instrument-run that includes ~8 lanes) with a much lower per-base cost, but the read length is significantly shorter (~30–100 nt). For CLIP-Seq, normally single-end short reads are sufficient (unlike RNA-seq) unless more complex RNA structural motif is involved, and one sample run would give sufficient coverage, too. Although Illumina and SOLiD platforms are in principle more useful; historically 454 and later more Illumina have been used in published CLIP-Seq studies. However, new sequencing technologies are still being actively developed and "portable" HITS systems, e.g., Ion Torrent, 454 Jr. and MiSeq will emerge in more individual labs in the near future.

12.1.3 Genomic Mapping of CLIP-Seq Tags

Because detailed analysis will depend on the sequencing platform and/or even specific protocols, we use Zhang et al. (2010) as an example for the typical steps usually involved. After raw sequencing, reads are obtained and filtered to exclude those that failed quality controls (e.g., quality score ≥20, data generated by Illumina 1G sequencer with 32 nt in length). Reads (CLIP-tags) can be mapped to the genome by a favorite read-mapping program (e.g., ELAND included in the Illumina Genome Analyzer pipeline). For efficiency, each read can be trimmed iteratively at the 3′ lower quality end and aligned using different sizes from 25 to 32 nt (newer Illumina sequencing read can be doubled these sizes), requiring ≤2 mismatches. A read is kept only if mapped to an unambiguous locus. If unambiguous mapping was possible with different sizes, the one with minimum mismatches and maximum size is retained. For each individual CLIP experiment, tags with the same starting genomic coordinate should be collapsed to remove potential RT-PCR duplicates, and identify unique tags for further analysis.

12.1.4 Clustering of CLIP-Seq Tags

Various clustering or peak-calling algorithms may be used to segment the RNA sequence region into binding and non-binding regions. In Zhang et al. (2010), a two-state Hidden Markov Model (HMM) was used to define the (Nova-bound) tag-clusters. Briefly, the algorithm first calculated the number of overlapping CLIP

tags at each nucleotide position, and then sampled the resulting CLIP tag coverage profile at a 5-nt resolution. This sampled profile was used to segment the genome into CLIP clusters and non-cluster regions, as represented by the two states, "+" and "–", respectively. To reduce computation, they first partitioned the genome into segments by grouping neighboring reads ≤200 nt apart. Only segments with ≥2 reads were kept for further analysis (there were 386,723 segments, mean 224 nt, median 134 nt, std 328 nt in the original experiment). They then ran HMM for a two-round procedure which is conceptually similar to the Baum-Welch algorithm, an iterative method to decompose unlabeled data and to estimate model parameters. The resulting clusters were ranked by peak height (PH), i.e., the number of tags in the position with the highest coverage. They identified ~280,000 CLIP tag clusters from 20 independent HITS-CLIP experiments.

Yeo et al. (2009) used an alternative method to identify enriched FOX2-bound regions. They first computationally extended each mapped read in the 5′-to-3′ direction by 100 nt (the average length of RNA fragments). Then they determined the false-discovery rate (FDR) for each position by computing the "background" frequency after randomly placing the same number of extended reads within the gene for 100 iterations. For a particular height, the modified FDR was computed as the ratio of the probability of observing background positions of at least that height to one standard deviation above the average probability of observing actual positions of at least that height. Finally, binding clusters were defined by grouping positions that satisfied FDR <0.001 and occurred within 50 nt of each other. They identified ~6,000 FOX2 binding clusters in the CLIP-seq analysis of human embryonic stem cells.

12.1.5 Motif Analysis

Once binding clusters are identified, motif enrichment analysis may be carried out. RNA-binding motifs are in principle more subtle than DNA-binding motifs, not only because it is known that many functional RNA-binding motifs are very short and degenerate (tend to occur in cluster), but many longer binding sites also often have low complexity and may contain secondary (or even higher order) structure components (Kazan et al. 2010), perhaps reflecting the greater flexibility in RNA–RBP interactions and dynamics (Serganov and Patel 2008). Fortunately most of the initial CLIP-seq studies were carried out for RBPs with binding motifs largely known, and any word counting or alignment-based motif finding tools would work. Motif discovery tools such as MEME (Bailey and Elkan 1994), Gibbs motif sampler (e.g., Thompson et al. 2005), and DWE (Sumazin et al. 2005) or DME (Smith et al. 2005) may be used to identify binding site motifs. Careful modeling of binding motifs can yield valuable structural and functional insight to RNA–RBP interaction. One such example is the "RNA nucleosome" structures revealed after iCLIP-seq analysis of hnRNP C-binding sites (König et al. 2010).

12.2 Integrative Analysis

Although identifying protein–RNA interactions with high resolution in a genome-wide manner is necessary, single genome-wide data sets are often not sufficient for getting significant insight into splicing regulation mechanism on its own. The success of genome-wide studies lies in the integration of multiple, independent large-scale data sets.

12.2.1 RNA Splicing Maps

Combining CLIP-seq data with the transcriptome analysis of splicing profiles allows building the so-called RNA splicing maps that determine the position-dependent regulatory effects of protein–RNA interactions (Witten and Ule 2011). The initial approach used with Nova and Fox proteins combined bioinformatically identified binding sites (based on conservation, motif scores, etc.) with splicing profiles identified by splicing-junction microarrays (Ule et al. 2006) or Rosetta custom whole-transcript microarrays (Zhang et al. 2008). Later protein–RNA binding sites were determined experimentally by CLIP-seq directly (Licatalosi et al. 2008; Yeo et al. 2009). Moreover, instead of microarrays, splicing profiles can now be derived from RNA-seq (Brooks et al. 2011).

To construct an RNA splicing map for a given RBP in a given cell type, the first step is to identify differential regulation splicing transcripts (the potential targets) under the RBP perturbation (usually by over expression, or by knock-out or knock-down) from the mRNA expression data. There are many patterns of alternative splicing; most genome-wide studies have been focused on the major class: namely, cassette exon (inclusion or skipping event); other minor classes, such as alternative 5′- or 3′-splicing and intron retention are also often considered. The second step is to map the RBP-binding sites within or near the alternatively spliced exons in each class using CLIP-seq data and/or bioinformatic motif analysis. The third is to model how the position (and maybe also the affinity measured by the motif score) of the binding site will affect the splicing pattern change (outcome of AS: alternative splicing) quantitatively. Such modeling is very similar to model TFBSs and differential transcription, many machine learning or statistical classification and regression methods may be used. In the case of modeling Nova-YCAY interaction or Fox-UGCAUG interaction sites, the splicing maps for both show that the conserved binding sites upstream or downstream of alternative exons significantly correlate with (hence, can predict) RBP-dependent exon skipping or inclusion events, respectively.

In the future, combination of CLIP-seq and RNA-seq will become the most comprehensive and powerful approach for building the RNA-splicing maps. Current limitation comes mainly from the difficulty of splicing isoform identification and quantitation in transcriptome analysis with RNA-seq data (see Chap. 11, Zhang and Zhang 2011). Some new tools allow specific detection, and the quantification of

alternative exons has become available, e.g., Bayes factor (BF) approach by MISO (Katz et al. 2010) and inclusion ratio (IR) approach by SpliceTrap (Wu et al. 2011). By combining CLIP-seq and RNA-seq data, the RNA map for hnRNP H is generated (Katz et al. 2010) which confirmed the early study (Xiao et al. 2009) using bioinformatically predicted binding motif (poly-G runs), namely, that the regulation (inclusion) effect is stronger for hnRNP H binding in the downstream intron and is reversed for events with exonic-binding sites (or CLIP tags).

12.2.2 Functional Target Identification

Visualization of CLIP-seq and mRNA-Seq data in the genome browser provides effective and intuitive assess of data quality, and integrative analysis with other information (such as sequence conservation) already included in the genome browser. In general, multiple tracks are generated for each sample: an exon intensity track in the "wiggle" format that gives read coverage profiles at a nucleotide-level resolution and an exon-junction track that lists individual exon-junction reads. Additional tracks often display CLIP-tag clusters, binding site motif predictions, and conservation profiles. By zoom-in and zoom-out, it is easy to get an intuitive sense if reads are mostly in exons, if exon intensity changes specifically in the alternatively spliced region, where are binding sides localized and if their positions correlate with nearby alternative splicing events (Fig. 12.1).

As demonstrated recently, the integration of multiple dataset information is the key for identification of true functional targets. For example, Zhang et al. (2010) showed that in addition to CLIP clusters, bioinformatically predicted YCAY clusters are equally important when integrated with the expression data for identification of true functional targets of Nova in the mouse brain. In order to integrate diverse types of genome-wide datasets, they designed a sophisticated Bayesian network (BN) model with 17 notes (variables) for four types of data: CLIP clusters and YCAY clusters in each annotated cassette exon or flanking upstream and downstream intron flanking regions, differentially expressed transcripts from splicing microarray comparison of wild-type and Nova KO brains, and evolutionary conservation signatures. Starting from 13,357 annotated cassette exons, 363 are predicted as the direct Nava targets (FDR $\leq$0.01) with 588 Nova-regulated AS events, achieving an impressive estimated sensitivity of 75–78% and ~90% validation rate.

12.2.3 RNA Regulation Network and Combinatorial Controls

With the knowledge of regulator–target links, RNA regulation network can be built using conventional Gene Regulation Network (GRN) analysis tools. Multiple CLIP-seq data will facilitate the study of combinatorial controls of interacting RBPs, as was done for example in Nova – Fox2 case (Zhang et al. 2010). CLIP-seq technology

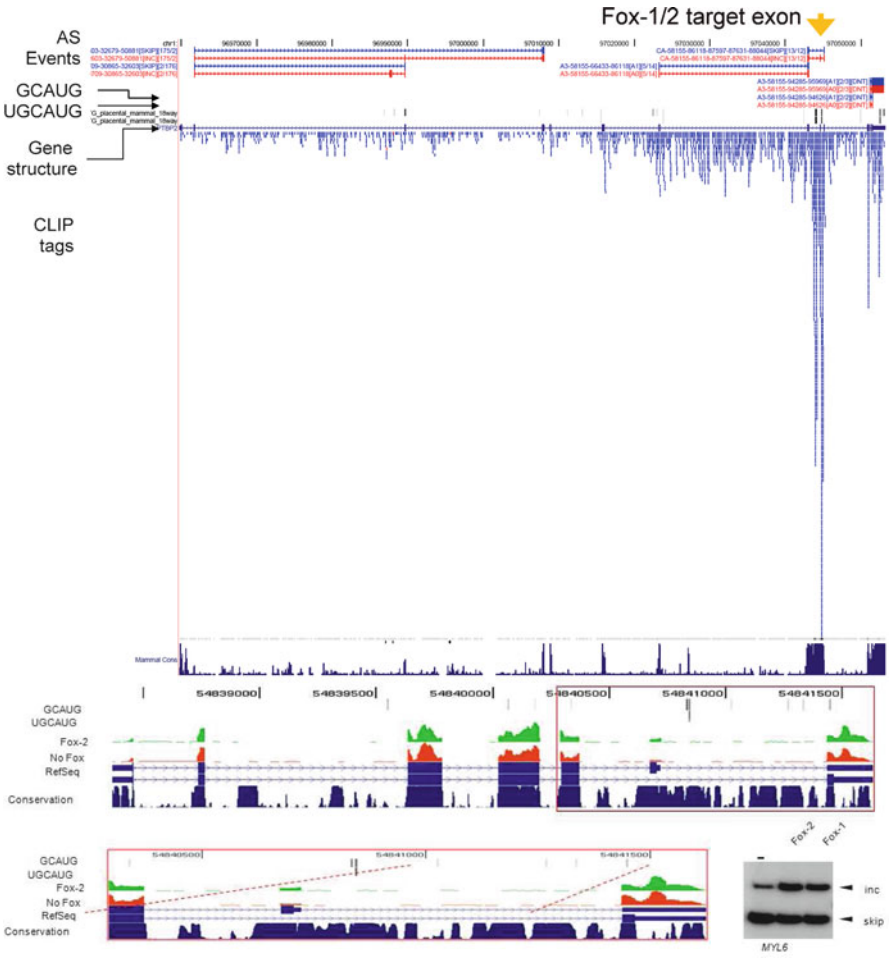

Fig. 12.1 A typical example of visualization of CLIP-seq tags and PE-mRNA-Seq to identify Fox-2 target exons (Courtesy of Dr. Chaolin Zhang)

has been applied to other post-transcriptional regulation analysis (Table 12.1). For example, CLIP-seq was extended to the study of ternary interactions between an RBP (Ago), mRNA, and miRNAs (Chi et al. 2009). Different miRNA targets may be "painted" on a standard pathway network (see Fig. 5c in Chi et al. 2009) to reveal their biological function.

Recently, computational portals are being developed to facilitate CLIP-seq data manipulation and analysis. For example, CLIPZ (Khorshid et al. 2011) at http://www.clipz.unibas.ch has been developed as a database and analysis environment for experimentally determined binding sites of RBPs, and aims to provide an open

Table 12.1 RNA–RBP studied by CLIP or CLIP-seq

RBP	Experimental method	Tissue/cell type	Binding motif	References
Nova	CLIP and CLIP-seq	Mouse brain	YCAY	Ule et al. (2003), Licatalosi et al. (2008), Racca et al. (2010), Yano et al. (2010) and Zhang et al. (2010)
PTB	CLIP-seq	Hela	Y-rich: UCUUC and CUCUCU	Xue et al. (2009)
Tagged-Khd1	CLIP-seq	*S. carevisiae*	UGCAU	Wolf et al. (2010)
hnRNP A1	CLIP	Hela	UAGGGA/U	Guil and Caceres (2007)
Fox2/RBM9	CLIP-seq	hESC	(U)GCAUG	Yeo et al. (2009)
SFRS1/SF2/ASF	CLIP and CLIP-seq	HEK293T	GAAGAA	Sanford et al. (2008, 2009)
Tagged-Rrm4	CLIP	Flamentous fungus *Ustilago maydis*	CA-rich: CACC, CAAC	Becht et al. (2006)
CUGBP1	CLIP	Mouse hindbrain	CUG triplet repeat	Daughters et al. (2009)
Tagged-snRNPs	CLIP-seq	*S. cerevisiae*	rRNA, snoRNA	Granneman et al. (2009) and Bohnsack et al. (2009)
Ro homologo Rsr	CLIP	Eubacterium Deinococcus radiodurnas	16S and 23S rRNAs	Wurtmann and Wolin (2010)
hnRNP C	CLIP-seq	Hela		König et al. 2010
hnRNP H1	CLIP-seq	293 T cells	Poly(G) runs	Katz et al. (2010)
Ago	CLIP-seq	Mouse brain, hela	miRNA seeds	Chi et al. (2009)
Tagged-Ago	PAR-CLIP	HEK293	miRNA seeds	Hafner et al. (2010)
Alg-1	CLIP-seq	*C. elegans*	miRNA seeds	Zisoulis et al. (2010)
Ago2	CLIP-seq	mESC	Pre-miRNA seeds GCACUU and G-rich enhancers	Leung et al. (2011)
Msy2	CLIP	Mouse seminiferous tubules	piRNAs, other small RNAs	Xu et al. (2009)

Modified from Darnell (2010)

access repository of information for broader class of post-transcriptional regulatory elements. Predictive modeling tools (e.g., Wen et al. 2011) and databases (e.g., starBase: Yang et al. 2011) of miRNA–mRNA interaction maps have also been developed. Future studies will shed light on the interaction of miRNA and other RBPs in dynamic regulation of common mRNA targets.

References

Bailey, T., and Elkan, C. (1994). Fitting a mixture model by expectation maximization to discover motifs in biopolymers Paper presented at: Proc Int Conf Intell Syst Mol Biol (Menlo Park, California, AAAI Press).

Becht P, et al. (2006) The RNA-binding protein Rrm4 is essential for polarity in *Ustilago maydis* and shuttles along microtubules. *J Cell Sci,* 119:4964–4973.

Bohnsack MT, et al (2009) Prp43 bound at different sites on the pre-rRNA performs distinct functions in ribosome synthesis. *Mol Cell*, 36:583–592

Brooks, AN, et al. (2011) Conservation of an RNA regulatory map between Drosophila and mammals. Genome Res. 21:193–202.

Chi SW, et al. (2009) Argonaute HITS-CLIP decodes microRNA-mRNA interaction maps. Nature, 460:479–86.

Darnell RB. 2010. HITS-CLIP: panoramic views of protein-RNA regulation in living cells. *Wiley Interdiscip Rev RNA* 1:266–286.

Daughters RS, et al. (2009) RNA gain-of function in spinocerebellar ataxia type 8. *PLoS Genet*, 5:e1000600.

Granneman S, et al. (2009) Identification of protein binding sites on U3 snoRNA and pre-rRNA by UV cross-linking and high-throughput analysis of cDNAs. *Proc Natl Acad Sci USA*, 106: 9613–9618.

Guil S, Caceres JF.(2007) The multifunctional RNA-binding protein hnRNP A1 is required for processing of miR-18a. *Nat Struct Mol Biol*, 14:591–596.

Hafner M, et al. (2010) Transcriptome wide identification of RNA-binding protein and microRNA target sites by PAR-CLIP. *Cell*, 141:129–141.

Jensen KB, Darnell RB. CLIP: crosslinking and immunoprecipitation of *in vivo* RNA targets of RNA binding proteins. *Methods Mol Biol* 2008, 488:85–98.

Katz Y, Wang ET, Airold EM, Burge CB. (2010) Analysis and design of RNA sequencing experiments for identifying isoform regulation. *Nat methods*, 7:1009–15.

Kazan H, et al. (2010) RNAcontext: a new method for leaning the sequence and structure binding preferences of RNA-binding proteins. *PLoS Comput Biol* 6(7):e1000832.

Khorshid M, Rodak C, Zavolan M. (2011) CLIPZ: a database and analysis environment for experimentally determined binding sites of RNA-binding proteins. *Nucl Acid Res*, 39:D245–52.

König, J. et al. (2010) iCLIP reveals the function of hnRNP particles insplicing at individual nucleotide resolution. Nat. Struct.Mol.Biol. 17, 909–915.

Leung AK, et al (2011) Genome-wide identification of Ago2 binding sites from mouse embryonic stem cells with and without mature microRNAs. Nat Struct. Mol Biol 18:237–44.

Licatalosi DD. et al. (2008) HITS-CLIP yields genome-wide insights into brain alternative RNA processing. Nature 456, 464–469.

Licatalosi DD, Darnell RB. 2010. RNA processing and its regulation: global insights into biological networks. *Nat Rev Genet* 11(1): 75–87.

Pepke S, Wold B, Mortazavi A. (2009) Computation for ChIP-seq and RNA-seq studies. *Nat Methods*, 6(11 Suppl):S22–32.

Racca C, et al. (2010) The neuronal splicing factor Nova co-localizes with target RNAs in the dendrite. Front Neural Circuits, 4:5.

Sanford JR, et al. (2008) Identification of nuclear and cytoplasmic mRNA targets for the shuttling protein SF2/ASF. *PLoS ONE*, 3:e3369.
Sanford JR, et al. (2009) Splicing factor SFRS1 recognizes a functionally diverse landscape of RNA transcripts. *Genome Res*, 19:381–394.
Serganov A, Patel DJ. (2008). Towards deciphering the principles underlying an mRNA recognition code. *Curr Opin Struct Biol* 18:120–9.
Shendure J, Ji H. (2008) Next-generation DNA sequencing. *Nat Biotech*, 26:1135–45.
Smith, A.D., et al. (2005). Identifying tissue-selective transcription factor binding sites in vertebrate promoters. Proc Natl Acad Sci U S A *102*, 1560–1565.
Sumazin P, et al. (2005) DWE: Discriminating Word Enumerator – Bioinformatics 21(1):31–38
Thompson W, et al. (2005). Using the Gibbs motif sampler to find conserved domains in DNA and protein sequences. Curr Protoc Bioinformatics. Chapter 2:Unit 2.8.
Ule J, et al. (2003) CLIP identifies Nova-regulated RNA networks in the brain. *Science* 302: 1212–1215.
Ule J et al. (2006) An RNA map predicting Nova-dependent splicing regulation. *Nature* 444: 580–586.
Wen J, Parker BJ, Jacobsen A, Krogh A. (2011) MicroRNA transfection and AGO-bound CLIP-seq data sets reveal distinct determinants of miRNA action. *RNA*. 17:820–34.
Witten JT, Ule J. (2011) Understanding splicing regulation through RNA splicing maps. *Trends Genet*, 27:89–97.
Wold B, Myers RM. (2008) Sequence census methods for functional genomics. *Nat Methods*, 5:19–21.
Wolf JJ, et al. (2010) Feed-forward regulation of a cell fate determinant by an RNA-binding protein generates asymmetry in yeast. *Genetics*, 185:513–522.
Wu J, Akerman M, et al. (2011) SpliceTrap: a method to quantify alternative splicing under single cellular conditions. Bioinformatics, Sep 6 [Epub ahead of print] PMID: 21896509.
Wurtmann EJ, Wolin SL.(2010) A role for a bacterial ortholog of the Ro autoantigen in starvation-induced rRNA degradation. *Proc Natl Acad Sci USA*, 107:4022–4027.
Xiao X, Wang Z, et al. (2009) Splice site streng-dependent activity and genetic buffering by ply-G runs. *Nat Struct Mol Biol*, 16:1094–100.
Xu M, et al (2009) MIWI-independent small RNAs (MSY-RNAs) bind to the RNA-binding protein, MSY2, in male germ cells. *Proc Natl Acad Sci USA*, 106:12371–12376.
Xue Y, et al. (2009) Genome wide analysis of PTB-RNA interactions reveals a strategy used by the general splicing repressor to modulate exon inclusion or skipping. *Mol Cell*, 36:996–1006.
Yang et al (2011) starBase: a database for exploring microRNA-mRNA interaction maps from Argonaute CLIP_Seq and Degradome-Seq data. Nucl. Acids Res. 39(Database issue): D202–209.
Yano M, et al. (2010) Nova2 regulates neuronal migration through an RNA switch in disabled-1 signaling. *Neuron*, 66:848–858
Yeo GW, et al. (2009) An RNA code for the FOX2 splicing regulator revealed by mapping RNA-protein interactions in stem cells. Nat. Struc. Mol. Biol., 16:130–137.
Zhang C, et al. (2008) Defining the regulatory network of the tissue-specific splicing factors Fox-1 and Fox-2. Genes Dev 22:2550–63.
Zhang C, et al. (2010) Integrative model defines the nova splicing-regulatory network and its combinatorial controls. Science, 329:439–443.
Zhang C, Darnell RB. (2011). Mapping in vivo protein-RNA interactions at single-nucleotide resolution from HITS-CLIP data. *Nat Biotech*, 29:607–614.
Zhang C, Zhang MQ. (2011) Analysis of splicing variants by high-throughput sequencing. RNA Book on Splicing, Chapter 51: Wiley-VHC.
Zisoulis DG, et al. (2010) Comprehensive discovery of endogenous Argonaute binding sites in *Caenorhabditis elegans*. *Nat Struct Mol Biol*, 17:173–179.

Chapter 13
Analysis of Metagenomics Data

Elizabeth M. Glass and Folker Meyer

Abstract Improved sampling of diverse environments and advances in the development and application of next-generation sequencing technologies are accelerating the rate at which new metagenomes are produced. Over the past few years, the major challenge associated with metagenomics has shifted from generating to analyzing sequences. Metagenomic analysis includes the identification, and functional and evolutionary analysis of the genomic sequences of a community of organisms. There are many challenges involved in the analysis of these data sets including sparse metadata, a high volume of sequence data, genomic heterogeneity, and incomplete sequences. Because of the nature of metagenomic data, analysis is very complex and requires new approaches and significant compute resources. Recently, several computational systems and tools have been developed and applied to analyze their functional and phylogenetic composition.

The metagenomics RAST server (MG-RAST) is a high-throughput system that has been built to provide high-performance computing to researchers interested in analyzing metagenomic data. It has removed one of the primary bottlenecks in metagenome sequence analysis, the availability of high-performance computing for annotating data.

E.M. Glass
Mathematics and Computer Science Division, Argonne National Laboratory,
Argonne, IL 60439, USA

Computation Institute, The University of Chicago, 5735 South Ellis Avenue,
Chicago, IL 60637, USA

F. Meyer (✉)
Mathematics and Computer Science Division, Argonne National Laboratory,
Argonne, IL 60439, USA

Computation Institute, The University of Chicago, 5735 South Ellis Avenue,
Chicago, IL 60637, USA

Institute for Genomics and Systems Biology, The University of Chicago,
900 East 57th Street, Chicago, IL 60637, USA
e-mail: folker@anl.gov

N. Rodríguez-Ezpeleta et al. (eds.), *Bioinformatics for High Throughput Sequencing*,
DOI 10.1007/978-1-4614-0782-9_13,

13.1 Introduction

Studying uncultivable microorganisms has been a major obstacle to understanding natural microbial populations within the context of their environment. Metagenomics is expanding quickly, as next-generation sequencing approaches become more widespread and applied to an increasing number of environments. It has bypassed the need for cloning and has enabled a new approach to comparative metagenomics (Ronaghi et al. 1996, 1998; Margulies et al. 2005). Now sequence abundance can be used to contextualize datasets for driving pattern recognition and uncovering unique properties within natural microbial communities.

Regardless of the sequencing approach used to generate data, the first steps in analysis of any metagenome involve comparative analysis against various ribosomal and protein and nucleotide databases. These comparisons have a large computational cost but provide the basic data types for many subsequent analyses, including phylogenetic comparisons, functional annotations, binning of sequences, phylogenomic profiling, and metabolic reconstructions and modeling. Analysis of single metagenomes can provide a greater understanding of a microbial community, but the comparison of multiple metagenomes provides greater insight.

Sequence data, however, must be accompanied by enough contextual information (metadata), such as sample characteristics, to make individual investigations reproducible and enable valid interpretation (Field et al. 2009). Community-driven minimum information checklists (Taylor et al. 2008), common ontologies (Smith et al. 2007), and formats (Jones et al. 2007; Sansone et al. 2008) have major roles to play. Therefore, data describing such information as a sample's environment, sample origin, isolation, and treatment are an important resource to link to sequence data in order to enable meaningful comparative analysis. The Genomics Standards Consortium (GSC) has defined the Minimum Information About a (Meta)Genome Sequence (MIGS/MIMS) (Kottmann et al. 2008), which describes core descriptors of environmental context (habitat). MIGS/MIMS extends the minimum information provided by the International Nucleotide Sequence Database Collaboration (INSDC) (Cochrane et al. 2011).

Recently, several computational systems and tools have been developed and applied to analyze their functional and phylogenetic composition. One such system, MG-RAST (Meyer et al. 2008), is available over the web to researchers, and access is not limited to specific groups or data types. This system has a scalable compute backend that has enabled the analysis of over 10,000 metagenomes (as of January 2011).

13.2 Metagenomic Analysis

Metagenomic analysis is not straightforward. The data is much more complex than what has previously been seen in genomics. Metagenomic sequence data have lower sequence redundancy, lower sequence quality, short read lengths, increased polymorphisms, and relative abundance (simple vs. complex communities). In addition to these inherent issues and the evolution of sequencing technologies and chemistries,

the size of the data is changing. The scientific community has already seen the size of these data sets quickly move from Megabase pairs (Mbps) to Gigabasepairs (Gbps) and now Terabases, which require significant compute resources.

Given the sufficient compute resources, there are several different approaches that can be taken with raw sequence reads. The analysis "path" and the tools you choose can influence your results. There is no "one size fits all" tool or best practice established for analyzing metagenomic data sets. Various approaches have strengths and weaknesses and are constantly evolving.

However, the major metagenomic analysis pipelines such as MG-RAST, IMG/M (Markowitz et al. 2008), and CAMERA (Sun et al. 2011) provide compelling analysis strategies and features as well as distinct implementations of common operations. The MG-RAST server is the most widely used tool for the analysis of shotgun metagenomics and provides a basis for sequence analysis of large, complex data sets. Over 4,000 users have submitted data sets and several hundred users work on the system each day.

The MG-RAST system accepts shotgun metagenomic DNA sequence data in different formats and from a variety of platforms, providing initial quality control and normalization of the data. The pipeline also accepts assembled sequences in fasta format. Sequence data may be compressed by one of several common computer programs to speed upload. Users may choose to upload raw unassembled reads or assembled contigs. The system also provides a GSC compliant metadata editor to enter relevant information about a sample. This information is then incorporated into the analysis and querying capabilities. The server provides several methods to access different data types, including phylogenetic and metabolic reconstructions, and the ability to compare the metabolism and annotations of one or more metagenomes and genomes. In addition, the server offers browsing of data and a comprehensive search capability. Access to the data is password protected, and all data generated by the automated pipeline are available for download and analysis in variety of common formats. One of the more widely used features is the ability to share data prior to publication, leading to networks of shared data sets.

13.2.1 Metadata

It is apparent that the full potential of comparative metagenome analysis can be achieved only in the context of the metadata (information describing the sample). The selection of samples based on rich metadata is crucial for understanding large-scale patterns when multiple metagenomes are compared. The GSC has proposed a minimal set of data, called the Minimum Information about a (Meta)Genome Sequence (MIGS/MIMS) that should be collected with every metagenome sequence. Although this is an evolving standard, the MG-RAST server is MIGS/MIMS-compliant. Metadata is requested from the user at the time of sequence submission to MG-RAST. Metadata can be added to at any point after submission and a minimal set is required for sharing or publishing (making public). This data is stored with the user's data and is made available to them.

13.2.2 Preprocessing

Preprocessing of sequence reads before analysis (assembly, gene prediction, and annotation) is an overlooked aspect of metagenomic analysis. Preprocessing includes steps in filtering data for vectors, host contaminants, and quality trimming. Mistakes in any of these steps can have significant downstream affect on analyses (Koonin 2007). MG-RAST employs a normalization step, generating unique internal IDs, and removing duplicate sequences. Users can select filtering for contaminants. It also includes a runtime-efficient method for obtaining a quality estimate for each sample and removal of sequencing artifacts.

13.2.3 Identifying Genes

Sequence length is an important factor in determining an approach to gene calling. Shorter reads' lengths pose an obvious and significant challenge. The most commonly used method for identifying genes in metagenomic reads is via similarity searches using metagenomic sequences against databases of known proteins. BLAST (Altschul et al. 1997) has become too costly in terms of computation. Faster alternatives such as BLAT (Kent 2002), doing assembly and feature prediction, greatly reduce the computational burden of comparing all pairs of short reads. MG-RAST relies on BLAT to perform sequence similarity searches as it provides significant speed-ups over BLAST, offers very similar results, and looses little sensitivity in our tests.

MG-RAST screens for potential protein-encoding genes (PEGs) via a BLAT search against the MG-RAST nonredundant database. This strategy will reveal already known genes that are present in the metagenome. A drawback to using this approach as a sole means to identify genes is that many genes are most probably not present in the databases because of the bias toward culturable organisms. Therefore, MG-RAST performs feature prediction using FragGeneScan (Rho et al. 2010), before running similarity searches. FragGeneScan predicts coding regions in sequences that are greater than or equal to 80 bp.

In parallel with feature prediction and BLAT similarity searches against the protein database, the sequence data is also compared to other databases by using the appropriate algorithms and significant selection criteria. These databases include several ribosomal databases, including GREENGENES (DeSantis et al. 2006), RDP-II (Cole et al. 2007), and Silva (Pruesse et al. 2007). The search criteria are specific for each database. For example, using Sblat against the rDNA databases enables users to screen for ribosomal RNA genes, but much more stringent selection criteria are used to identify candidate RNA genes than for identifying protein-encoding genes (by default, the similarity must exceed 50 bp in length and have an expect value less than $1 \times 10–5$). Lastly, these matches to the MG-RAST database and ribosomal databases are used to compute the derived data. A phylogenomic reconstruction of the sample is computed by using both the phylogenetic information contained in the

non-redundant database and the similarities to the ribosomal RNA databases. Functional classifications of the PEGs are computed by projecting against protein functional annotations based on these similarity searches. These annotations become the raw input to an automatically generated initial metabolic reconstruction of the sample, as well as subsequent metabolic model for the sample by providing suggestions for metabolic fluxes and flows, reactions, and enzymes.

While the existing version relied on sequence comparison with the non-redundant database provided by the SEED (Overbeek et al. 2005) and SEED subsystems solely, the new version is based on a database emanating from the Genomics Standards Consortiums M5 platform. This non-redundant database provides a non-redundant integration of many databases [e.g., INSD, SEED, IMG, KEGG (Kanehisa et al. 2004), and EGGNOGs (Jensen et al. 2008)], thus allowing supporting multiple different views on the data with one similarity search.

13.2.4 Multiple Supported Classification Schemes

A number of competing naming schemes to describe functional classification of genes and proteins exist. While the use of consistent SEED subsystem-based annotations provides many advantages, other databases provide different functional hierarchies (e.g. SEED subsystems, IMG, COG/NOGs) or ontologies [GO (Barrell et al. 2009)]. Enabled by this protein database, we provide the ability to "on-the-fly" switch between different annotation resources. Allowing users to view their data through mapping to different classification schemes enables them to tease out differences or similarities between metagenomic data sets not visible otherwise.

The user interface for MG-RAST was designed to provide easy navigation and use of comparative tools. There are multiple views for browsing and analysis of the data, as well as a means to download all result tables and the sequences for every subset displayed. Users are also enabled to modify the displayed results by modifying search parameters used to compute the functional, metabolic, and phylogenetic reconstruction. This allows more stringent match criteria (e.g., expectation value, score, overall percent identity, length of match, and number of mismatches) and by restricting the matches, the derived data is dynamically changed. The default parameters have been chosen by empirical testing and represent a tradeoff between accuracy and specificity.

13.2.5 Annotations

Users can view and search their annotated metagenome based on annotation source (see description of MG-RAST database) through various avenues. Metagenome Overview provides a summary of the sequence and annotation statistics against the various databases. More details are presented in the Sequence Profiles, which

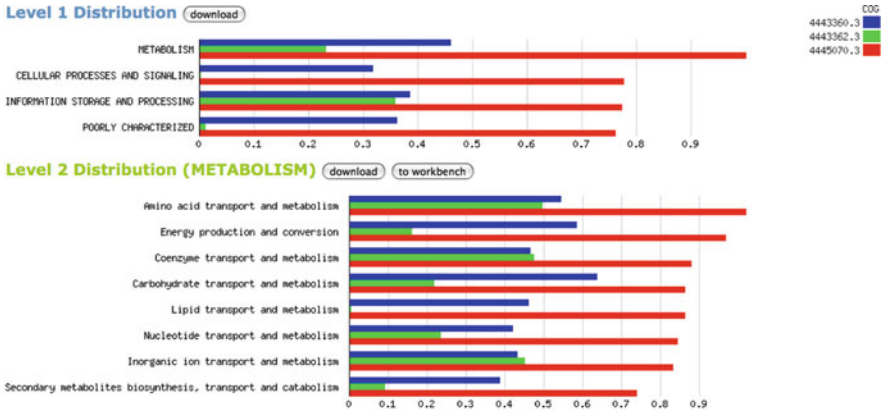

Fig. 13.1 Sequences are compared to the MG-RAST protein database that provides a non-redundant integration of many databases (INSDC, SEED, IMG, KEGG, and eggNOGs), supporting many complementary views into the data with one similarity search. Shown are the functional distributions based on COG annotations

display the metabolic and phylogenetic distributions in a given sample. Views are in the form of charts and tables and data is downloadable for each profile (Fig. 13.1). Like all analyses in MG-RAST, the user can modify inclusion parameters and export results. Each metabolic or phylogenetic/phylogenomic profile can also be viewed singularly or compared with other metagenomes using a circular tree comparison tool (Fig. 13.2).

13.2.6 Comparative Metagenomics

Considering that comparative analysis is the core driver for discovery-based biology, MG-RAST enables more than just views of the analysis results of a given metagenome, the system supports comparative analysis. Therefore, comparative metagenomics tools are central to the utility of the MG-RAST platform. Several tools have been developed and integrated into the MG-RAST framework, allowing users to compare a metagenome to either (1) other metagenomes, (2) individual genomes, or (3) both metagenomes and genomes.

13.2.6.1 Comparative Heat Maps

Metabolic: PEGs identified to have functions belonging to a SEED subsytem(s) and KEGG pathways are mapped to that subsystem/pathway. When these functional roles are linked to specific genes across metagenomes and a populated subsystem emerges. The utility of this organization is extended by subsystem connections that allow linkage of genes between subsystems.

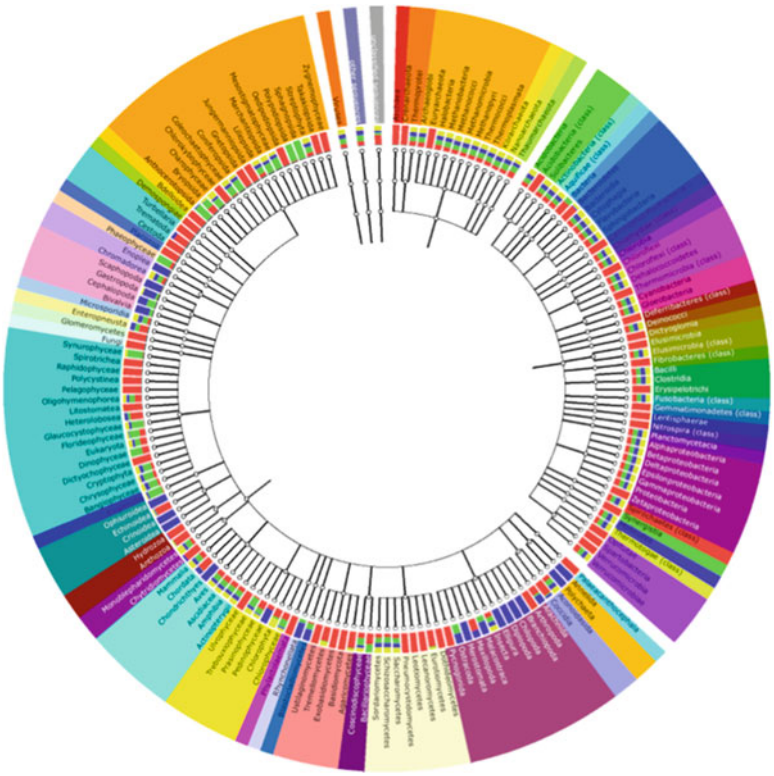

Fig. 13.2 An example of a comparative view in MG-RAST. A circular tree representing phylogenetic profiles from four samples is compared. Each node can be expanded to get detailed information about the distribution for each sample. Color shading of the family names indicates class membership

Each subsystem present in a sample is scored by counting the number of sequences that are similar to a protein in each subsystem. This score is divided by the total number of sequences from the sample that are similar to any protein in a subsystem, to give a fraction of sequences in subsystems that are in a given subsystem. This approach allows comparisons between samples that have different numbers of sequences. Because the fractions tend to be small, the scores can be factored for display purposes. Moreover, the display can be limited or expanded to include various levels in the subsystem hierarchy, to specific areas of metabolism, or other subsystem groups, as chosen by the user.

Phylogenetic: The taxonomic heat map works in an analogous fashion but highlights the different taxonomic profiles in each sample, as determined by the phylogenetic or phylogenomic approaches selected by the end user (e.g., 16S comparisons, phylogenomics from BLAT results). Again, samples may be grouped in a nonquantitative fashion to rapidly highlight particular phylogenetic groups that predominate in different samples.

13.2.6.2 Principal Component Analysis

Many comparative analyses use multivariate statistics when several metagenomic datasets are involved, or when several types of factors are thought to affect the observed compositions of the communities. MG-RAST has incorporated an R-based PCA (principal component analysis) to its suite of comparative tools.

13.2.6.3 Recruitment Plot

The recruitment plot tool is set up to provide a selected sequenced microbial genome as a scaffold to map metagenome-derived sequences. As in the heat map, sequences that have been annotated from a metagenome are used as the queries. The initial view provides a ranked list of microbial genomes that contain the most number of matched sequences from the metagenome. This gives an indication of the relative representations in terms of genomic content found within the metagenome.

13.2.7 Metabolic Reconstructions and Models

Metagenomics also has the potential to provide insights into the critical biochemical mechanisms in each environment. Models in the MG-RAST are based on the initially assembled metabolic reconstructions. The functional roles from the reconstruction are then mapped to reactions in the SEED and KEGG biochemistry databases, and this mapping is used to assemble a reaction list for the model. Models are based on a steady state and undergo flux balance analysis.

13.3 Results and Discussion

Improved sampling of diverse environments, combined with the advances in the development and application of next-generation sequencing technologies, is accelerating the pace at which new metagenomes are generated. In fact, the amount of sequence data being produced will quickly outpace the ability of scientists to analyze it. Analysis of metagenomic data needs to incorporate scalable computing resources.

The process of building MG-RAST is the result of several years of planning and engineering. The system provides integration of metagenome data, microbial genomics, and manually curated annotations. The metagenomics analysis pipeline was designed to allow for interactive analysis and the system as a whole has been built by using an extensible format allowing the integration of new datasets and algorithms without a need for recomputation of existing results. The system has been restructured to be scalable. This means MG-RAST uses cloud computing, which decouples it from a particular dataset and allows vast compute resources, to conduct the analysis.

The MG-RAST server handles both assembled and unassembled data. Each approach has advantages that should be considered when comparing metagenomes. For example, a case where sequences should be assembled is when comparisons between samples are being calculated, as the assembly process loses the frequency information critical for determining differences between samples. In contrast, assembled sequences tend to be longer and therefore more likely to accurately identify gene function or phylogenetic source from binning (McHardy et al. 2007).

The analytical methods integrated into MG-RAST provide core annotations and analysis tools to compare and contrast sets of metagenomes (Edwards et al. 2006; Fierer et al. 2007; Mou et al. 2008). The approach underlying the subsystems-based functional analysis of metagenomes has been validated with 90 different samples from nine major biomes. The analysis demonstrated that the biomes could clearly be separated by their functional composition (Dinsdale et al. 2008). All of the metagenomes present in that study are included in the publicly available datasets visible on the MG-RAST server.

Although the service contains core functionality for the annotation and analysis of metagenomes, many of the techniques traditionally used for genome analysis either do not work with metagenomes or show significant performance degradation (Krause et al. 2006). Therefore, new analytical methods are needed to fully understand metagenomics data. The most obvious problem is with the large number of unknown sequences in any sample. Others and we are developing new binning, clustering, and coding region prediction tools to handle these unknown sequences, and effective tools will be incorporated into the pipeline when available. Another problem is that the rapid pace with which sequence data is being generated outpaces increases in computational speed, and therefore improvements in common search algorithms are required to ensure that sequence space can be accurately and efficiently searched.

13.4 Internet Resources

MG-RAST (http://metagenomics.anl.gov)

Acknowledgments This work was supported by the US Department of Energy, under Contract DE-AC02-06CH11357.

References

Altschul SF, Madden TL, Schaffer AA, Zhang J, Zhang Z, et al. 1997. Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. Nucleic Acids Res. 25: 3389–3402.

Barrell D, Dimmer E, Huntley RP, Binns D, O'Donovan C, Apweiler R. The GOA database in 2009 – an integrated Gene Ontology Annotation resource. Nucleic Acids Res. 2009 Jan;37(Database issue):D396–403.

Cochrane G, Karsch-Mizrachi I, Nakamura Y; International Nucleotide Sequence Database Collaboration. The International Nucleotide Sequence Database Collaboration. Nucleic Acids Res. 2011 Jan;39(Database issue):D15–8.

Cole JR, Chai B, Farris RJ, Wang Q, Kulam-Syed-Mohideen AS, et al. 2007. The ribosomal database project (RDP-II): introducing myRDP space and quality controlled public data. Nucleic Acids Res. 35(Database issue)D:169-72.

DeSantis TZ, Hugenholtz P, Larsen N, Rojas M, Brodie EL, et al. 2006. Greengenes, a chimera-checked 16S rRNA gene database and workbench compatible with ARB. Appl Environ Microbiol. 72(7):5069-72.

Dinsdale EA, Edwards RA, Hall D, Angly F, Breitbart M, et al. 2008. Functional metagenomic profiling of nine biomes. Nature. 452(7187):629–32.

Edwards RA, Rodriguez-Brito B, Wegley L, Haynes M, Breitbart M, et al. 2006. Using pyrosequencing to shed light on deep mine microbial ecology. BMC Genomics. 7:57.

Field D, Sansone SA, Collis A, Booth T, Dukes P, Gregurick SK, et al. Megascience. Omics data sharing. Science. 2009;326(5950):234–236.

Fierer N, Breitbart M, Nulton J, Salamon P, Lozupone C, et al. 2007. Metagenomic and small-subunit rRNA analyses reveal the genetic diversity of bacteria archaea, fungi, and viruses in soil. Appl Environ Microbiol. 73(21):7059–7066.

Jensen LJ, Julien P, Kuhn M, von Mering C, Muller J, Doerks T, Bork P. eggNOG: automated construction and annotation of orthologous groups of genes. Nucleic Acids Res. 2008 Jan;36(Database issue):D250–4.

Jones AR, Miller M, Aebersold R, Apweiler R, Ball CA, Brazma A, Degreef J, Hardy N, Hermjakob H, Hubbard SJ, Hussey P, Igra M, Jenkins H, Julian RK Jr, Laursen K, Oliver SG, Paton NW, Sansone SA, Sarkans U, Stoeckert CJ Jr, Taylor CF, Whetzel PL, White JA, Spellman P, Pizarro A. (2007). The Functional Genomics Experiment model (FuGE): an extensible framework for standards in functional genomics. Nat Biotechnol. 25(10):1127–1133.

Kanehisa M, Goto S, Kawashima S, Okuno Y, Hattori M. 2004. The KEGG resource for deciphering the genome. Nucleic Acids Res. 1;32(Database issue):D277–80.

Kent WJ. BLAT – the BLAST-like alignment tool. Genome Res. 2002 Apr;12(4):656–64.

Koonin EV. Metagenomic sorcery and the expanding protein universe. Nat Biotechnol. 2007 May;25(5):540–2.

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

Krause L, Diaz NN, Bartels D, Edwards RA, Puhler A, et al. 2006. Finding novel genes in bacterial communities isolated from the environment. Bioinformatics. 22(14):e281–289.

Margulies M, Egholm M, Altman WE, Attiya S, Bader JS, et al. 2005. Genome sequencing in microfabricated high-density picolitre reactors. Nature 437: 376–380.

Markowitz VM, Ivanova NN, Szeto E, Palaniappan K, Chu K, Dalevi D, Chen IM, Grechkin Y, Dubchak I, Anderson I, Lykidis A, Mavromatis K, Hugenholtz P, Kyrpides NC. IMG/M: a data management and analysis system for metagenomes. Nucleic Acids Res. 2008 Jan;36(Database issue):D534–8.

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

Meyer F, Paarmann D, D'Souza M, Olson R, Glass EM, Kubal M, Paczian T, Rodriguez A, Stevens R, Wilke A, Wilkening J, Edwards RA. The metagenomics RAST server – a public resource for the automatic phylogenetic and functional analysis of metagenomes. BMC Bioinformatics. 2008 Sep 19;9:386.

Mou XSS, Edwards RA, Hodson RE, Moran MA. 2008. Bacterial carbon processing by generalist species in the coastal ocean. Nature. 451(7179):708–711.

Overbeek R, Begley T, Butler RM, Choudhuri JV, Chuang HY, et al. 2005. The subsystems approach to genome annotation and its use in the project to annotate 1000 genomes. Nucleic Acids Res. 7;33(17):5691–702.

Pruesse E, Quast C, Knittel K, Fuchs BM, Ludwig W, et al. 2007. SILVA: a comprehensive online resource for quality checked and aligned ribosomal RNA sequence data compatible with ARB. Nucleic Acids Res. 35(21):7188–7196.

Rho M, Tang H, Ye Y. FragGeneScan: predicting genes in short and error-prone reads. Nucleic Acids Res. 2010 Nov 1;38(20):e191.

Ronaghi M, Uhlen M, Nyren P. 1998. A sequencing method based on real-time pyrophosphate. Science 281: 363, 365.

Ronaghi M, Karamohamed S, Pettersson B, Uhlen M, Nyren P. 1996. Real-time DNA sequencing using detection of pyrophosphate release. Anal Biochem 242: 84–89.

Sansone SA, Rocca-Serra P, Brandizi M, Brazma A, Field D, Fostel J, Garrow AG, Gilbert J, Goodsaid F, Hardy N, Jones P, Lister A, Miller M, Morrison N, Rayner T, Sklyar N, Taylor C, Tong W, Warner G, Wiemann S. (2008). The First MGED RSBI (ISA-TAB) Workshop: "Can a Simple Format Work for Complex Studies?". *OMICS*. 12(2):143–149.

Smith B, Ashburner M, Rosse C, Bard J, Bug W, Ceusters W, Goldberg JL, Eilbeck K, Ireland A, Mungall CJ, Leontis N, Rocca-Serra P, Ruttenberg A, Sansone SA, Shah N, Whetzel PL, Suzanna L. (2007). The OBO Foundry: Coordinated evolution of ontologies to support biomedical data integration. Nat Biotechnol. 25(11):1251–1255.

Sun S, Chen J, Li W, Altintas I, Lin A, Peltier S, Stocks K, Allen EE, Ellisman M, Grethe J, Wooley J. Community cyberinfrastructure for Advanced Microbial Ecology Research and Analysis: the CAMERA resource. Nucleic Acids Res. 2011 Jan;39(Database issue):D546–51.

Taylor CF, Field D, Sansone SA, Aerts J, Apweiler R, Ashburner M, Ball CA, Binz PA, Bogue M, Booth T, Brazma A, Brinkman RR, Michael Clark A, Deutsch EW, Fiehn O, Fostel J, Ghazal P, Gibson F, Gray T, Grimes G, Hancock JM, Hardy NW, Hermjakob H, Julian RK Jr, Kane M, Kettner C, Kinsinger C, Kolker E, Kuiper M, Le Novère N, Leebens-Mack J, Lewis SE, Lord P, Mallon AM, Marthandan N, Masuya H, McNally R, Mehrle A, Morrison N, Orchard S, Quackenbush J, Reecy JM, Robertson DG, Rocca-Serra P, Rodriguez H, Rosenfelder H, Santoyo-Lopez J, Scheuermann RH, Schober D, Smith B, Snape J, Stoeckert CJ Jr, Tipton K, Sterk P, Untergasser A, Vandesompele J, Wiemann S. (2008) MIBBI: A Minimum Information Checklist Resource. Nat Biotechnol 26(8):889–996.

Chapter 14
High-Throughput Sequencing Data Analysis Software: Current State and Future Developments

Konrad Paszkiewicz and David J. Studholme

Abstract In previous chapters of this book, there is detailed treatment of the technicalities of such problems as de novo sequence assembly and sequence alignment. In this chapter, we take a different perspective. Drawing on nearly a decade of the authors' collective experience in providing bioinformatics support to bench-based biologists, we focus on the practical applications and on the biologist end-user's experience. We attempt to make some observations, speculations and recommendations that might help the "wet" biologist who wishes to take responsibility for dealing their own data.

14.1 Introduction/Pre-amble

Biology is becoming increasingly data-rich science. This is especially true in recent years with the advent of high-throughput technologies such as proteomics, metabolomics and second-generation sequencing. With the ever-increasing performance of second-generation sequencers and the advent of the third generation of technologies, this trend is set to continue. Data-rich science demands that computational techniques and tools become as central to molecular biology as aseptic technique, restriction digests and electrophoresis have been in the last few decades. Typically the bottleneck in many high-throughput projects is no longer the data generation but the data analysis. It is not the actual computation time that is the bottleneck, but rather the availability of the bioinformatician who needs to make decisions and manual interventions in the course of the analysis, integration with other datasets

K. Paszkiewicz • D.J. Studholme (✉)
School of Biosciences, University of Exeter, Geoffrey Pope Building,
Stocker Road, EX4 4QD Exeter, UK
e-mail: d.j.studholme@exeter.ac.uk

N. Rodríguez-Ezpeleta et al. (eds.), *Bioinformatics for High Throughput Sequencing*,
DOI 10.1007/978-1-4614-0782-9_14,

and visualisation of the results. Unfortunately, the undergraduate and post-graduate training of biologists has not yet caught up with this new landscape. Similarly, the computational tools have not yet caught up. Despite the impressive array of clever algorithms and neat solutions available to data analysis problems, effective use of these tools usually still requires a level of computer-literacy that is beyond most laboratory-based biologists. There are relatively few individuals who are highly proficient in both "wet" laboratory-based disciplines and "dry" computational methods. This gap between the computational skills of the biologist and the usability of the software needs to be closed, and probably will over time. But that could take decades and we need to deal with a deluge of data right now. So in the meantime, many biologists will need to call on the help of professional specialists in applied bioinformatics and/or acquire some core computational competencies right now.

In previous chapters of this book, there is detailed treatment of the technicalities of such problems as de novo sequence assembly and sequence alignment. In this chapter, we take a different perspective. Drawing on nearly a decade of the authors' collective experience in providing bioinformatics support to bench-based biologists, we focus on the practical applications and on the biologist end-user's experience. We attempt to make some observations, speculations and recommendations that might help the "wet" biologist who wishes to take responsibility for dealing their own data.

14.2 Overview of Software Available for NGS

There are hundreds of software packages available that provide some aspect of analysing second-generation sequence data. The SeqAnswers website (2011) list nearly 400. Most of these packages are aimed at a specific task or step in a sequence-analysis workflow. For example, there is a bewildering plethora of software tools available for aligning a set of short sequence reads (e.g. from the Illumina GA2 or ABI SOLiD platforms) against a reference genome sequence. Which one should we use? Not surprisingly, there is no simple and straightforward answer to this question. Each package has its own unique strengths and weaknesses. For example, the majority of the alignment tools are only able to handle letter-space data and cannot deal with SOLiD's colourspace data. However, SHRiMP (Rumble et al. 2009) and SOCS (Ondov et al. 2008) were amongst the first freely available tools for mapping SOLiD data against a reference genome. Similarly, not all alignment methods take into account quality scores of the base-calls in the query sequences; MAQ (Li and Durbin 2008) was the first to offer this feature. Furthermore, many of the alignment programs offer extra features in addition to alignment; e.g. MAQ was one of the first to offer an integrated SNP-caller and useful reports on alignment statistics. On the other hand, the initial releases of MAQ could only perform ungapped alignment, which was a serious limitation when aligning intron-free cDNA sequences against genomic DNA containing introns. Subsequently, many alignment-based tools offered solutions for gapped alignment (e.g. BWA, Li and Durbin 2009; NovoAlign, http://novocraft.com)

and identification of intron–exon boundaries in RNA-seq data (TopHat, Trapnell et al. 2009; MapSplice, Wang et al. 2010). However, gapped alignment inevitably comes at the expense of speed and usually does not significantly improve sensitivity. Another confounding factor is the fact that many of these packages are actively maintained and developed. This is something of a double-edged sword. The progressive improvements mean that nothing stays still. Whereas, 2 years ago it might have been straightforward to recommend MAQ the ungapped alignment tool of choice, MAQ is now obsolete and replaced by BWA (Li and Durbin 2009). Also, many of the other tools have acquired new features and/or improved speed, sensitivity and accuracy since their original release. Published papers describing these tools are usually out of date almost as soon as they are published. New improvements developed after acceptance for publication will, of course, not be absent from the manuscript. The most up-to-date information about a software package should normally be found in its user manual and its project website.

14.3 The Two Types of Analysis: Alignment and De Novo Assembly

From a data-analysis point of view, second-generation sequencing projects fall into two types: those centred around de novo assembly and those based around alignment. Some projects will, of course, use a combination of both approaches. For example, in comparative genomics there will often be a shared core component of the genome that can be tackled through alignment of sequence reads against a reference sequence but may also be a variable section of the genome that is completely absent from the reference sequence and so must be tackled through de novo assembly. Even in a purely de novo genome-sequencing project, there is often a requirement to align the original sequence reads against the assembly for quality control purposes, as the assembly software does not keep track of the positions of each read.

14.3.1 Alignment-Based Analysis of Second-Generation Sequence Data

Alignment of multiple sequence reads against a reference sequence comprises the key step in many applications of second-generation sequencing. Once genomic DNA reads are aligned against a reference genome sequence, it is possible to infer variants such as SNPs, CNVs, insertions and deletions. From an alignment of RNA-Seq reads, mRNA abundance can be inferred for quantitative expression profiling. Furthermore, the alignment may be used to infer splice-variants and a transcription start sites or to discover novel transcripts. Alignments of ChIP-Seq or Methyl-Seq data against a genome can be mined for peaks of coverage that indicate protein-binding sites or

DNA methylation sites. With new sequencing platforms generating as much as gigabases of sequence per day, the traditional workhorses of sequence alignment such as FASTA, BLAST and BLAT are just not fast enough and neither are they optimised for dealing with short sequence reads and therefore may have poor sensitivity. Fortunately, in the last 4 years many researchers and developers have turned their attention to this and created and released excellent tools for aligning second-generation sequence datasets against reference sequences. Most of these methods are characterised by the creation of an auxiliary data structure called an index to speed-up the alignment. The index usually consists of either a hash table (also found in BLAST) or a suffix tree.

In principle, it should be possible to combine any chosen alignment method with any software for the subsequent steps (variant discovery, abundance quantification, peak finding, etc.). This relies on the alignments being produced in a standard format that is readable by the downstream software. The emerging de facto standard file-format for alignments is SAM, along with its binary compressed derivative, BAM. These were made popular by the 1000 Genomes Project. When selecting an alignment tool, we would very strongly recommend choosing one that complies with this standard format to facilitate modularity in the evolving workflow. Good examples include BWA (Li and Durbin 2009) and Bowtie (Langmead et al. 2009).

As a consequence of the frenzy of activity in developing algorithms over the last few years, alignment of short reads against a reference is rarely a major bottleneck in analysis nowadays. Given a modern workstation equipped with 2 Gb RAM, one can typically align a few million Illumina reads against a bacterial genome in a matter of minutes using tools such as BWA or Bowtie. To align tens of millions of reads against a human genome would require more RAM (8 Gb should be sufficient) and might take on the order of hours rather than minutes.

Much of the recent activity in this field has been aimed at optimising the alignment of short reads, typically 30–100 nt, generated by Illumina and SOLiD platforms. However, in coming years, longer reads will make something of a comeback. Not only does the 454 GS-FLX already generate reads of several hundred nucleotides in length, but the read-length of the other technologies is steadily increasing. Similarly, the next wave of sequencing technologies, the so-called third generation, will likely offer relatively long reads, perhaps several kilobases in length. Currently, most of the alignment packages designed for second-generation sequence data do not cope well with reads longer than about 200 nt. The notable exceptions are BWA and Mosaik, which can both align 454 and Sanger reads as well as short reads.

14.3.2 Variant Detection

One of the most common applications of second-generation sequencing is the resequencing of genomes in order to identify SNPs and other sequence variants among individuals of the same species. For example, the discovery of new SNPs can be fed into genome-wide association studies aimed at discovering the genetic basis

for diseases and other phenotypes in humans. Variants can also be invaluable for characterising diseases, such as cancers, where the particular catalogue of SNPs and other variants can be used for a more precise diagnosis that may have an impact on the choice of therapy. A patient's genotype, as indicated by its SNPs will likely be increasingly used to inform therapeutic intervention in the age of personalised medicine. SNPs can also be enormously valuable in assisting breeding programmes for crop improvement and as epidemiological markers for monitoring the spread of microbial pathogens.

Given deep coverage of a reference sequence by aligned second-generation sequence reads, it is relatively straightforward to detect consistent discrepancies between the sequence reads and the reference sequence and there are many useful tools available for doing this including SAMtools (Li et al. 2009). Reliable inference is more difficult when coverage is shallow. The situation is even further complicated if the genotype in question is non-homozygous or contains multiple paralogous genes whose sequences are very similar but non-identical. Malhis and Jones (2010) Slider package uses a Bayesian approach to incorporate previous knowledge of SNPs as priors to optimise SNP-calling with low-coverage data. VarScan (Koboldt et al. 2009) was specifically designed to infer variants from pooled samples, where there is likely genetic heterogeneity among individuals in the pool; but there are still opportunities in this area to develop methods that reliably distinguish between rare variants and sequencing errors. Furthermore, we are still clearly a long way from software that can be used routinely by medical practitioners in a clinical (rather than academic research) setting.

14.3.3 De Novo Sequence Assembly

For the biologist faced with assembling real data, which of the many available software tools is the "best"? Any answer to that question will soon become out of date; this is an active field and existing software is continually being improved whilst new programs are being developed. The key issues will, however, be usability and quality of the final assembly. Several factors contribute to usability, including hardware and software requirements, ease of installation and execution, as well as speed. The quality of an assembly has two dimensions: contiguity (lengths of the contigs or scaffolds) and accuracy. Cultural issues may also be important. For example, the level of support available from online forums and mailing lists.

To the best of our knowledge, there has been no comparative survey of assembly tools by a "neutral" researcher. Such a survey could be a challenging task. Outcomes might depend on properties of a specific dataset. Some programs perform better on some datasets than on others. Each program differs in respect of how it resolves errors and inconsistencies in the data. Algorithms based on the de Bruijn graph, e.g. Velvet (Zerbino and Birney 2008), are highly sensitive to choice of k-mer size. This means that, even after having chosen which software to use, it is equally important to choose the optimal parameter values. There is an urgent need for a comprehensive

comparison (or competition) between the candidates on a suitably broad selection of datasets. Such a study should utilise a range of different datasets varying in factors such as size, error-rate, heterozygosity, repeat structure, sequence complexity, sampling bias, read lengths, insert lengths (for paired reads or mate pairs), etc. Rather than being a single once-off study revealing a snapshot of the situation at a given time, ideally the comparison should be continually updated as new software is released. Once set up initially, updates to such a comparison database could be largely automated.

Some guidelines can be made based on the published literature as well as our own experience. For assembling a whole-genome dataset, from a single insert-length library, Velvet would be a good choice, especially for small genomes up to about 40 Mb. Simpson et al. (2009) reported that ABySS, Velvet and EULER-SR performed much better than SSAKE and Edena on Illumina reads from a 5-Mb bacterial genome. However, Velvet generally runs much faster and with a smaller memory footprint than ABySS for relatively small datasets (e.g. bacterial genomes). On Illumina datasets from five microbial genomes, Velvet gave longer scaffolds and greater accuracy than EULER-SR (Simpson et al. 2009). ALLPATHS2 yielded significantly more contiguous and more accurate assemblies but only when provided with multiple DNA libraries with differing insert lengths (Maccallum et al. 2009). SOAP de novo produced more contiguous and more complete assemblies of a human genome than did ABySS and also produced better assemblies than ABySS, Velvet, EULER-SR, SSAKE and Edena on a bacterial genome (Li et al. 2010). The contiguity of assemblies by QSRA on small genomes were comparable with Velvet, but no data were provided on their accuracy (Bryant et al. 2009). Large memory requirements mean that assembly of non-hierarchical reads from large (e.g. mammalian) genomes is only practically feasible using a parallelisation strategy such as that of ABySS. However, a better solution might be to generate hierarchical sequence data from such genomes, as exemplified by (Sundquist et al. 2007; Hiatt et al. 2010; Young et al. 2010; Sorber et al. 2008), though these methods are more laborious.

De novo assembly is confounded by repetitive elements, low coverage and sequencing errors. Problems of low coverage and/or sequencing errors can usually be overcome by additional sequencing and stringent filtering. Repetitive elements therefore tend to be the greatest hurdle to achieving a good assembly. Where long repetitive stretches of the genome occur, unless reads or paired-end inserts are able to span the length of the repeat, the assembly will remain repeat limited. Typically the best method here is to use paired-end reads with both long (e.g. 10 kb) and short inserts (e.g. 400 bp). Shorter inserts will tend to provide the greatest yield of sequence data whilst the sparse long inserts will enable repeats to be spanned. This can in fact be a useful feature when determining whether an assembly is as good as the data theoretically allows. By analysing the ends of contigs/scaffolds to determine whether the ends contain repeats that have a typical length in excess of the largest read length or insert size. In this way one can assess whether or not an assembly is limited by the size of repetitive elements, in which case further sequencing may be the only way forward, or whether there may be further scope for improvement using the existing data.

Correlated SNPs are positions in an assembly where at any given position most of the reads contain one base, but multiple other reads have another base. Because sequencing errors generally occur at random, such correlated discrepancies can indicate the presence of a mis-assembly or the presence of a polyploid genome. In the case of a haploid genome, correlated SNPs may indicate that near-identical repeats have been collapsed into a single copy. In the case of polyploid genomes, if the frequency of correlated SNPs is higher than expected based on the number of homologous chromosomes, it is strong evidence for a collapsed repeat.

A random shearing process of input DNA should ideally result in a uniform coverage of all locations in the genome. A variety of factors can prevent the DNA shearing process from occurring truly randomly. However, significant increases in the level of coverage within a small region can be indicative of a collapsed repeat. Certainly, packages such as AmosValidate (Phillippy et al. 2008) implement such procedures well. However, with the proliferation of de novo sequencing projects, it is anticipated that additional features and metrics will be developed to evaluate assembly quality.

What exactly is "an optimal assembly"? The one with the highest N50? Longest single contig? Of course, the optimal assembly is the assembly which most closely resembles the biological sequence. This assembly minimises the number of mis-assemblies and incorrect bases. Assessing this without a reference sequence is a great challenge – what exactly does one use as a metric?

It is now possible to assemble complete genome sequences, even for mammals, from short sequence reads only. However, these are only of "draft" quality, containing many gaps and are by no means "finished". Most genome sequences published recently are based on long reads (capillary or 454 sequence) or mixtures of long and short reads. However, recent innovations in bioinformatics and in vitro library preparation make assembly of short reads increasingly tractable. With the prevailing trend towards increasing read-lengths in technologies such as Illumina, it is quite possible that in 5 years from now, de novo assembly of short reads will be obsolete as sequencing becomes dominated by new long-read technologies such as Pacific Biosciences' SMRT platform, presenting new challenges for sequence assembly. However, the current crop of sequencers are continued to pump out huge volumes of short-read data and will continue to do so for some time into the future.

14.3.4 RNA-Seq

Whilst in the long run a genomic sequence may be desirable or even necessary, for the purposes of many projects, there is an argument that simply sequencing a cDNA library may provide sufficient information to answer some experimental questions. This enables a much smaller and more manageable subset of data to be analysed whilst preserving information regarding alternative splicing, exon usage and quantitative levels of mRNA expression. It also has the advantage that there is no need to undertake complex and potentially labour-intensive genome assembly and gene

prediction. Although de novo assembly will still be necessary, such projects are less memory intensive and repetitive/non-unique sequences pose less of a problem (although complex isoforms can confound assembly from short-reads).

As third-generation sequence information with multi-kilobase read lengths become available, this assembly step will be bypassed entirely – entire transcriptomes should be sequenced without the need for assembly.

The usual first step in analysing transcript data is to map (i.e. align) sequence reads against a reference genome. When dealing with a eukaryotic transcriptome that undergoes RNA splicing, the alignment step needs to take into account exon–exon junctions. One approach is to align reads against a set of predicted cDNA sequences rather than the raw genome sequence. The success of this approach is limited by prior knowledge of splice junctions and cannot discover new ones. An alternative is to use an alignment algorithm such as TopHat (Trapnell et al. 2009), which allows ab initio discovery of splice sites from RNA-Seq data. TopHat first uses the Bowtie alignment tool (Langmead et al. 2009) to map sequence reads against the reference genome sequence. Among the reads that fail to align will be those that span an exon–exon boundary. This genome-wide multiple alignment reveals islands of genomic sequence to which RNA-Seq reads map and thus reveals the approximate locations of the exons. TopHat then predicts all likely splice donor and acceptor sites in the vicinity of these empirically revealed exons and hence predicts all likely splice junctions, assuming splicing between pairs of splice sites that are nearby, but not necessarily adjacent. The RNA-Seq reads that initially failed to map to the genome are then compared against the catalogue of predicted potential exon–exon junctions. Thus, TopHat offers an efficient means of discovering transcripts in RNA-Seq data without the need for prior knowledge of splice sites. It does, however, depend on the splice sites having canonical splice-junction motifs that are conserved between the transcriptome under investigation and the reference genome. Therefore, it is vulnerable to false positives and false negatives arising from genetic variation. Transcripts generated through non-canonical splicing will also be missed; such transcripts are known in plants, oomycetes and fungi (Russell et al. 2006). To avoid these limitations, RNA-Seq data may be assembled de novo, i.e. without use of a reference genome.

De novo assembly of transcript data presents additional challenges to those encountered when assembling genomic DNA sequence. Transcription is discontinuous, leading to much less contiguity in the transcriptome than the genome. Transcriptome assembly also needs to capture the various different isoforms that can arise from a single gene via alternative splicing, alternative transcriptional start and end sites and other forms of RNA processing. The situation is further complicated by contamination of the cDNA library with genomic DNA. Despite these challenges, Birol et al. (2009) successfully assembled a transcriptome from a human cancer cell line using the ABySS assembler. They first used a DNA molecular denaturisation and re-association method to normalise the cDNA library. Without normalisation, the frequency distribution of cDNA sequences would be heavily biased, with a few transcripts dominating and many rare transcripts being below the limits of detection. The optimal parameter values for de novo assembly vary with sequence

depth. Therefore, heterogeneity in depths of coverage between different transcripts will have a detrimental effect of assembly of RNA-Seq data. Even after molecular normalisation, cDNA libraries are expected to show some bias; so, Birol and colleagues chose a very low threshold depth-of-coverage value for trimming false branches from the de Bruijn graph. They also took care not to discard "bubble" structures in the de Bruijn graph, which might represent alternative isoforms. This strategy successfully led to discovery of novel transcripts. Recently, at least two de novo sequence assembly tools have been released that are specifically designed to deal with RNA-Seq data. One of these, called Oases, is based on the Velvet assembler. The other, Trans-ABySS (Robertson et al. 2010), consists of a pipeline around the ABySS assembler. The most important parameter for assembly methods such as ABySS based on the De Bruijn graph is the k (i.e. the length of the k-mers). The Trans-ABySS method essentially performs a series of ABySS assemblies using a range of values for k and then merges the results into a non-redundant set of contigs.

14.3.5 Visualisation

One area where there is much room for improvement is tools for visualisation of alignments and assemblies. It is inadvisable to blindly trust the output of programs that process alignments. The results reported in the output files may not be what you were expecting and may contain artefacts. It is good practice to visually inspect the alignment and check a random subset of your predicted peaks, SNPs, unusually spliced transcripts, or other feature of interest on the alignment itself. There are now several excellent freely available tools for inspecting alignments. The most basic is the text-based tview program in SAMtools. It is fast, relatively straightforward to use and is sufficient for routine use. However, it does not offer sufficient flexibility (e.g. smooth zooming) for preparing figures for publication or for presentations. Another option is the BAMview plug-in for the Artemis annotation tool. This simultaneously offers all the functionality of Artemis, including graphical representation of annotated features such as genes and plots of G+C content, etc. But Artemis is primarily intended as an annotation tool rather than a visualisation tool. Tablet is a freely available Java application that is flexible and generates visually appealing views of alignments and also can read the standard SAM/BAM format. However, none of these tools easily allow the simultaneous visualisation of several alignments based on a common reference sequence.

The Integrative Genomics Viewer (IGV) from the Broad Institute (http://www.broadinstitute.org/software/igv/) does offer the ability to view multiple BAM tracks relative to a common reference sequence along with the ability to upload GFF annotation. This makes it useful for both comparative genomics and for early-stage genome sequencing projects where multiple contigs are present.

The Broad Institute also hosts a sister program ARGO which is also capable of displaying BAM alignment information as well as additional details from BLAST and Genbank-formatted files. It also contains a comparative genomics viewer (ComBo)

which can be a very useful tool when comparing individual chromosomes between species or samples.

Both IGV and ARGO have the advantage of running as Java WebStart tools and can be launched from a Windows or Mac web-browser on a users desktop without the need to install any software. Although current 32-bit machines on user desktops may preclude the use of these tools for very large projects, with the advent of Windows 7 and the ability to address >4 Gb RAM on 64-bit machines, this problem may be alleviated in coming years.

One of the most common tools used in model-organism projects over the past 5 years has been the GMOD Gbrowse platform. This system displays overlaying tracks to display different features and is used as a base-platform by a number of model organism consortia (e.g. Flybase, TAIR etc). This is a powerful and flexible system capable of displaying SAM/BAM information as well as more detailed information regarding micro-array, RNA-seq and sequence variation. However, it is reliant on centralised management and a backend database, potentially making it difficult for a user without significant Unix experience to set up for their own non-model organism.

Most of our discussion so far has centred around hanging data around a one-dimensional representation of the genome. However, the next generation of visualisation tools need to innovatively and efficiently accommodate multidimensional datasets. There is a need for creative solutions to integrate both molecular data and non-molecular phenotypic data and patient history. We need go beyond a linear map of the genome and visualise its network of many dynamic interactions within the cell and its external environment.

14.3.6 File Formats

Data storage and long-term archival of the vast quantities of data being produced is critical. At the present rate of growth, the volumes of data being generated exceed the projected growth of non-volatile storage in the larger genome centres. A single Illumina HiSeq 2000 can produce over 1 Tb of data within a week. Although perhaps only half of this need be archived, it still represents a considerable challenge.

Some innovations have already been made in terms of storage of quality scores. Rather than storing these as numbers, these are now often stored as ASCII characters. However, this will need to be reduced further in larger centres if data volumes are not to outrun capacity. One suggestion is to align sequences to a dummy reference and only store the alignments. In theory, this should require less space than storing each individual read separately. Illumina are themselves now looking at this approach to reduce their data storage from 30 bytes per base to less than 6.

Beyond pure data storage is the storage of metadata. While the NCBI and EBI archives insist upon metadata upon submission to their databases, for smaller sequencing centres this can be a challenge to manage on a day-to-day basis. FASTQ, SFF and other NGS formats have little if any scope for such information to be

encoded. The Pacific Biosciences SMRT sequencer is designed to produce HDF5 files (Hierarchical Data Format). Though a proprietary format, it does at least enable metadata to be stored and parsed in a standardised fashion.

Interestingly, the SAM format (Li et al. 2009) already has this capability within its headers. In addition to storing the command which generated the file, information regarding the sample, library, sequencer and sequencing centre can be stored. This is, however, reliant on the feature being used consistently prior to publication.

On a more practical level, currently end-users need to be aware of some nuances of file format. For example, the FastQ format, a popular medium for representing sequence reads and their associated quality scores, comes in at least three different flavours: "Solexa", "Illumina" and "Sanger". Most alignment tools assume that the data are in the "Sanger" variant and attempting to align data in "Solexa" or "Illumina" FastQ format will lead to erroneous results. This kind of unnecessary confusion must be avoided in the development of future standards and ideally, software should perhaps intelligently discern what kind of input they are being provided with.

14.3.7 Monolithic Tools and Platforms

One of the main barriers coming between biologists and their data is the apparent lack of a single integrated "one-stop-shop" for the whole analysis workflow. Bioinformaticians tend to work with a set of command-line tools, each one performing a single step and spewing-out arcane output files that, despite being text-based, are certainly not human-readable. This way of working is very much in the tradition of Unix culture, which extols such virtues as: "small is beautiful", "make each program to do one thing well". A great strength of this approach is that tools can be strung together in modular pipelines, offering great power and flexibility. However, the average biologist not steeped in the traditions of the Unix operating system is more at home using large monolithic computer applications that integrate many different tools and tasks into a single graphical user interface. A good example of this approach is Agilent's GeneSpring platform, which will be familiar to many biologists who have analysed gene-expression data from microarrays. On the other hand, the more Unix-oriented bioinformatician would probably eschew GeneSpring and opt for something like Bioconductor, which is a set of modules and tools implemented in the R programming language. Each one of the packages in the Bioconductor toolbox does one job well and for each job, there are often several alternative tools to choose from. Once equipped with the basic skills in using R and Bioconductor, the bioinformatician is usually very resistant to giving up all that flexibility and control in favour of a single integrated graphical application.

A similar situation is unfolding in the world of second-generation sequence analysis. To the bioinformatician who is comfortable with working on the command line and is fluent in a multi-purpose scripting language, a rich selection of tools are available for nearly all steps in any conceivable analysis workflow. And, for any step for which tools are not available, the bioinformatician will quickly and efficiently

write a script to plug the gap. On the other hand, several applications are now available that enable at least some analysis to be performed from a desktop computer by a biologist rather than a specialist bioinformatician. Many of these tools are still in their early stages and additional features and programs will undoubtedly be added in future. Most of these are provided by commercial vendors and require payment of a hefty license fee. However, these commercial packages offer the advantage of relatively well-tested and supported programs with easy-to-use graphical interfaces. The financial cost of the license may be more than justified if it buys the laboratory-based biologist self-sufficiency in data analysis, or at least the ability to start exploring one's hard-won data without having to wait for availability of a specialist bioinformatician. We would also like to point out that as Unix-steeped command-line enthusiasts, we professional bioinformaticians do not see these easy-to-use monolithic applications as a threat to our livelihoods; on the contrary, we want to encourage our colleagues to take responsibility and ownership of their datasets and have more than enough interesting bioinformatics challenges to fill our time and satisfy our desire for new challenges.

The down-side of adopting a commercial solution is, inevitably, some loss of flexibility and configurability, though this need not be too great. A significant danger is the temptation to simply apply a pre-configured workflow and treat it as a "black box" without fully considering or understanding whether each of the steps is appropriate for this particular project's objectives and this dataset. Whereas the open-source command-line tools that a bioinformatician draws upon have usually been subject to official peer-review as well as informal scrutiny of any interested party, the inner workings of proprietary software are not always so clear. A further concern is that by putting all one's metaphorical eggs in one basket by building the analysis infrastructure on a single commercial product, one is dependent on that vendor's ongoing maintenance and development of the product and its continued commitment to the current licensing costs and conditions. On the other hand, with the modular approach, if one component of the pipeline (say, a short-read assembly tool) comes to be no longer suitable, then it can simply be substituted with another open-source component that does the same job.

In the past, these packages have tried to provide easy one-stop-shop systems for individual biologists and labs without access to bioinformatics support or familiarity with Unix-based tools. Especially in the early stages of a new technology, open-source community efforts are nearly always limited to command-line tools. This gap is likely to close, however, as sequencing companies often have initiatives to help software providers (whether community-driven or commercial), provide timely and efficient tools to access datasets. These are extremely important if new sequencing technologies are not to take the community by surprise.

Avadis NGS (http://www.avadis-ngs.com) offers workflows for RNA-seq, ChIP-seq and DNA variant analysis. It was developed on the same platform as used for the development of GeneSpring GX; so, it has the same look and feel as GeneSpring and many of the features behave in the same way. It Supports the SAM/BAM format as the main import format for pre-aligned data.

Similarly, CLC Genomics Workbench software is able to perform similar analyses and even has an API to enable bioinformaticians to plug-in custom tools for biologists to use. However, the take-up of such plug-in systems will be highly dependent on the number of users who adopt such commercial systems. There is little incentive for a bioinformatician to develop software if it is tied to a platform few users are able to access.

Another commercial approach to integrated data analysis is offered by Genome Quest (http://www.genomequest.com/). This is a web-based solution. You transfer raw reads, BAM files, or called variants into GQ via Aspera or FTP (or send a disk). They map to arbitrary reference genomes, call variants, and then annotate them with a variety of data including dbSNP, pharmGKB. They have integrated diagnostic gene panel tests into the annotation as well as 1,000 genomes data. You can add your own annotation tracks and you can do large-scale genome to genome comparisons of your genomes/exomes over public data. Also supports RNA-seq, CHiP-seq, de novo assembly, and metagenomics, as well as general purpose large-scale sequence searching (e.g., all by all comparisons of large datasets). All sequence formats are accepted, including Illumina, SOLiD, Ion Torrent, 454 and Pacific Biosciences. GenomeQuest integrates with Ingenuity, Geneious, GeneSpring.

Galaxy is an open-source web-based front-end to provide a standard interface for many different types of programs. These include programs for sequence assembly, taxonomic classification, sequence similarity searches and assorted tools for data manipulation. In addition, Galaxy offers the ability to keep a record of which steps and parameters were used, pipeline custom analyses and share data and results with other users. The great benefit of Galaxy is that analyses are run remotely so that your PC needs to have nothing more than a web-browser and internet connection. Also other researchers can easily add programs to the Galaxy framework. However, difficulties in transferring large datasets between sites mean that an installation at the user's home institution is usually needed to deal with sequencing data. This may be alleviated if Galaxy servers can access NCBI SRA and EBI ENA archives directly to obtain raw sequence data. Overall, however, the Galaxy framework provides easy access to powerful tools to manipulate and analyse data without the complexity of command line tools or the need to learn to use Unix-style operating systems.

The disadvantages of commercial software are, of course, the initial and sometimes recurring cost of licensing and/or support, often a lack of proper benchmarking and a lack of proper review of the underlying algorithm and code-base.

It should be noted that all such tools either do not permit most parameters to be set (e.g. CLC Genomic Workbench) or permit parameter setting but require an understanding of their meaning (e.g. Galaxy). For example, an assembly using de-bruijn graph-based de novo assemblers require a sweep of parameter space to optimise the assembly (see assessing the quality of an assembly below). This can mean dozens of assemblies to evaluate prior to accepting one for downstream analysis. In summary, the above methodologies offer a way in to various types of analysis. However, to get the most up-to-date and customisable experience, it is always best to learn some basic tools oneself.

14.3.8 Learn a Scripting Language

For the biologist that wants to take ownership of their data analysis, proficiency in a scripting language is extremely useful and perhaps essential. Despite the plethora of useful (and not so useful) computer programs that are available, there are still gaps not filled. We can illustrate this with a few examples from our own experience. For example, we have a backlog of Illumina GA and GA2 sequence datasets generated at various times over the last 3 years and generated at various sites, including our own. During this period, the Illumina base-calling software underwent several upgrades. So we could never be absolutely certain which version of FastQ format our data files were in. However, being fluent in Perl, within a few minutes we were able to write a script (in Perl) that reads through a whole FastQ file and infers which version of FastQ it is based on the frequency distribution of encoded quality scores. Another task for which no tools seemed to exist is, given a list of SNPs in a bacterial genome, determining which are silent and which are non-silent. Again, this was a relatively easy problem to solve using Perl. Other common uses for Perl scripts include automating the running of large numbers of repetitive tasks. For example, we write a simple Perl wrapper script to manage the alignment of 100 bacterial genomic Illumina sequence datasets against a reference genome using BWA and use SAMtools to detect SNPs. To run this analysis manually would require thousands of keystrokes and many hours at the computer terminal. Once the script is deployed, the computer can be left to get on with it.

Although we use Perl, there are at least several alternative scripting languages that are approximately as useful. One of the attractions of Perl is that it provides access to the large and mature set of tools provided by the BioPerl project. Specialist bioinformatics modules are also available for languages including Ruby and Python, amongst others. Arguably, Python is easier to learn than Perl and Ruby is, in many ways, a more elegant language. But the deciding factor in choosing a language may come down to what others around you are using so that you can draw on their support and expertise.

14.3.9 Data Pre-processing

Precisely how "dirty" data can be whilst permitting an "optimal" assembly (see below), is dependent on multiple factors. These include the nature of the sequencing platform, read lengths and the package used for assembly and the amount of sequence remaining after filtering as compared with the size of the genome and the type of sample. There is as yet no optimal and universal set of parameters for each sequencing platform and application.

Anecdotally, a good balance can be struck by removing or trimming reads containing adaptor or other contaminating sequence and only retaining reads with a given proportion of high-quality reads (e.g. 90% of bases must have quality scores

>30 for Illumina reads). If performing SNP calling or other variant analyses it is also crucial to remove PCR duplicates. Typically these can be readily identified if paired-end reads are used.

When preprocessing such datasets the wet-lab process to generate the libraries and the potential biases they can introduce must always be born in mind. For instance, the shearing process used to generate acceptable DNA fragment lengths in RNA-seq experiments will always bias against shorter transcripts. Normalising by the gene-length will not correct this issue. The limitations of current technology must always be considered when interpreting final results.

14.3.9.1 Metagenomics

One of the long-term goals of this field is to be able to take a sample of soil, water or other material containing organic matter, extract all biologically relevant material present and to then characterise and compare them. This could involve, cell-based population studies, protein and metabolite characterisation using mass-spectrometry or DNA/RNA sequencing.

In diverse environments, this could potentially involve sequencing petabases of RNA or DNA (mention direct RNA sequencing) to ensure that all members of the environmental population are sequenced to an adequate depth.

Once sequenced, it would in theory be possible to reconstruct all genomes or transcriptomes present, and profile cell-based, protein and metabolite changes against each other. One could envisage observing changes due to temperature, light, pH and invading populations. There are some major hurdles to overcome before this becomes feasible on a routine basis, however:

1. Sequence length. Short sequences are more likely to be identical between two species than longer ones.
2. Sequencing error rates. Are single base differences between two sequences due to errors or do they truly represent their hosts?
3. Sequencing volumes
4. Assembly algorithms
5. Computational hardware
6. Analysis pipelines

Until recently these problems have been side-stepped by sequencing ribosomal tag sequences which it is thought represent individual sub-species. In recent years, however, the more ambitious approach has been undertaken in the Human Gut Microbiome project demonstrating that it is possible to perform metagenomics with currently available hardware. However, this is far from routine and requires considerable development before it is can be considered as straightforward as a typical genome assembly.

Current programs which are beginning to deal with the data in a user-friendly and intuitive manner is the Metagenome Analyser (MEGAN). Although this will not perform any assembly, it will analyse reads or contigs which have been run

through NCBI Blast and report taxonomy information, GO and KEGG information in an easy to interpret format. With such software it is straightforward to visualise whether particular species are present and even whether particular bio-chemical pathways are likely to be present.

K-mer-based approaches to genome assembly generally contain assumptions which do not lend themselves to metagenome assembly. For example, some packages such as EulerSR contain error-correction algorithms which remove and/or correct relatively low-coverage *k*-mers on the assumption that these are likely to be errors. However, this is only a valid assumption when large coverage of a genome is available and there are no near-identical sequences within a genome. In a metagenome, this may not be a valid assumption.

As yet there is no single "Metagenome" standard due to the lack of datasets, however, these will undoubtedly appear once we as a community learn the best ways of dealing with such data.

14.3.9.2 Applications in Personalised Medicine

Personalised medicine, utilising whole genome information is likely to become a key focus for development of high-throughput processing and analysis. Commercial providers such as 23andMe and the now defunct DECODE provide coverage of known markers for disease along with basic information to interpret this information.

However, the lack of large-scale studies in many markers mean that the statistics in use to predict risk ratios often vary between providers and can often change drastically if new studies overturn previous results. Presenting such information to users who may have little or no training in either genetics or statistics is a major challenge.

An even greater challenge will be utilising whole exome/genome data. Whilst known markers for particular diseases provide relatively straightforward indicators for clinicians, the lack of genomes and association data for many less common diseases, or diseases with low penetrance means that a great deal of additional work still remains.

14.3.9.3 Computer Environment: Use a 64-Bit Unix-Like System

Choice of computer operating system currently still has an impact on efficiency of data analysis. In general, bench-based biological scientists tend to use Microsoft Windows, whereas bioinformaticians tend to favour Linux or some other Unix-like operating system. The Macintosh OS X occupies a kind of middle ground appealing to members of both communities with its Unix-like core and its slick user interface. Most of the existing tools surveyed in this book are primarily developed for use in a Linux-like environment. Windows 7 is the first all 64-bit OS released by Microsoft (with the exception of Windows 7 Home Basic). This enables the use of >3.5 Gb of

RAM and PCs will soon begin to ship to take advantage of this. How long before University IT systems are upgraded to take advantage of this however is uncertain. However, it does enable developers to design applications with looser constraints.

14.3.9.4 NGS Datasets are Large

The proliferation of sequencing projects and other high-throughput biological data will inevitably mean that data integration and dissemination will be a crucial issue. How does one ensure good QC and curation of data when there may only be 2–3 individuals in the world with an interest in the project? Individual researchers will not (in all probability) have the expertise necessary to maintain GMOD style databases and web-front-ends.

14.4 Concluding Remarks

Computational challenges of data analysis, visualisation and data integration are now the bottlenecks in genomics, no longer the DNA sequencing itself. Innovative new approaches will be needed to overcome these challenges. In integrating datasets, we need to go beyond the one-dimensional genome and integrate heterogeneous data-types, both molecular and on-molecular. Computational tools must be available that are specifically tailored to non-bioinformaticians. In particular, well-engineered robust software will be needed to support personalised medicine. This software will have to perform analysis of large datasets, communicate with vast existing databases whilst securely dealing with patient privacy concerns. Finally, to meet these changes, the next generation of genomics scientists needs to include multidisciplinarians with expertise in biological sciences as well as in at least one mathematical, engineering or computational discipline.

References

Birol I, Jackman SD, Nielsen CB, Qian JQ, Varhol R, Stazyk G, Morin RD, Zhao Y, Hirst M, Schein JE, Horsman DE, Connors JM, Gascoyne RD, Marra MA, Jones SJ (2009) *De novo* transcriptome assembly with ABySS. Bioinformatics 25:2872–7.

Bryant DW Jr, Wong WK, Mockler TC (2009) QSRA: a quality-value guided *de novo* short read assembler. BMC Bioinformatics 10:69.

Cock PJ, Fields CJ, Goto N, Heuer ML, Rice PM (2010) The Sanger FASTQ file format for sequences with quality scores, and the Solexa/Illumina FASTQ variants. Nucleic Acids Res 38:1767–71.

Hiatt JB, Patwardhan RP, Turner EH, et al (2010) Parallel, tag-directed assembly of locally derived short sequence reads. Nat Methods 7:119–22.

Koboldt DC, Chen K, Wylie T, Larson DE, McLellan MD, Mardis ER, Weinstock GM, Wilson RK, Ding L (2009) VarScan: variant detection in massively parallel sequencing of individual and pooled samples. Bioinformatics 25:2283–5.

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:R25. doi 10.1186/gb-2009-10-3-r25.

Li H, Durbin R (2009) Fast and accurate short read alignment with Burrows-Wheeler transform. Bioinformatics 25:1754–60.

Li H, Handsaker B, Wysoker A, Fennell T, Ruan J, Homer N, Marth G, Abecasis G, Durbin R (2009) 1000 Genome Project Data Processing Subgroup. The Sequence Alignment/Map format and SAMtools. Bioinformatics 25:2078–9.

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

Li R, Zhu H, Ruan J, et al (2010) De novo assembly of human genomes with massively parallel short read sequencing. Genome Res 20:265–72.

Maccallum I, Przybylski D, Gnerre S, et al (2009) ALLPATHS 2: small genomes assembled accurately and with high continuity from short paired reads. Genome Biol 10:R103.

Malhis N, Jones SJ (2010) High quality SNP calling using Illumina data at shallow coverage. Bioinformatics 26:1029–35.

Ondov BD, Varadarajan A, Passalacqua KD, Bergman NH (2008) Efficient mapping of Applied Biosystems SOLiD sequence data to a reference genome for functional genomic applications. Bioinformatics 24:2776–7.

Phillippy AM, Schatz MC, Pop M (2008) Genome assembly forensics: finding the elusive mis-assembly. Genome Biol 9:R55.

Robertson G, Schein J, Chiu R, Corbett R, Field M, Jackman SD, Mungall K, Lee S, Okada HM, Qian JQ, Griffith M, Raymond A, Thiessen N, Cezard T, Butterfield YS, Newsome R, Chan SK, She R, Varhol R, Kamoh B, Prabhu AL, Tam A, Zhao Y, Moore RA, Hirst M, Marra MA, Jones SJ, Hoodless PA, Birol I (2010) De novo assembly and analysis of RNA-seq data. Nat Methods 7:909–12.

Rumble SM, Lacroute P, Dalca AV, Fiume M, Sidow A, et al (2009) SHRiMP: Accurate Mapping of Short Color-space Reads. PLoS Comput Biol 5(5):e1000386. doi:10.1371/journal.pcbi.1000386.

Russell AG, Charette JM, Spencer DF, Gray MW (2006) An early evolutionary origin for the minor spliceosome. Nature 443:863–6.

SeqAnswers (2011) http://seqanswers.com/wiki/Software/list. Accessed 12 Feb 2011.

Simpson JT, Wong K, Jackman SD, Schein JE, Jones SJ, Birol I (2009) ABySS: a parallel assembler for short read sequence data. Genome Res 19:1117–23.

Sorber K, Chiu C, Webster D, et al (2008) The long march: a sample preparation technique that enhances contig length and coverage by high-throughput short-read sequencing. PLoS One 3:e3495.

Sundquist A, Ronaghi M, Tang H, et al (2007) Whole-genome sequencing and assembly with high-throughput, short-read technologies. PLoS One 2:e484.

Trapnell C, Pachter L, Salzberg SL (2009) TopHat: discovering splice junctions with RNA-Seq. Bioinformatics 25:1105–11.

Wang K, Singh D, Zeng Z, Coleman SJ, Huang Y, Savich GL, He X, Mieczkowski P, Grimm SA, Perou CM, MacLeod JN, Chiang DY, Prins JF, Liu J (2010) MapSplice: accuratemapping of RNA-seq reads for splice junction discovery. Nucleic Acids Res 38:e178.

Young AL, Abaan HO, Zerbino D, et al (2010) A new strategy for genome assembly using short sequence reads and reduced representation libraries. Genome Res 20:249–56.

Zerbino DR, Birney E (2008) Velvet: algorithms for de novo short read assembly using de Bruijn graphs. Genome Res 2008 18:821–9.

Index

N. Rodríguez-Ezpeleta et al. (eds.), *Bioinformatics for High Throughput Sequencing*,
DOI 10.1007/978-1-4614-0782-9, © Springer Science+Business Media, LLC 2012