From the Programme for Genomics and Bioinformatics
Department of Cell and Molecular Biology
Karolinska Institutet, Stockholm, Sweden

Solving Repeat Problems in Shotgun Sequencing

Erik Arner

Stockholm, 2006

Karolinska Institutet

Abstract

Shotgun sequencing is the most powerful strategy for large scale sequencing. Two main approaches exist: clone-by-clone and whole genome shotgun (WGS). In the clone-by-clone strategy, overlapping clones are amplified and then sheared in a random fashion. In the WGS approach, a sufficient amount of cells from the target organism are obtained, and the random shearing is performed on extracted DNA.

In both approaches, the resulting fragments are cloned and the fragment ends are subsequently sequenced, producing sequence reads. If a sufficient amount of sequence has been obtained, the reads will overlap in a way that makes it possible to deduce their correct order. A number of computer programs have been developed for this task. However, none of these programs are capable of producing correct assemblies if the target sequence contains repeats. This is because assembly algorithms in general are greedy, which means that when faced with different alternatives for the positioning of a read, the algorithm will fit the read at the first available position meeting the criteria for inclusion into the assembly. The resulting assemblies typically have the repeat regions degenerated, truncating the regions into a few copies with abnormally high shotgun coverage. This phenomenon occurs even when the repeat copies differ from each other, since the assembly programs are unable to distinguish the subtle differences between repeat elements from the sequencing errors produced by the sequencing apparatus .

The work presented here is aimed at solving the repeat problem by detecting and utilizing single base differences between nearly identical repeats. In paper I, a statistical method for detecting repeat differences in the presence of sequencing errors was developed, implemented, and tested on simulated data. We showed that it is possible to obtain high specificity as well as sensitivity compared to other methods, by evaluating coinciding deviations from consensus in pairs of columns in multiple alignments. In paper II, a finishing tool (DNPTrapper) that visualizes the differences and enables manual and semi-automatic resolution of repeat regions was constructed and tested with simulated data as well as real data from the *Trypanosoma cruzi* WGS project. Results showed that using DNPTrapper, it is possible to resolve and analyze complicated repeat regions previously considered difficult or even impossible to resolve. Finally in paper III, five repeated genes in *T. cruzi* were analyzed using DNPTrapper. Different repeat characteristics in the parasite were described, and it was shown that thorough analysis of repeat regions is required for correcting erroneous consensus sequences of repeated genes in the assembly.

Keywords: Shotgun sequencing, repeated DNA, genomics, whole genome shotgun, hierarchical shotgun, mate pairs, DNP, bioinformatics

ISBN 91-7140-996-3

Printed by
Larserics Digital Print AB

You need a little clarity?
Check the similarity!
– KRS-ONE

Publications included in this thesis

The thesis is based on the following papers, referred to by the Roman numerals I-III.

I. Martti T. Tammi, **Erik Arner**, Tom Britton, Björn Andersson.
Separation of nearly identical repeats in shotgun assemblies using defined nucleotide positions, DNPs.
Bioinformatics, 18(3):379–88, 2002.

II. **Erik Arner**, Martti T. Tammi, Anh-Nhi Tran, Ellen Kindlund, Björn Andersson.
DNPTrapper: an assembly editing tool for finishing and analysis of complex repeat regions.
BMC Bioinformatics, 7:155, 2006.

III. **Erik Arner**, Ellen Kindlund, Daniel Nilsson, Fatima Farzana, Marcela Ferella, Martti T. Tammi, Björn Andersson.
Database of Trypanosoma cruzi repeated genes: 20 000 novel coding sequences.
Manuscript.

Related publications

i. Martti T. Tammi, **Erik Arner**, Björn Andersson.
TRAP: Tandem Repeat Assembly Program produces improved shotgun assemblies of repetitive sequences.
Computer Methods and Programs in Biomedicine, 70(1):47–59, 2003.

ii. Martti T. Tammi, **Erik Arner**, Ellen Kindlund, Björn Andersson.
Correcting errors in shotgun sequences.
Nucleic Acids Research, 31(15):4663–72, 2003.

iii. Martti T. Tammi, **Erik Arner**, Ellen Kindlund, Björn Andersson.
ReDiT: Repeat Discrepancy Tagger–a shotgun assembly finishing aid.
Bioinformatics, 20(5):803–4, 2004.

iv. Najib M. El-Sayed, Peter J. Myler, Daniella C. Bartholomeu, Daniel Nilsson, Gautam Aggarwal, Anh-Nhi Tran, Elodie Ghedin, Elizabeth A. Worthey, Arthur L. Delcher, Gaëlle Blandin, Scott J. Westenberger, Elisabet Caler, Gustavo C. Cerqueira, Carole Branche, Brian Haas, Atashi Anupama, **Erik Arner**, Lena Åslund, Philip Attipoe, Esteban Bontempi, Frédéric Bringaud, Peter Burton, Eithon Cadag, David A. Campbell, Mark Carrington, Jonathan Crabtree, Hamid Darban, Jose Franco da Silveira, Pieter de Jong, Kimberly Edwards, Paul T. Englund, Gholam Fazelina, Tamara Feldblyum, Marcela Ferella, Alberto Carlos Frasch, Keith Gull, David Horn, Lihua Hou, Yiting Huang, Ellen Kindlund, Michele Klingbeil, Sindy Kluge, Hean Koo, Daniela Lacerda, Mariano J. Levin, Hernan Lorenzi, Tin Louie, Carlos Renato Machado, Richard McCulloch, Alan McKenna, Yumi Mizuno, Jeremy C. Mottram, Siri Nelson, Stephen Ochaya, Kazutoyo Osoegawa, Grace Pai, Marilyn Parsons,Martin Pentony, Ulf Pettersson, Mihai Pop, Jose Luis Ramirez, Joel Rinta, Laura Robertson, Steven L. Salzberg, Daniel O. Sanchez, Amber Seyler, Reuben Sharma, Jyoti Shetty, Anjana J. Simpson, Ellen Sisky, Martti T. Tammi, Rick Tarleton, Santuza Teixeira, Susan Van Aken, Christy Vogt, Pauline N. Ward, Bill Wickstead, Jennifer Wortman, Owen White, Claire M. Fraser, Kenneth D. Stuart, Björn Andersson.
The Genome Sequence of *Trypanosoma cruzi*, Etiologic Agent of Chagas Disease.
Science, 309(5733):409–415, 2005.

Contents

Abbreviations

BAC Bacterial artificial chromosome

BLAST Basic local alignment search tool

DNA Deoxyribonucleic acid

DNP Defined nucleotide position

HS Hierarchical shotgun

kb Kilobases

NP Non-polynomial

PCR Polymerase chain reaction

RNA Ribonucleic acid

SNP Single nucleotide polymorphism

WGS Whole genome shotgun

Part I

Introduction

Chapter 1

Repeated DNA

1.1 Genomes and Genomics

The genome of an organism is defined as its set of genes and non-protein-coding sequences, as they appear in the DNA present in all of its cells. Genomics is the study of genomes; after the sequencing of the first free-living organism, *Haemophilus influenzae* in 1995 [1], genomics has emerged as a major field of research in life sciences.

Before the advent of large scale, high-throughput methods for gene and genome sequencing, genes were studied one or a few at a time by molecular biologists. Assessing the function of a gene required considerable effort and time. The gene had to be cloned and subsequently expressed in a suitable system. The expressed protein could then be extracted and analyzed further for activity, and different assays could be applied in order to find out if the protein had functional similarities with previously known proteins.

The work leading up to the publishing of the draft genome sequence of *Homo sapiens* [2, 3], and the still ongoing aftermath, has led to an explosion with regard to improvements in sequencing techniques, the computer methods and programs associated with them, and the number of genomes subjected to sequencing. The number of sequenced genomes is growing exponentially each year (see the Genomes OnLine Database (GOLD, [4]) for finished and ongoing projects), and bacterial genomes are nowadays routinely sequenced and assembled at genome centers within the course of a day. The task of determining the probable function of a gene can now be performed in a matter of seconds using a computer and an internet connection, provided that the sequence is known. A database search against all known genes in all known species quickly reveals if the gene is similar to something already characterized, and to what degree. This is often all that is needed in order to make an educated guess about the function of the gene, as high sequence homology usually implies similarity in function. Although the laborious process of studying individual genes still comes in handy when the intricate details of protein activity and function are being

analyzed, or in the case that the gene of interest has no known counterparts in other species, access to whole genomes has enabled a shift in focus to large scale studies of genes and how they are expressed. Through the use of microarrays, it is now possible to measure the expression levels of all genes of an organism simultaneously in a single experiment. This allows for a system-wide approach in molecular biology, complementing previously developed techniques for studying specific pathways up close.

Through comparative genomics, the gene content of a sequenced genome can immediately be estimated to a large part, by comparing the genome to that of an already sequenced, related species. This has the effect that focus can be directed at once towards the parts that are specific for that organism, under the assumption that homologous regions essentially have the same function in both species. Comparisons between species also makes it possible to track speciation events and determine the evolutionary distance between species. In addition, a rough estimate of the genome arrangement of an organism can be obtained quickly at low cost using comparative sequencing.

All is not genes that is DNA. When the concept of introns was introduced for eukaryote species [5], it was initially thought that genomes consisted of exons, introns and RNA genes. As more non-coding sequence was discovered, it was sometimes referred to as "junk DNA", essentially meaning "we have no idea what it does". Through genome sequencing and genomics, continuous study of genes and their genomic surroundings has revealed the presence of regulatory elements such as promoters and enhancers, as well as sequence with structural and spacing functions. It has become increasingly apparent that in order to get a comprehensive understanding of the biology of humans and other species, it is crucial to find out exactly what is present in their genomes, in addition to the information contained in the genes.

However, the process of sequencing a genome is not completely straightforward, especially when it comes to higher organisms. Apart from biological limitations that hinder preparation and cloning of DNA with particular traits using current methods (such as the heterochromatin of higher eukaryotes), the repeated sequences present in most genomes confound the commonly used computer programs that aid in the process of putting the genomes together from the raw data produced by sequencing machines. The problems arising from repeats consitute the key obstacles in genome sequencing today. The work presented in this thesis is aimed at solving some of these problems.

1.2 Repeated DNA

Repeated DNA is a prominent feature of the genomes of most higher organisms, and exists in several types. The repeated elements can be anything from large, several kilobases (kb) long segments, to short mono-, di- or trinucleotide sequences. They can be repeated once or occur in thousands of copies, dispersed or in tandem. The repeated sequences can be decodable in the form of protein-coding genes, RNA genes, and regulatory elements such as transcription

factor binding sites, but also have purely structural functions related to DNA conformation or spacing between coding sequences.

Different organisms have different types and amounts of repeated DNA in their genome. As an example, the human genome is estimated to consist of more than 50% repeated sequences [6], whereas the corresponding ratios in sweet corn (*Zea mays*) and roundworm (*Caenorhabditis elegans*) are 77% and 16.5% respectively [7]. The following sections will briefly describe some of the various kinds of repeats that exist in different organisms, as well as their function and origin, in more detail. It should be noted that definitions of different types of repeats often overlap in the literature, and that a thorough characterization of all kinds of repeated DNA would require a thesis in itself.

1.2.1 Short repeats

A large part of the repeat content in the genome of an organism is present in the form of short elements of various kinds, repeated to varying degrees. For a good review on short repeats, see [7], on which this subsection, as well as the section on transposons, mostly is based.

VNTRs

Variable nucleotide tandem repeats (VNTRs) are units of length 2 - 100 bases, repeated in tandem in varying copy numbers up to a thousand times. They can facilitate gene expression by acting as enhancers and promoters, but also act as silencers. Other functions include signals for recombination, chromatin packing and other structural modifications of the DNA molecule. Sometimes referred to as minisatellites, VNTRs are also found at the telomeres.

Tandem array satellites

Typically 100 - 200 bases long and often repeated thousands of times in tandem, tandem array satellites are a major component of heterochromatin in higher eukaryotes, coupled to centromere organization and function, where they interact with histones and other chromatin packing proteins. It is however unclear whether repeats at the centromere are essential for centromere function, or if centromere function gives rise to repeats [8].

Microsatellites

Microsatellites consist of units that are 1 - 4 bases long, and are usually repeated between 10 and 100 times. They can have gene silencing and spacing effects, and copy numbers often vary between individuals which makes them usable for genotyping and different kinds of population studies.

1.2.2 Transposons and retrotransposons

Transposons are mobile genetic elements that can insert themselves at different locations in the genome. They work either by "cut and paste" (transposons) or "copy and paste" (retrotransposons), and can be anything from a few hundred bases to several kb in length. They contribute greatly to the plasticity of a genome due to their mobility, and are involved in processes as diverse as transcription, post-transcriptional control, translation, DNA replication, and chromatin organization.

Well known types of retrotransposons are LINEs and SINEs (Long/Short Interspersed Nucleotide Elements), of which LINE-1 and *Alu* are important examples in human.

1.2.3 Segmental duplications

In higher mammals, a common repeat phenomenon is segmental duplications, usually defined as large segments of DNA (1 - 100 kb) occurring at two or more locations in the genome with high (>90%) sequence similarity [9]. Segmental duplications are present in roughly 5% of the human genome [10] and are considered to be a major driving force in the evolution of vertebrates. The genomes of humans and other primates are especially enriched for duplications – compared to many other species, segmental duplications in human are larger, more interspersed, more recent and more common [11]. Although human duplications involve similar genes as in rat and mouse (immunity/defense, growth/development), duplications in these species more often occur in gene poor regions [12, 13]. Recent events have given rise to novel genes expressed in the human brain [14].

Segmental duplications are also features of plant genomes such as those of *Arabidopsis thaliana* and rice [15, 16], that have very similar distributions of tandemly duplicated genes often involved in stress reactions.

Duplications provide evolutionary possibilities to a genome through processes of neofunctionalization and subfunctionalization [17]. The former term describes the process where a duplicated copy is free to diverge into a different protein or regulatory element, while the ancestral copy is maintained by selective pressure, whereas the latter refers to a process where the function of a protein is divided between several gene variants, thus providing modularity.

There are several models describing the formation of segmental duplication, the two most important being transposition and unequal crossing-over (UCO). For a description of these mechanisms, see [17], where it is also suggested that combinations of transposition and UCO can be regarded as "gene factories", according to the functionalization models described above.

1.2.4 Repeated genes

A special case of duplications is when a gene or a group of genes are repeated several times in tandem. Several reasons have been proposed to explain why this

arrangement might be beneficial to an organism. One is the ability to mobilize high expression levels of different genes in different situations, such as histones in DNA replication and rRNA during rapid cell growth [17].

A recently introduced concept is "noise control" [18], where the idea is that the presence of multiple copies of genes makes expression levels less sensitive to stochastic fluctuations in other parts of the transcription machinery. This may be especially important in organisms essentially lacking transcriptional control, like *Trypanosoma cruzi* [19].

A third benefit, of which there are several examples of in biology, is microfunctionalization [17], where an organism can keep several copies of the same gene with subtle differences (such as with immunoglobulins and olfactory receptors in human) and thereby maintain a high flexibility in antigen recognition. Disease resistance genes in plants [20, 21], as well as the defensin system in human [22, 23] also seem to follow this pattern.

Conversely, pathogens like different bacteria [24] as well as trypanosomatid parasites [25, 26] appear to utilize this mechanism for drug resistance and evasion of the host immune system at infection. They keep arrays of subtly differing surface antigens unexpressed and transfer the suitable variant to an expression site through gene conversion.

1.2.5 Genomic disorders

Various diseases are connected to duplications and other repeats in the human genome. Duplicated segments, *duplicons*, facilitate chromosomal rearrangements like duplications, inversions and deletions, through the process of non-allelic homologous recombination. This can lead to disorders due to dosage imbalance in gene expression. Disorders range from color blindness to serious developmental and mental retardation syndromes, increased susceptibility to panic disorders and phobias, and infertility [27]. Duplicons can be simple or complex, i.e. consisting of duplicons within duplicons, and can cause rearrangements within and between chromosomes [28].

Copy number variation is also implicated in cancer, where over-expression due to an increased copy number of oncogenes has been observed. Examples include ovarian cancer [29] and melanoma [30].

Expansion of short, simple repeats, often tri-nucleotide sequences but also units of longer lengths, cause a number of hereditary disorders such as Huntington's disease and fragile X [31]. In these kinds of diseases, the repeat array expands over generations, with the disease often occurring in a mild form in the generation before the severe version.

1.3 Shotgun Sequencing

1.3.1 Sanger Sequencing

In 1977, Frederick Sanger *et al.* developed a method for determining a target DNA sequence [32] that today, 30 years later, still is the prevailing sequencing method. Current high-throughput implementations of the method, which essentially is a chain termination and gel separation technique, are able to produce high quality sequence spanning up to 1000 bases of DNA (adenine (A), thymine (T), guanine (G), and cytocine (C)) in one run. Most genomes of interest are obviously several orders of magnitude larger than this – e.g. the human genome, which consists of approximately three billion bases.

1.3.2 The Shotgun Method

A solution to this problem was devised in 1980, also by Sanger [33], and is referred to as *shotgun sequencing*. Inspired by techniques used when Holley *et al.* for the first time in history determined the nucleotide sequence in a nucleic acid (a yeast tRNA [34]), the main principle is to obtain an overlapping set of subsequences and use the overlaps to puzzle the sequence back together. The following sections decribe shotgun sequencing in more detail.

In shotgun sequencing, the sequence of interest is first amplified, e.g. by growing transformed bacteria in culture. The amplified DNA is sheared in a random fashion, producing a redundant set of fragments (spanning the target sequence several times) that are cloned and can be sequenced from the ends. An alternative to amplifying the sequence beforehand is used in whole genome shotgun (WGS, described in more detail below), where an abundant amount of the entire cellular DNA is extracted and sheared in the same way.

Each sequence obtained in this way is referred to as a *read*. Since there is redundancy in the amount of sequence extracted, and the shearing is performed in a random way, the reads will overlap to different extents. The overlaps can be detected using string matching computer algorithms, and using this information it is possible to deduce the positional layout of the reads in a multiple alignment. Finally, a consensus sequence is computed at each column in the alignment, resulting in a contigous sequence, *contig*, that ideally is identical to the original, target sequence. This process is referred to as *assembly* (figure 1.1).

1.3.3 Shotgun Sequencing Assembly Programs

Numerous assembly programs have been developed over the years, e.g. SEQAID [35], GAP [36], the CAP suite of programs [37, 38, 39, 40], ARACHNE [41, 42], Phrap (Phil Green, unpublished), and the Celera Assembler [43]. Phrap (http://www.phrap.org) is, arguably, the most widely used assembly program and is also the basis of more recent assemblers like Phusion [44] and RePS [45].

The programs differ in the details, but the main features of most of them can be outlined as follows: 1. Preparation, 2. Computation of pairwise overlaps,

3. Read layout, and 4. Consensus generation. An exception to this scheme is EULER [46], which will be described in section 1.4.3.

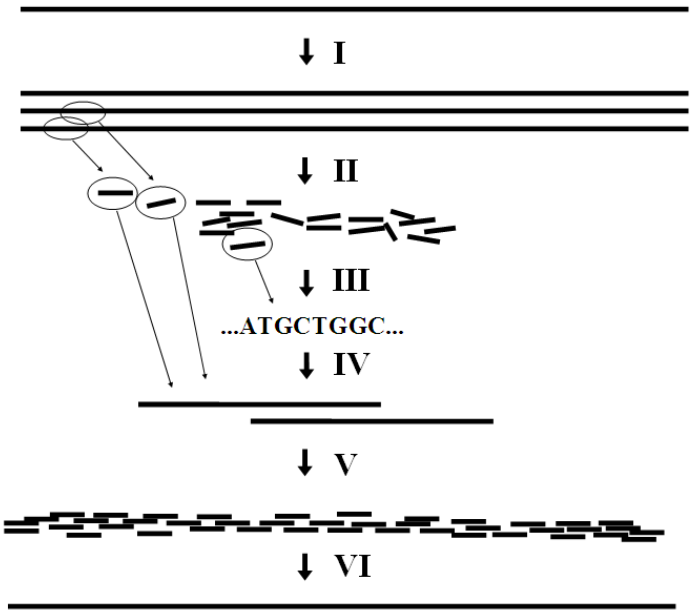

Figure 1.1: Schematics of shotgun sequencing. I. Amplification of target sequence. II. Random shearing. III. Cloning, sequencing and base calling. IV. Detection of pairwise overlaps. V. Assembly. VI. Computation of consensus sequence.

Preparation

An initial step in the assembly stage is to screen the input reads for different sequence features that render them unsuitable to include in the assembly in part or completely. These features include contamination (e.g. by vector sequence or host organism for fragment cloning) and the presence of low complexity regions, e.g. a long stretch of the same nucleotide repeating itself over and over.

The reads are also quality trimmed, since the Sanger technique produces reads that have a high quality in the middle, with rapidly increasing error rates towards the ends. This procedure is simplified by methods of computing error probabilities for each individual base in the reads, based on analysis of the electropherograms obtained from the sequencing apparatus (figure 1.2). Phred

[47, 48] and LifeTrace [49] are examples of software developed for this purpose, Phred being the most widely used.

Furthermore, the reverse complement of each sequence needs to be computed and added to the dataset, since there is no way of knowing from which strand an individual read was sequenced.

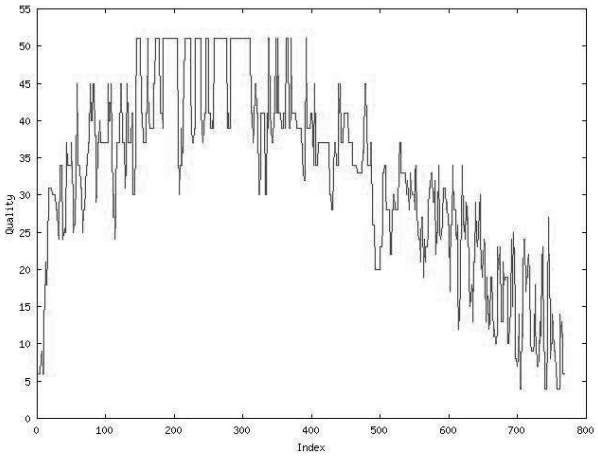

Figure 1.2: A quality profile of a sequence read from a shotgun project. Quality (y-axis) increases rapidly for increasing read index (x-axis) in the beginning of the read, and decreases towards the end. A quality value q corresponds to an error probability $\epsilon = 10^{-q/10}$.

Computation of pairwise overlaps

After preparation, pairwise overlaps between all reads are computed. This could easily be a limiting step for shotgun assembly, since the number of necessary comparisons grows quadratically with the number of input sequences. A common way to get around this problem is to scan all sequences and record the positional occurence of each k-letter word in the data set. All reads sharing k-tuples are then compared pairwise, using the position of the k-tuple as centerpoint in a banded matrix [50], and the quality of the overlap (based on the number of mismatches and gaps between the sequences) is computed using dynamic programming [51]. This technique is similar to the one used in BLAST [52], and basically reduces the quadratic problem to a linear one. Additional scoring of the overlaps can be performed using the sequencing error probabilities.

Read layout

In the layout stage, it is computationally infeasible to try all possible combinations of overlaps and choose the one that appears to be best. This is because

the layout problem is NP-complete [53], and analogous to the classic Travelling Salesman Problem.

Instead, computation of the layout of reads is usually performed by a greedy method, where the starting point of the assembly is the best scoring overlap. The reads of this overlap are added to the assembly with their relative position according to the overlap retained. Subsequently, the next read to be added is chosen from the highest scoring overlap involving any of the reads already added to the assembly. This is performed in an iterative fashion, until all overlaps in the data set have been considered.

Different heuristic considerations are often coupled to this process, where additional demands are required to be met by a candidate read before its inclusion. One such requirement can be that the read is required to have high scoring overlaps with all reads that span the region where it is to be included.

An alternative and completely analogous way of describing the layout algorithm is to view all reads as vertices in an undirected graph, where the edges represent pairwise overlaps between reads. The task then becomes to prune and divide the graph according to the following algorithm:

1. Start a new contig by picking the highest scoring edge in the dataset.

2. Add vertices connected by the new edge to a *vertex list*, if they are not already present in the list.

3. Add edges emerging from newly added vertex/vertices to an *edge list*, sorted by score.

4. Go through the *edge list*, removing all edges from the graph until an edge connected to a vertex not already in the *vertex list* is encountered.

5. Use heuristics to decide if the edge should be kept or not. If yes, go to step 2. If no, remove the edge from the graph and go to step 4.

This procedure is repeated until all edges in the *edge list* has been examined. If there are still unexamined edges left in the data set, a new contig is started by repeatedly picking the remaining highest scoring edge and going through the algorithm, until all edges in the entire dataset have been considered. The result is a division of the original graph into several, pruned subgraphs, which then can be converted to alignments (contigs) of reads. This usually also entails a step of optimizing the alignment locally, using algorithms such as ReAligner [54].

Consensus generation

When the layout is complete, the consensus sequence is computed. This can be performed in different ways. One strategy is a simple majority vote on each column, where the most abundant base on the column is chosen as consensus. Phrap divides the alignment into segments, where the consensus sequence of each segment is defined as the sequence of the read with the highest mean quality in that segment. Other programs use different statistical methods involving the individual error probabilities for each base in the reads, usually also taking the coverage on each strand into consideration.

1.3.4 Mate Pairs and Scaffolds

Due to the random sampling of the target sequence, the read coverage across the region is unevenly distributed. The coverage at a given position is distributed according to the Poisson distribution, with the mean coverage of the shotgun project as the mean variable [55].

The consequence is that, depending on the mean shotgun coverage, there will be varying amounts of gaps in the contiguos sequence, where no read has sampled the target sequence. In addition to gaps caused by the randomness of the sampling, the target sequence may include regions that are hard to clone, resulting in low or no coverage in these regions. The reads may thus be assembled into several contigs instead of just one. The gaps can be closed by designing PCR primers at the end of contigs, and obtaining additional sequence at these positions. This process was greatly simplified in 1990, when Edwards *et al.* [56] modified the fragment sequence protocol to read the sequence from both ends of the shotgun fragment insert. Since the fragment insert size is known, additional positional information regarding the reads can be obtained. Such paired fragment reads are most often referred to as *mate pairs*, and can be used to order contigs positionally. This is done by identifying contig pairs that have mutual mates, and place them adjacent to each other in *scaffolds*(figure 1.3). Mate pairs can also be used to verify the correctness of read placement within contigs, since any read in a contig should have its mate placed at a known distance in the alignment. It is common to create several libraries with different insert sizes, in order to get different levels of resolution in the paired end analysis.

Scaffolding algorithms are built into all whole genome assemblers; in addition, separate scaffolding software exists that given a set of contigs and paired end sequences creates scaffolds. Examples are Bambus [57] and GigAssembler[58].

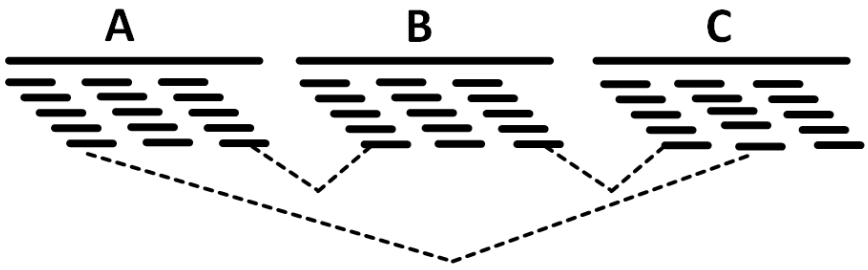

Figure 1.3: Three contigs ordered horizontally in a scaffold using paired ends. Dashed lines indicate pairs of different insert sizes.

1.3.5 Finishing

As discussed in the previous section, uneven shotgun coverage across the genome results in gaps between contigs where no sequence has been obtained. To complete the target sequence, additional sequencing is required in these regions, in a process referred to as *finishing*. This is a long and tedious process – the still ongoing finishing of the human genome is estimated to constitute about 50% of the total time and cost of the whole project [59]. Regions in need of finishing can be pinpointed automatically by software such as Autofinish [60], but the task is complicated by artificial gaps caused by the presence of repeated sequences as described in the following sections.

Assembly problems that cannot be resolved automatically can in many cases be worked out using manual analysis and editing. This is enabled by tools that allow close-up inspection of the layout of individual contigs, as well as birds-eye view of several contigs with potential paired end links between them. Highlighting of problematic regions and suggestions for primer design where extra sequencing is needed are also typical features of finishing software. Consed [61] and the Staden Package [62] are the most popular finishing tools. There are also other approaches in finishing – examples are CAAT-Box [63], where it is possible to annotate the genome during the finishing process, and MGView [64], which simplifies finishing of microbial genomes by allowing for comparisons with already sequenced genomes during the finishing phase.

1.3.6 Shotgun Sequencing of large genomes

Two different strategies for shotgun sequencing of large genomes have been used extensively in the past ten years, hierarchical shotgun (HS) and whole genome shotgun (WGS). In the HS strategy, sometimes referred to as "BAC walking" or "clone-by-clone sequencing", the genome is mapped beforehand, and a minimal set of overlapping clones, usually bacterial artificial chromosomes (BACs), are sequenced and assembled individually, and finally merged. In the WGS approach, the whole genome is randomly fragmented, and the assembly is performed in a single step (figure 1.4).

The main advantage of WGS is that it is considerably less expensive than HS. However, systematic errors, e.g. contamination, have global effects in WGS and can remain undetected until after *fait accompli*, as the project reaches the assembly stage. In HS, such complications can be detected at an early stage for individual BACs, thus having local effects only. Similarly, repeated regions have a global effect in WGS, complicating assembly to a higher degree than in HS. HS, WGS and the repeat problem will be described in more detail in section 1.4.2.

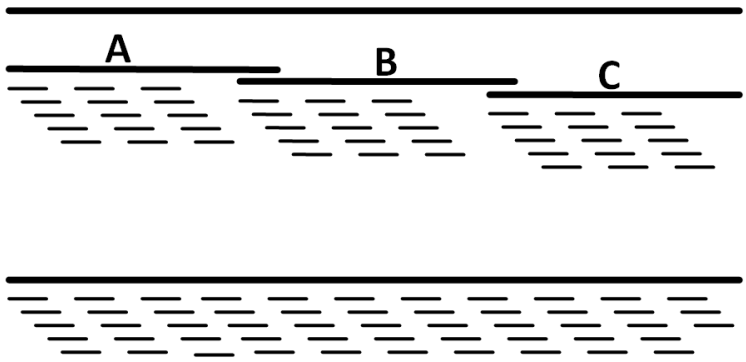

Figure 1.4: Hierarchical shotgun (HS, top) and whole-genome shotgun (WGS, bottom). In HS, overlapping BACs (A, B, and C) covering the genome are sequenced, assembled individually and subsequently joined. In WGS, the whole genome is randomly sequenced and the assembly is performed in one step.

1.4 The repeat problem in shotgun sequencing

1.4.1 Repeats and the assembly algorithm

The key problem in shotgun sequencing is the presence of repeated sequences in the target sequence, and the problem is purely computational. In the generic assembly algorithm described above, repeated sequences cause severe problems in the overlapping phase. The whole idea of shotgun sequencing is based on the assumption that a pair of reads that overlap sample the same part of the target sequence. For repeated regions, this is not true. Figure 1.5 illustrates the problem. When identical or near-identical sequences are present at several locations in the genome, reads sampling these regions appear to overlap. When the repeat units are organized in tandem arrays as described above (section 1.2.4), the result is usually that the repeat units are collapsed into a smaller number of copies. In the assembly, these regions are characterized by an unusually high coverage of the region. When a consensus sequence is computed for the region, the sequence of the collapsed copies will be a mixture of the different repeat units present in the genome, and the resulting consensus sequence of a given unit may not even exist in the target genome at all.

Moreover, if the repeat units are dispersed throughout the genome, large artificial rearrangements may occur due to the greedy nature of the assembly algorithm. When the assembly extends from unique sequence into the repeat, and the next read for inclusion is chosen from the list of high scoring overlaps, exactly which repeat unit the read actually samples becomes a matter of pure chance. If the repeat unit is longer than a read length, any read sampling any of

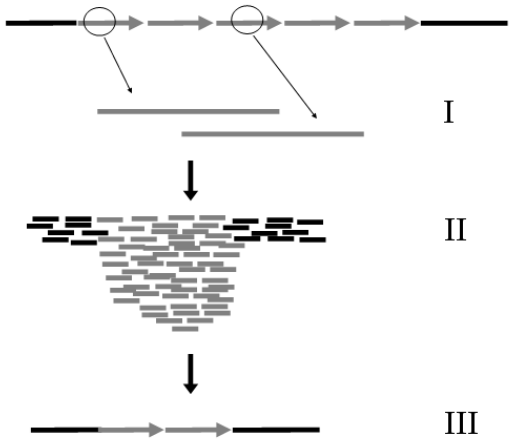

Figure 1.5: Misassembly of repeated sequences. Gray arrows indicate repeat copies in the target sequence, gray bars indicate sequence reads sampling the repeat region, black bars indicate reads sampling unique parts. I. Sequence reads sampling different repeat copies appear to overlap. II. The resulting assembly is erroneous, piling reads from different repeat copies. III. The consensus sequence is erroneously computed, with repeat copies merged.

the repeat copies at the same coordinate in the repeat unit will match nicely as the assembly progresses into the unit. The assembly coverage in this area will typically be very high, as in the tandem example above, and as the assembly approaches the end of the repeat unit, reads bridging into unique sequence will be incorporated at random. Well past the repeat border, the assembly proceeds normally, but there is a high risk that two completely disparate regions of the genome have been connected (figure 1.6).

The problem with rearrangements can in many cases be solved with the use of mate pairs and other paired end sequences, such as BAC ends. Using this kind of information, it is possible to detect problematic areas since mate pairs will be erroneously positioned in relation to each other in repeat regions. Such positions in an assembly can be broken, and the mate pairs can be used to rearrange the pieces correctly. However, if the repeat region is longer than the

average shotgun fragment, mate pairs cannot be trusted either, since a region of the repeat will exist where neither read in any mate pair is anchored in unique sequence.

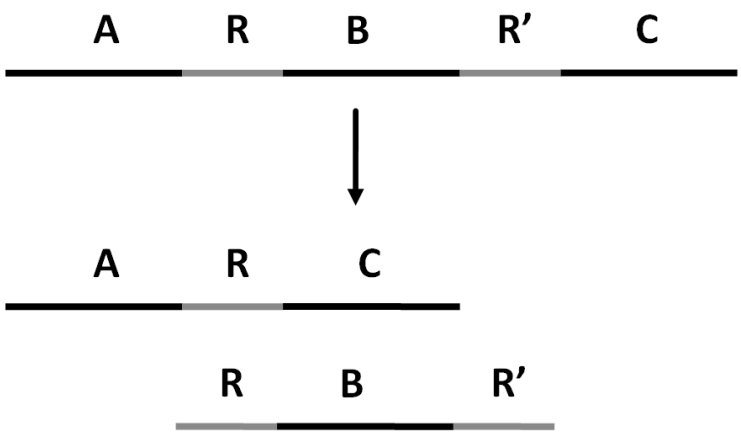

Figure 1.6: Misassembly due to highly similar repeats. A, B, and C are different segments of the target genome, connected by highly similar repeats R and R'. In assembly, the segments are erroneously joined, producing two incorrect contigs.

1.4.2 HS, WGS and Repeats

When the private initiative to sequence the human genome indepently from the public effort was announced in 1998 by Craig Venter's newly formed company Celera Genomics [65], the company stated that it would use the WGS strategy. Apart from the obvious competition with the public initiative, the choice of strategy was quite controversial and had been preceeded with a few years of debate regarding the merits of the WGS approach for higher eukaryotes [66, 67, 68, 69]. While the main arguments for the WGS approach are time and money – Venter and colleagues estimated the total cost of their effort to one tenth of that of the public initiative (300 million vs. 3 billion USD), and they also projected to finish faster – the strongest arguments for the HS approach is that it produces more accurate sequence. This is mainly because the effect of repeats is compartmentalized to distinct BACs, as opposed to having an impact on the whole assembly. Another argument against WGS that was brought forward was that other kinds of errors, such as systematic contaminations at different labs or data tracking errors, could not be detected until in the assembly step, where it

was too late. These kinds of errors will only have a local effect in an HS scheme, while they affect the entire project in the WGS approach.

After publication of both draft sequences, the debate arose again [70, 71, 72, 73, 74]. Celera's assembly was not performed using Celera WGS data only; it was completed with "perfectly shredded BACtigs", i.e. the consensus sequence of the public BACs was divided into uniformly covering "faux reads", spanning the genome at 2X, that were added to the Celera data. Critics claimed that the adding of this data preserved the public layout for all intents and purposes, and basically rendered the Celera data unnecessary. Celera strongly refuted this, claiming that the public data had little or no effect on the overall layout, and in some respects made the Celera assembly more difficult.

At present, there is wide consensus that the WGS approach alone is not entirely suitable for mammalian sized genomes [75, 12]. The problems that repeats cause in WGS are now acknowledged also among the proponents of this approach, and a common opinion is that a combination of the two approaches is the most optimal – timewise, costwise and qualitywise [76, 77, 78]. A subsequent version [79] of the Celera assembler [43] was specifically designed for such combined data, and more recent assemblers Atlas [80] and FASSI [81] also take this approach.

1.4.3 Methods of dealing with repeats

Repeats at the local level

The most common way of dealing with repeats in sequencing projects (and assembly programs, to some extent) is to try to detect which reads sample repeated regions, and to exclude these reads from the dataset. In other words, repeats are dealt with by deciding not to deal with them. This clearly has computational benefits, since the assembly process is mostly straightforward in the absence of repeats. Moreover, the presence of repeats leads to significantly increased running times and memory requirements.

Most assembly software performs an initial comparison of all reads against known repeats in the target genome. This can be centromeric and telomeric sequence, retrotransposons, ribosomal DNA etc. Parts of reads matching known repeats are masked and not used in subsequent steps. For the Celera Assembler [43], this is the only type of repeat handling at the read level that is performed.

PCAP [40] attempts to discard reads from repeat regions from initial assembly by computing an overlap coverage for each read, and excluding parts of reads that have an overlap coverage exceeding a threshold. ARACHNE also uses a similar scheme, where reads containing exact k-tuple matches to an extent above a threshold are excluded.

Other assembly algorithms generally feature a step in the overlapping phase, where attempts are made to weed out false overlaps due to repeats. Phrap computes a log likelihood ratio (LLR) score for each pairwise overlap, where the overlaps are screened for mismatches. Quality values are used to calculate the probability of mismatches being due to sequencing errors, or due to single base

differences between repeat units occurring at a predefined rate (the default is 95% similarity). ARACHNE [41, 42] uses a similar method to remove false overlaps between reads that weren't detected in the k-tuple screening (see above).

A major difference between the PCAP and Phrap approaches, is that while PCAP attempts to exclude repeat *reads* from the dataset altogether, Phrap tries to discard repeat *overlaps*. This is an important distinction, since the exclusion of reads has the consequence that the sequence they sample will not make it into the assembly at all. By keeping the reads and attempting to pinpoint the false overlaps, Phrap at least has a theoretical chance to perform a realistic reconstruction of the target sequence. It should be noted, however, that for repeats more similar than 95-98%, Phrap is usually not able to separate them correctly, resulting in erroneous and unreliable assemblies.

The EULER assembler [46, 82, 83] takes a completely different approach to assembly and thereby also repeat handling. Instead of the traditional overlap-layout-consensus method, EULER transforms the assembly problem into an Eulerian path problem by dividing all reads into overlapping k-tuples that are collapsed, treating all $(k-1)$-tuples as vertices in a graph where the edges then represent links between $(k-1)$-tuples, i.e. k-tuples. Repeated sequences appear as "tangles" in the graph. The problem then becomes to visit all edges in the graph exactly once, which can be done in linear time. Problems with this approach are that it requires error free data, and that the division of reads into k-tuples discards important positional information needed to resolve repeats. These problems can partly be solved by inclusion of error correction algorithms, the aligning of reads to the graph in order to resolve tangles, as well as the use of paired end sequences (mate pairs) for further repeat resolution. Although the authors of EULER claim that it solves the repeat problem, repeats longer that the shotgun fragment insert length cannot be reliably resolved using this method. However, a great benefit of the EULER algorithm is that it retains a (basically) complete but significantly less complex version of the overlap graph, and pinpoints repeated regions in a stringent and intuitive fashion.

Repeats at the global level

After initial assembly, a common strategy is to analyze contigs for unusually high read coverage, and/or the presence of reads that match the consensus sequence only in part. Finishing software (see section 1.3.5) can subsequently be used to try to resolve repeated contigs. This is a very time consuming and tedious process though, often requiring additional mapping and sequencing if the goal is to produce completely accurate sequence – see [10] for an example on how a 3 kb sequence, repeated four times interspersed by short tandem repeats, on chromosome 11 in the human genome was resolved. A common way to handle these kinds of problematic areas in the genome is to acknowledge their existence, and exclude them from further analysis and finishing. In the HS approach, repeats will only have local effects anyway, provided that the tiling of BACs across the genome is correct.

In the WGS approach, repeats cause large, erroneous rearrangements of

genomic sequence in addition to local problems such as collapsed sequence. The solution to this is usually an iterative approach where mate pairs and other paired end sequences are used in a heuristic fashion. Contigs and scaffolds are broken and rejoined on the basis of paired ends, until no discrepancies exist. In some programs (e.g. [40, 43]), repeat reads that were previously discarded are assembled and used for gap closure. Separate software for mate pair analysis and detection of erroneous rearrangements has also been developed [84, 85].

Chapter 2

Present investigation

The aims of this study were to develop algorithms and software for separation of highly similar repeats in shotgun sequencing, and to apply these methods to complex repeat regions that common assembly methods are incapable of resolving. Paper I describes a method for detecting single base differences between repeat copies, as they appear in shotgun fragment reads distorted by sequencing errors. In paper II, the method was implemented in a finishing and analysis tool specifically designed for repeat data. This tool was used in paper III, where five repeated regions of the *Trypanosoma cruzi* genome were studied.

2.1 Paper I – Separation of nearly identical repeats in shotgun assemblies using defined nucleotide positions, DNPs

2.1.1 Main problem

Erroneous assemblies produced by commonly used assembly programs are often caused by false overlaps between reads sampling different copies of a repeat region. If the repeats are too similar, overlap methods will often fail to identify false overlaps even when the reads are sampling unique sites that distinguish one repeat copy from another, because of the problems of identifying these sites in the presence of sequencing errors. Moreover, even if the sequencing procedure was completely error free, and overlaps thus could be discarded on the basis of single mismatches along the overlap, this would still be no guarantee against misassemblies. This is due to the greedy nature of the layout algorithm as described in section 1.3.3, and ultimately means that completely identical repeats are inseparable, except under certain, limited conditions with the help of paired end sequences.

However, if the repeats are not completely identical and instead contain subtle differences, it should in theory be possible to assemble them correctly, provided that they can be detected and that enough differences exist so that

it is possible to sample more than one difference with one read. Figure 2.1 illustrates this principle; if an assembly has extended into a repeat region, it is necessary for the differences to be spaced in a fashion that a read always can be unambigously positioned. This requires two anchor points; one to anchor the newly inserted read, and one to anchor further reads that extend the contig.

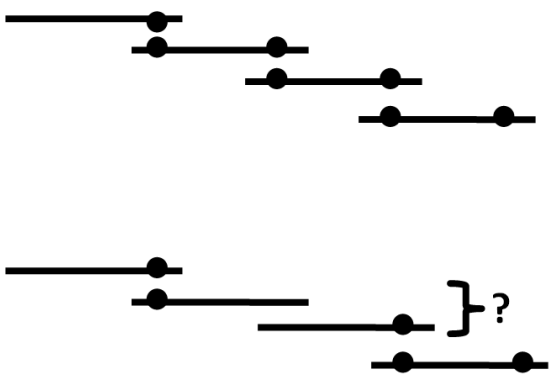

Figure 2.1: For correct assembly of repeats, detected differences must be explicitly used. Bars indicate reads; dots indicate detected differences. If no difference is present along the alignment, it is impossible to determine which reads belong together even if they overlap.

Although adding the above heuristic to the generic layout algorithm is fairly simple, no current assembly program acknowledges this distinction between discarding overlaps and explicitly utilizing single base differences in assembly. However, similar heuristics have recently been used for "shotgun haplotyping" [86].

This lack of explicit identification and use of single base differences in assembly was the primary motivation for the work presented in paper I. It should be noted that requirements on specificity (here defined as the proportion of positions marked as true differences that actually are true) are high, if the purpose is to use single base differences for assembly in the fashion described above. Since the layout algorithm is greedy also with the extra requirement of matching bases at unique sites, the incorrect incorporation of a single read at the wrong position may render the assembly invalid. Similarly, requirements on sensitivity (defined as the proportion of true differences that are actually detected) are also high, since failure to detect enough differences will lead to insufficient amounts of the anchor points needed to extend contigs within a repeat region.

2.1.2 Analyzing multiple alignments instead of pairwise overlaps

Paper I describes a method for detecting single base differences between repeat copies, as they appear in reads sampling individual repeat copies in the presence of sequencing errors. The ideas presented in the paper came out of a simple, intuitive notion that, while it is hard to determine whether a mismatch between two reads that appear to overlap represents a sequencing error or a real difference between repeats, the real differences are trivial to spot when looking at an alignment of reads, since they appear as the same consensus-deviating base multiple times on a column in the alignment. In other words, sequencing errors should appear randomly distributed in an alignment, whereas single base differences are distributed in a systematic fashion. This idea has also been exploited in SNP finding software such as PolyBayes [87] and PolyFreq [88].

2.1.3 One column

The first approach we tried was to use the error probabilities of individual bases to compute the expected number of sequencing errors on a column in the alignment, and to compare that to the number of deviations from consensus that are observed on the column. The number of sequencing errors on a column are distributed according to Poisson statistics, and by integrating the Poisson distribution for a given expectation value (given by the base error probabilities) from the observed number of deviations from consensus to infinity, the total probability P_{obs} of observing that number of sequencing errors or more by chance is obtained. A suitable threshold can be set, e.g. $P_{obs} \leq 0.001$, to reach arbitrary specificity.

The problem with this approach is that, under realistic conditions of coverage and repeat copy numbers, the distribution of sequencing errors greatly overlaps the distribution of true single base differences we expect to see on a true column. This number follows the distribution of shotgun coverage, which is also a Poisson variable. The overlap between distributions is basically not a problem if the number of expected errors on a column is close to 0. However, inherent in the repeat problem lies the fact that alignments of apparently overlapping reads will be very deep, which increases the expected number of sequencing errors, and pushes the Poisson distribution to the right. Consider the hypothetical situation in figure 2.2; 20 repeat copies sequenced at coverage 5, with a mean error rate of 0.01 for each base on the column, will give one expected error and 5 expected deviations from consensus. The distributions for these expectation values overlap, and in order to maintain a high specificity, roughly half of the true differences have to be discarded.

2.1.4 Two columns

We present a solution to this problem in paper I, where instead of computing the expected number of sequencing errors on one column, the expected number

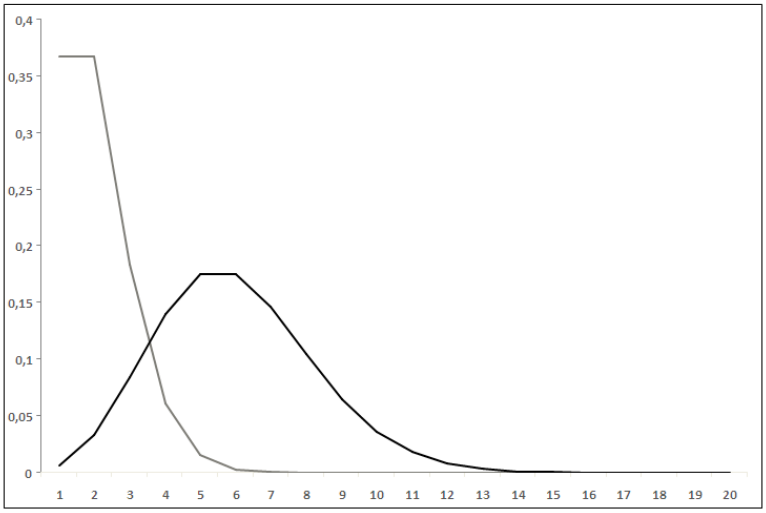

Figure 2.2: The distributions of sequencing errors (left) and number of true differences (right) on a column with 100 bases (20 repeats sequenced at coverage 5), each having an error probability of 0.01.

of coinciding deviations on two columns is computed, i.e. the expected number of reads differing from the consensus in the same two positions. It turns out that this variable is also Poisson distributed, but with a significantly lower expectation value, effectively pushing the distribution to the left, and giving it a shorter tail. The distribution of coinciding deviations also moves towards the left, but to a much lesser degree. Therefore, it is possible to maintain a high specificity, while keeping a high sensitivity. It could be argued that the requirement of reads sampling two single base differences in order for them to be detected imposes a limitation to this method. This may be true for other, similar problems such as SNP detection; for assembly purposes, two single base differences within a read length is required for proper assembly as described above.

Two variants were implemented and tested – the basic method and the extended method. In the basic method, different thresholds D_{min} of the number of coinciding deviations required for marking deviant bases as defined nucleotide positions (DNPs) were assessed with regard to sensitivity and specificity, with-

out additional statistical computation. In the extended method, the thresholds were maintained, with the addition that the probability of observing the observed number of coinciding deviations or more was calculated, and weighted with the probability of observing a high number of sequencing errors on either column. The weighting was added to minimize the effect of erroneous DNP calls due to a high number of sequencing errors on one column, which was a phenomenon we observed when developing the method.

The results showed, predictably, that the difference in specificity between the two methods decreases with increasing D_{min}. This is because setting a higher threshold has the exact effect of moving into a region where the distributions no longer overlap. The effect of quality trimming the reads more stringently has a similar effect, since eliminating low quality bases from analysis effectively moves the distribution of sequencing errors, and thereby also the distribution of coinciding deviations due to chance, to the left. We found that, by using the extended method in combination with less stringent quality trimming, it is possible to achieve higher specificity without decreasing sensitivity under generous trimming conditions.

2.1.5 Competition

Around the time of publication of paper I, three other methods were introduced that exploit multiple alignments, rather than read pairs, for discriminating sequencing errors from single base differences .

The first method, used in the ARACHNE whole genome shotgun assembler [41] considers one column at a time, locates deviations and evaluates them according to base error probabilities as in our first, one column approach.

The second, presented by Kececioglu *et al.*, [89] uses a k-star algorithm to locate and discard false overlaps in contigs with an unexpectedly high coverage characterized by repeat regions. The major difference between this method and our DNP method is that although the k-star algorithm analyzes correlating deviations from consensus in a multiple alignment, it only considers pairs of reads at a time. This is, in some respects, comparable to our method with the D_{min} parameter set to 2, which in our case yielded a very high sensitivity at the cost of very low specificity. The results in [89] were inconclusive, and the method was only tested on repeats differing 5% and 10%, similarity levels which are easily resolved by Phrap and other assembly software.

The third method also exploiting the multiplicity of single base differences in several reads at the same time was introduced in the error correction step of the EULER assembler [46]. In this method, all l-tuples occurring in the data set are analyzed for multiplicity, and considered *solid* if they occur in a multiplicity exceeding a threshold. Non-solid l-tuples are located and transformed into solid ones iteratively, using a minimal number of substitutions. This method does not consider base error probablilities at all, and is comparable to our initial one column approach without statistics. Indeed, in a subsequent publication (paper ii), we showed that the DNP method could be used for error correction in shotgun data, and that it outperforms the EULER method.

The advantage of the DNP method over these methods lies in the fact that the DNP method considers two columns in an alignment, and all reads covering these two positions, simultaneously. The ARACHNE and EULER methods consider all reads but one column at a time, whereas Kececioglu's method looks at multiple columns but for read pairs. Consequently, these methods suffer from the trade-off between specificity and sensitivity as described in section 2.1.3.

Since the publication of paper I, other multiple alignment based methods for detecting single base differences have emerged. I will here briefly mention two of them. The first [90] is completely heuristic and analyzes overlapping reads in groups of four, attempting to find pairs or triplets of columns on the basis of which the reads can be divided into two subgroups with complete agreement within each subgroup. Although this method seems promising for large scale overlapping before the layout stage, its merits for separation of nearly identical repeats are unclear and remain to be thoroughly assessed.

Another recent method that seems very promising if raw data is available considers one column at at time, but also performs a rigorous analysis of the chromatogram positions corresponding to deviant bases [91]. The authors propose a scheme where shotgun reads are assembled into contigs, whereafter analysis is performed and errors corrected. This is performed in an iterative manner, reassembling the data after each round. A similar approach is used in the miraEST assembler [92], although the details of the trace data analysis in this method is not disclosed.

An alternative to the iterative strategy could be to analyze all overlapping reads before the layout stage, much like in our DNP method, to avoid overlooking undetectable assembly errors and many iterations. Such an approach would probably increase sensitivity but at high costs of running time, since initial construction of alignments for all reads is very time consuming.

2.1.6 Subsequent use of the DNP method

The basic version of the DNP method was implemented in a prototype assembly program (TRAP, paper i), where proof of principle for the extra heuristic for treatment of repeat reads as decribed in section 2.1.1 also was shown.

The extended version was implemented in software for error correction (MisEd, paper ii), tagging of .ace-files produced by Phrap (ReDiT, paper iii), and DNPTrapper – the repeat analysis and finishing tool described in paper II.

2.2 Paper II – DNPTrapper: an assembly editing tool for finishing and analysis of complex repeat regions

2.2.1 Main problem

The major bottleneck in sequencing projects is the finishing phase. Some finishing tasks, such as pinpointing where additional sequence is needed, are relatively straightforward for non-repeated sequence and can be automated to a large extent [60]. However, the problematic regions caused by repeats cannot be resolved using automatic methods and need to be manually inspected and curated. This is time consuming and costly, with the result that virtually all sequencing projects except bacterial genomes are left in an unfinished state.

A number of different tools have been developed for aiding manual finishing of complex regions. Most commonly used are Consed and the Staden package [61, 62]. Again, these tools prove very useful for non-repeated sequence, e.g. when dealing with regions of low sequence coverage or low sequence quality due to high GC content, but have shortcomings when it comes to dealing with repeats. The tools are not specifically designed with the repeat problem in mind, which is somewhat of a paradox as the greatest obstacles in finishing are due to repeated regions in the target sequence. Consequently, repeats are left out of the entire assembly and finishing stages in many projects, deemed as unsolvable.

Commonly used finishing tools have three major deficiencies when it comes to repeated sequences. First, they lack the necessary overview that is needed for analysis of repeat regions. Assembly finishing software generally offers a view at the project level, where the relationship between different contigs can be examined, and a view at the contig level, zoomed in on the alignment of individual contigs. There is nothing in between, which is a problem since repeat reads typically are assembled into contigs with very deep coverage. At a fixed zoom level, the user has to scroll up, down, left and right in order to get a survey of the region at hand, and has to analyze different parts of the alignment separately.

A second feature of common finishing tools that decrease their usefulness when dealing with repeats is their lack of flexibility. Although they often allow correction of erroneous base calls and and removal of mistakenly included reads from individual contigs, it is not possible to divide or manipulate the global alignment by moving reads around as the user sees fit. Modifying the layout of reads generally requires identifying false overlaps, instructing the software not to use them, and rerunning the assembly in hope of an improved result – which by no means is guaranteed.

Third, no current finishing tools have reliable methods of detecting DNPs in reads sampling single base differences between repeat units. Features for highlighting high quality mismatches against the consensus are usually included, but as was shown in paper I, with this information the user has to make a choice

between specificity and sensitivity, which makes the task of separating repeats more complicated than it has to be.

2.2.2 DNPTrapper

We developed an assembly analysis and finishing tool, DNPTrapper, that is described in paper II. The idea was conceived when we were asked by Najib El-Sayed at The Institute for Genomic Research (TIGR) to take a look at a BAC from the *T. cruzi* sequencing project, which contained an 8 kb sequence repeated in tandem, with an additional short repeat within the repeat at the end of the large unit. This type of arrangement is very common in the parasite, and since *T. cruzi* shotgun data was produced also at our laboratory, we were already familiar with the problems of traditional finishing software for analysis of repeat regions. Sensing a general need for finishing software developed specifically for repeats, we developed prototype software that in its original form had the three key features of what later became DNPTrapper: overview, flexibility and DNP detection and visualization. Alone, these are no revolutionary or novel concepts, but combined they allow for effective resolution of repeats.

Overview

Overview is achieved by the simple feature of zooming out of the contig to a level that enables simultaneous overlook of large parts of the alignment. In contrast, other finishing tools have a fixed view at the contig level, which is set to where individual bases in reads are readily intelligible. While this is very useful for detailed analysis and editing of individual reads, it makes it very cumbersome to study the deep alignments typical for repeat regions. Scrolling back and forth ad nauseum is required to get a sense of what the region looks like, the depth of coverage, the length of the repeat unit etc. By allowing the user to zoom out, DNPTrapper enables easier understanding of the structure and layout of the repeat.

Flexibility

Common finishing tools are rigid in the way the user can modify the layout. It is reasonable to prevent manual moving of reads horizontally, since such actions break the integrity of the alignment. However, when analyzing a repeat region, it is quite useful to be able to move reads vertically, grouping reads that share similar features such as DNP content. DNPTrapper allows such drag and drop of reads, and also allows the user to cut, copy and paste reads as they see fit, as well as creating new contigs. This is useful for trying out different solutions and scenarios involving subsets of the reads sampling a repeat. Additional flexibilty is provided in that different operations and algorithms, such as exporting consensus sequence, sort according to DNPs, re-aligning reads, and locating mate pairs, can be performed on all sequences in a contig as well as on chosen subsets. The program also allows for horizontal movement of

reads, which is useful when trying to order different repeat groups horizontally using DNPs or mate pairs. Agreed, the use of these features imposes a higher requirement of know-how on the user – on the other hand, resolution of repeats in itself requires familiarity with fundamental properties of multiple alignments.

DNP visualization

DNPs are detected using the extended method as descibed in paper I, and are visualized using colored dots. A DNP is defined by the deviant base type on a column, combined with the consensus base type. There are twelve possible pairwise combinations of four nucleotides, and thus twelve different colors for DNPs. This makes it possible to spot DNP patterns by eye (figure 2.3), and to use these patterns to assign reads to different groups according to DNP content. DNPTrapper also features a naïve DNP sorting algorithm in which reads are sorted according to DNPs using a greedy method. This can be useful as a starting point for repeat resolution, decreasing the amount of drag and drop operations that need to be performed in the subsequent analysis.

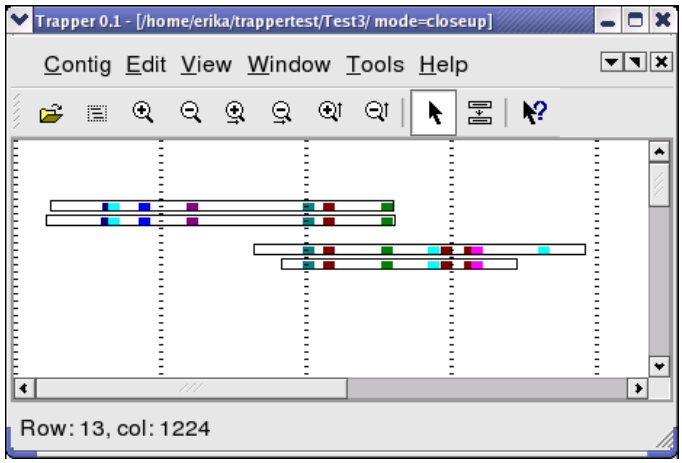

Figure 2.3: Zoomed in view of four reads with DNPs in DNPTrapper. Reads are represented by boxes, DNPs appear as colored dots. Each color represents one of twelve different DNP types.

Other features

Additional features of DNPTrapper include visualization of mate pairs, chromatograms, strand identity and quality trimming, as well as exporting to different file formats. These are all standard features of finishing software and were added for convenience. Future improvements may include additional features present in other programs; however, the main purpose of DNPTrapper is not necessarily to replace previous tools but rather to complement them.

Analysis of *T. cruzi* repeats

Apart from a simulated dataset which was included in paper II as proof of concept, two repeat regions from *T. cruzi* were analyzed in order to illustrate different kinds of repeat phenomena that can be observed using DNPTrapper. Two different repeated genes were studied, showing different characteristics. In the first, elongation factor 2 (EF2), the reads could be divided into two major groups, with no further reliable division into subgroups possible. For both groups, mate pair analysis showed that several mate pairs were located within the group, while no mate pairs with reads in both groups were observed. This strongly suggests that EF2 exists in two tandem arrays, one on each homolog, with the repeat unit conserved within a homolog and more divergent between homologs.

The second gene studied, monoglyceride lipase (MGL), showed completely different characteristics. Unlike EF2, numerous groups of reads were found, suggesting a number of different versions of the gene, present in different copy numbers.

If possible at all, performing these investigations using e.g. Consed would have been very cumbersome. The first and most time consuming part, which probably would take days or even weeks for a data set with many DNPs like MGL, would entail identifying which mismatches were due to sequencing errors and which ones described single base differences. This would require scrutinizing individual chromatograms in the absence of reliable DNP detection. The depth of the alignment for MGL (258) would require tedious scrolling up and down, and since some DNPs are present in several groups it would be virtually impossible to keep track of the combinations of DNPs that uniquely define each group. Since reads cannot be moved in Consed, the most reasonable approach would probably be an iterative one: choosing one or a few DNPs at a time, noting which reads have them, construct new data sets consisting of these reads and re-assemble them, choose other DNPs in the new contig and repeat the procedure. The result would be a large number of contigs, each representing a different repeat group. This result would then not include the immediate information on how the groups differ, which would have to be obtained by other tools.

In contrast, the division of reads from MGL into different repeat groups took little over an hour's work using DNPTrapper.

2.2.3 Subsequent use of DNPTrapper

Since its release, DNPTrapper has been downloaded approximately two times a week, indicating that it fills a void not covered by other finishing tools. In addition to being compatible with Phrap, the software is file compatible with AMOS (see Discussion and concluding remarks, chapter 3), and work is in progress to integrate DNPTrapper with its whole genome analysis tool Hawkeye.

During development, DNPTrapper was used to analyze two regions (heat shock protein 70 (HSP70) and cruzipain(CP)) for the *T. cruzi* WGS project([25], paper iv), as well as spliced leader sequences from *Leishmania major* (unpub-

lished). An additional five repeated genes of *T. cruzi* were analyzed and described in paper III.

2.3 Paper III – Database of Trypanosoma cruzi repeated genes: 20 000 novel coding sequences

2.3.1 Main problem

The genome of *Trypanosoma cruzi* is estimated to consist of >50% repeated sequences, and has many of its genes organized in tandem arrays [25]. Study of the structure, evolution and function of these repeats is crucial for a comprehensive understanding of the parasite biology, and may also be fundamental in developing drugs against Chagas' disease, caused by *T. cruzi*. For instance, the parasite's ability to avoid the host immune system is known to be closely coupled to the maintenance of multiple gene copies of surface antigens, subtly differing and expressed as needed.

Unfortunately, the repetitive nature of the *T. cruzi* genome combined with the decision to use the WGS approach in sequencing has resulted in an assembly which is far from complete, and erroneous in several places. In addition, the strain that was chosen for sequencing (CL Brener) is a hybrid between two polymorphic strains, which has complicated the assembly further. The assembler chosen for the task (Celera assembler, [43]) had severe initial problems in the assembly stage, largely due to its inability to handle polymorphic homologs, and had to be significantly reengineered before a reasonably acceptable assembly could be carried out. Still, approximately 26% of the sequenced reads did not make it into the assembly at all, indicating just how complex this parasite genome is.

In addition to an incomplete understanding of *T. cruzi* biology, the unfinished state of the project also has consequences that are purely practical. For repeated genes, there are no guarantees of correctness of the reported consensus sequence, since repeat copies may have collapsed resulting in a consensus which is a mixture of merged copies. This means that an annotated gene in the assembly may have a sequence that isn't represented in the true genome at all, which can have consequences for PCR and cloning of sequences that are chosen for investigation. The erroneous assembly of repeats also makes it difficult to distinguish between SNPs that are paralogous, allelic, or simply artefacts of misaligned reads.

2.3.2 Genome-wide and in-depth analysis of repeats in *T. cruzi*

In paper III, we describe a comprehensive study of repeated genes in *T. cruzi*, and provide an in-depth study of five genes at a more detailed level, highlighting different kinds of repeats that exist in the parasite. The genome-wide analysis

will only briefly be described here, since this author's main contribution was the in-depth study.

Previous to paper III, two investigations into *T. cruzi* repeats based on reads from the WGS project had been performed by Westenberger *et al.*, where two approaches were used to analyze rRNA genes and spliced leader sequences: phylogeny [93] and linkage [94] analysis. While these approaches are reasonable for studying sequences that are shorter than or equal to a read length, they cannot be used for the longer sequences that most of the genes in *T. cruzi* consist of. Phylogeny and linkage studies both require overlapping sequences for analysis; polymorphisms separated by more than a read length cannot be assessed using these methods.

2.3.3 Genome-wide analysis

The primary reason for performing the comprehensive repeat analysis in paper III was to quantify and pinpoint the repeated genes, in order to enable further, close-up investigations of these regions in a reliable manner. All annotated genes were used as query sequences in a database consisting of all shotgun reads from the WGS project, including the reads not present in the annotated assembly. The reads were retrieved and aligned using GRAT, an in-house developed sequence similarity search tool. Copy number estimations were made based on the depth of alignments and the known shotgun coverage. Furthermore, all annotated coding sequences were collapsed at 95% sequence similarity level, in order to determine which annotations are highly similar and may be subtle variations of the same gene.

The analysis showed that repeated genes in *T. cruzi* range in function, including metabolism, cell growth, DNA and protein synthesis, transport, and surface antigens. The results were stored in a database available to the community, where it is possible to query genes of interest to find out their repeatedness. The purpose of the database is to enable studies as the one presented below. The coverage analysis also revealed that the number of coding sequences may be twice as many as previously estimated. This gives an indication of how much information is missing from the published genome of *T. cruzi*.

2.3.4 In-depth study

Conserved tandem arrays divergent between homologs

Five regions were chosen for further analysis using DNPTrapper. The two first, tyrosine aminotransferase (TAT) and flagellar calcium binding protein (FCB), showed similar characteristics as those described for EF2 in paper II and HSP70 and CP in paper iv, namely conserved repeat arrays on each homolog, divergent between homologs. However, unlike the previous regions, both TAT and FCB could be further divided into several subgroups on each homolog. Paralogous and allelic non-synonymous SNPs in the coding region were analyzed using SIFT [95], which predicted no function-altering amino acid polymorphisms.

Conserved gene

The analysis showed that heat shock protein 85 (HSP85) is extremely conserved, with few occurrences of single base differences in the intergenic region and even fewer in the coding region. All of the differences were in synonymous locations, indicating that there is strong selective pressure to keep this gene intact in *T. cruzi*. Due to the lack of DNPs, it was not possible to assign the reads to different homologs based on this data alone. It is likely, however, that HSP85 indeed is present in two arrays on different homologs like the regions described above – in the annotated genome, which can be accessed and browsed at GeneDB [96], there are three annotations of HSP85, one of which is present in a short contig where HSP85 is flanked by a hypothetical protein. A highly similar (98%) hypothetical protein flanks HSP on one of the remaining two, long contigs and it is reasonable to assume that the short contig is constructed from reads that due to assembly problems did not make it into the larger contig.

Surface antigen

Trans-sialidase (TS) is annotated at 1 430 locations in the *T. cruzi* assembly. The different versions are highly divergent, and only a subset of twelve variants have been identified as having trans-sialidase activity, containing a critical tyrosine at a specific site. A Tyr → His substitution at this position in the protein has previously been shown to inactivate it [97], but this histidine is not present in any of the inactive annotations in the assembly. Trans-sialidase has also been proposed as having a role in host infection, as it is an antigen for cell receptors [98].

In the DNPTrapper analysis, twelve read groups with different DNP patterns were found that had reasonable coverage over the whole gene, three of which contained the histidine previously not observed in the assembly. The remaining nine contained the tyrosine crucial for activity. Out of these nine, five groups had consensus sequences not in perfect agreement with any of the previously annotated active copies. This is thus a case where consensus sequences in the assembly are erroneous, probably due to collapsed repeat copies.

Hypothetical protein

Finally, a gene that is being studied at our laboratory in a proteomics project was chosen for analysis. Mass spectrometry has shown that it is expressed in the epimastigote stage of the parasite, and it has been annotated as one of many hypothetical proteins. It is similar to transporter proteins in closely related *L. major* and several other species, and contains several potential transmembrane regions, as predicted by Phobius [99].

40 different DNP groups were identified, 17 of which had good shotgun coverage over most of the coding region. Out of the 46 amino acid changes resulting from single base differences that could be identified, 35 were located in the transmembrane regions.

2.3.5 Conclusions

The in-depth study shows that repeated genes in *T. cruzi* are organized in different ways. Some are highly conserved, while others contain numerous paralogous as well as allelic SNPs. Many of these features are hidden in the assembly, and are impossible to identify using publicly available data. This can lead to nasty surprises and unexpected experimental complications for those intending to study these regions in more detail, especially as the *T. cruzi* genome browser at GeneDB contains no information on depth of shotgun coverage at specific locations in the genome, and little other information that might indicate problematic regions due to repeats apart from links to locations of similarly annotated genes. A database such as the one presented in paper III, along with access to shotgun reads and the use of analysis tools like DNPTrapper, allows for additional information to be obtained *in silico* before embarking on expensive, experimental *in vitro* endeavours which may otherwise be severely slowed down by using incomplete and inaccurate data as the starting point. In itself, the database contains valuable information, but it is especially powerful combined with DNPTrapper, as the combination allows anyone interested in a specific repeated gene to analyze it thoroughly. This scheme is applicable to any other genome, and similar efforts should be carried out for other genome projects in order to enable studies of complex regions that remain unfinished.

Chapter 3

Discussion and concluding remarks

As more and more genomes are being sequenced, it is increasingly apparent that important information is lost in regions left unfinished [100, 101]. Apart from problems caused by unclonable sequence, the major problem in sequencing is computational and caused by highly similar sequences appearing at numerous locations in the target sequence.

Novel approaches to sequencing emerge, closely followed by novel variants of repeat problems. The recent concept of metagenomics [102], where entire communities appearing in distinct ecosystems, such as soil or deep water, are sequenced simultaneously, carries with it its own set of problems related to sequence homology within and between species. Sequencing of highly heterozygous organisms is another example where special consideration has to be taken to repeats [103], as is comparative sequencing [104].

Another emerging trend is the move towards more high-throughput methods producing short reads without mate pairs. In order to achieve the next major sequencing goal – the "$1 000 genome", where the human genome of individuals can be sequenced from scratch at a reasonable cost – throughput needs to improve several orders of magnitude. The cost per finished base is in 2006 approximately $0.01, which means that the cost of sequencing needs to decrease by a factor of 30 000 before this dream comes true. Consequently, there is an increasing interest in methods such as MALDI-TOF, pyrosequencing and sequencing by hybridization, where large amounts of short reads (<200 nt) can be produced at low costs. Companies such as 454 Life Sciences [105] and Solexa [106] have recently launched platforms able to produce millions of bases in a few hours, with the aim set to billions. However, the repeat problem is increasingly severe with shorter read lengths [107], and a combination with traditional Sanger sequencing may ultimately be required [108].

These new sequencing approaches and technologies will require further development of previous methods, as well as novel approaches to repeat handling.

Instrumental in developing such methods is access to reliable, finished assemblies against which newly developed algorithms can be benchmarked. So far, the lack of reliable tools for evaluation of different strategies has been a serious obstacle in the efforts of solving the repeat problem. This is inherent in the problem; since most complex regions are left unfinished due to the repeat problem, there is no gold standard against which a novel method can be benchmarked. Software for generating random, genomic sequence (e.g. GenRGenS [109]), and programs that simulate the shotgun procedure (e.g. GenFrag [110]) are of great help in the development of improved algorithms, but still fall short compared to real data. The need for benchmark datasets has been increasingly recognized and they are emerging (see e.g. http://www.tigr.org/tdb/benchmark), along with tools for assembly-to-assembly comparisons [111]. The Assembly Archive [112], where the read layout can be deposited in addition to the consensus sequence and raw data, may also become an important resource for development of assembly algorithms in the future.

Another promising development is the emergence of Open Source assembly software. Two major projects are in the workings, AMOS (A Modular Open Source-Assembler, http://amos.sourceforge.net) and BOA (Berkeley Open Assembler, [113]). This may have great benefits to the community, since it will promote modularity in problem solving – with open access to the source code of an entire assembler, it becomes possible to focus on particular problems without having to implement all the other parts required in assembly.

To conclude, repeats still remain a major and important problem in sequencing, which will require continuous development and improvement of methods such as those presented in this thesis.

A complete understanding of the biology of humans and other species will require precise knowledge of the genome, down to the last nucleotide.

Repeated or non-repeated.

Acknowledgements

A number of people have contributed to this work, directly and indirectly. I am particularly grateful to:

Björn Andersson – my supervisor. Thanks for giving me the opportunity to do this, for letting me pursue my own path, for guidance when needed, for sending me across the Atlantic Ocean and to South East Asia, and for always keeping calm.

Martti Tammi – my co-supervisor, co-author, and friend. You're the reason I got into this business in the first place, and the reason I followed through. Thanks for the continuous support, inspiration, and all the (sometimes) crazy ideas. We've been out on some wild goose chases, but sometimes we struck gold.

Daniel Nilsson – your encyclopedic knowledge about, well, EVERYTHING is only surpassed by your curiosity. Thanks for all the support, the philosophical discussions, the friendship, and the willingness to drop everything at hand to answer any questions, none of which remains unanswered when you embark on your googling expeditions.

Ellen Kindlund – always reliable, always thorough, always smart. Nothing escapes your sharp intellect and your ability to see through flawed reasoning. It's been a pleasure working with you – good luck come springtime!

The rest of the group, past and present: **Shane McCarthy** – one of the funniest and probably THE brightest person I've met. Good luck in NYC, you'll accomplish great things no doubt. **Carole Branche** – the lab got a lot more quiet when you left, and I really miss your laughter (and cursing in French). Thanks for the ultra-quick tour of Paris. **Marcela Ferella** – thanks for all the kindness, parties, and laughs. Hang in there! **Stephen Ochaya** – thanks for interesting discussions and perspectives. **Hamid Darban** – always the cool guy, always funny. **Daryoush Rahmani** – likewise, and thanks for all the computer help. **Staffan Alveteg**, **Erik Sjölund** and **Anders Florén** – thanks for skilled programming and nice discussions. **Anh-Nhi Tran** – thanks for helping me bring biology into the equation, and for your good mood – it's contagious. **Mia Blomqvist** – I'd hate to be on the receiving end of your biting and to-the-point sarcasms. Kudos, and good luck finishing! **Alan, Esteban, Sindy, Yumi, Delal, Johan, Lennie, Fang**, and all the others that came and went, thanks for adding to the group atmosphere.

Additional co-authors: **Fatima Farzana** – thanks for struggling with the repeats, asking challenging questions, and offering a perspective completely dif-

ferent than mine. I wish you the best of luck, although I know you won't need it to achieve greatness. **Tom Britton** – thanks for adding your expertise and knowledge in statistics to my project.

People at CMB and former CGB: department heads **Claes Wahlestedt, Tomas Perlman** and **Christer Höög** – thanks for providing a stimulating workplace. **David Fredman** – thanks for the laughs, for introducing me to your motley crew of friends in the music biz, and for trying to get me into shape. It failed the minute you left. **Rikard Dryselius** – it was great running into my old high school buddy here. See you in Japan. All the other graduate students, post-docs, group leaders and other staff that made this place great: **Emily, Geert, Markus, Pär, Lukas, Abhiman, Ivana, Camilla, Elin, Hagit, Omid, Abbas, Fredrik, Alistair, Albin, Timo, Carsten, Liam, Boris, Vivian, Gitt, Brittis, Christine, Zdravko, Iréne, Pierre, Bent** and numerous others whose names elude me at this late hour. **Matti Nikkola** – thanks for your support and for always making sure that everything is in order. **Kim Andersson** – thanks for great organizational skills (especially at "Solbacka"), friendship and for bringing **Maddie** here to brighten my day.

The Singapore crew: **Sarathi, Tina, Justin, Rahul, Hon Cheng, Xie Chao** and others – thanks for making my stay in Singapore extra enjoyable.

Other people: **Lasse Johansson** – thanks for scrutinizing my thesis. Shout outs also to all my friends (especially the Golberg posse) for all the support, fun and insanity over the years. Although I probably lost a significant number of brain cells along the way, I think the net benefit has been positive.

My family: **Anna**, my mother – thanks for always believing in me and looking out for me. If it was up to you, I'd be a professor by now. My father **Peter** – your support, guidance and knowledge in the world of science have been invaluable to me. **Karin**, my sister – thanks for always being on my side. I'd also like to thank my grandfather **Ored** – always showing a keen interest in my work – and my grandmother **Lisa** – the most generous person I know.

Finally, my girlfriend **My Persson** – thank you for choosing me every day, for all the love and support, for always being there, for making me want to better myself. Neko to kuruma, baby.

Bibliography

[1] R. D Fleischmann, M. D. Adams, O. White, R. A. Clayton, E. F. Kirkness, A. R. Kerlavage, C. J. Bult, J. F. Tomb, B. A. Dougherty, and J. M. Merrick. Whole-genome random sequencing and assembly of Haemophilus influenzae Rd. *Science*, 269(5223):496–512, Jul 1995.

[2] E. S. Lander, L. M. Linton, B. Birren, C. Nusbaum, M. C. Zody, J. Baldwin, K. Devon, K. Dewar, M. Doyle, W. FitzHugh, R. Funke, D. Gage, K. Harris, A. Heaford, J. Howland, L. Kann, J. Lehoczky, R. LeVine, P. McEwan, K. McKernan, J. Meldrim, J. P. Mesirov, C. Miranda, W. Morris, J. Naylor, C. Raymond, M. Rosetti, R. Santos, A. Sheridan, C. Sougnez, N. Stange-Thomann, N. Stojanovic, A. Subramanian, D. Wyman, J. Rogers, J. Sulston, R. Ainscough, S. Beck, D. Bentley, J. Burton, C. Clee, N. Carter, A. Coulson, R. Deadman, P. Deloukas, A. Dunham, I. Dunham, R. Durbin, L. French, D. Grafham, S. Gregory, T. Hubbard, S. Humphray, A. Hunt, M. Jones, C. Lloyd, A. McMurray, L. Matthews, S. Mercer, S. Milne, J. C. Mullikin, A. Mungall, R. Plumb, M. Ross, R. Shownkeen, S. Sims, R. H. Waterston, R. K. Wilson, L. W. Hillier, J. D. McPherson, M. A. Marra, E. R. Mardis, L. A. Fulton, A. T. Chinwalla, K. H. Pepin, W. R. Gish, S. L. Chissoe, M. C. Wendl, K. D. Delehaunty, T. L. Miner, A. Delehaunty, J. B. Kramer, L. L. Cook, R. S. Fulton, D. L. Johnson, P. J. Minx, S. W. Clifton, T. Hawkins, E. Branscomb, P. Predki, P. Richardson, S. Wenning, T. Slezak, N. Doggett, J. F. Cheng, A. Olsen, S. Lucas, C. Elkin, E. Uberbacher, M. Frazier, R. A. Gibbs, D. M. Muzny, S. E. Scherer, J. B. Bouck, E. J. Sodergren, K. C. Worley, C. M. Rives, J. H. Gorrell, M. L. Metzker, S. L. Naylor, R. S. Kucherlapati, D. L. Nelson, G. M. Weinstock, Y. Sakaki, A. Fujiyama, M. Hattori, T. Yada, A. Toyoda, T. Itoh, C. Kawagoe, H. Watanabe, Y. Totoki, T. Taylor, J. Weissenbach, R. Heilig, W. Saurin, F. Artiguenave, P. Brottier, T. Bruls, E. Pelletier, C. Robert, P. Wincker, D. R. Smith, L. Doucette-Stamm, M. Rubenfield, K. Weinstock, H. M. Lee, J. Dubois, A. Rosenthal, M. Platzer, G. Nyakatura, S. Taudien, A. Rump, H. Yang, J. Yu, J. Wang, G. Huang, J. Gu, L. Hood, L. Rowen, A. Madan, S. Qin, R. W. Davis, N. A. Federspiel, A. P. Abola, M. J. Proctor, R. M. Myers, J. Schmutz, M. Dickson, J. Grimwood, D. R. Cox, M. V. Olson, R. Kaul, C. Raymond, N. Shimizu,

K. Kawasaki, S. Minoshima, G. A. Evans, M. Athanasiou, R. Schultz, B. A. Roe, F. Chen, H. Pan, J. Ramser, H. Lehrach, R. Reinhardt, W. R. McCombie, M. de la Bastide, N. Dedhia, H. Blocker, K. Hornischer, G. Nordsiek, R. Agarwala, L. Aravind, J. A. Bailey, A. Bateman, S. Batzoglou, E. Birney, P. Bork, D. G. Brown, C. B. Burge, L. Cerutti, H. C. Chen, D. Church, M. Clamp, R. R. Copley, T. Doerks, S. R. Eddy, E. E. Eichler, T. S. Furey, J. Galagan, J. G. Gilbert, C. Harmon, Y. Hayashizaki, D. Haussler, H. Hermjakob, K. Hokamp, W. Jang, L. S. Johnson, T. A. Jones, S. Kasif, A. Kaspryzk, S. Kennedy, W. J. Kent, P. Kitts, E. V. Koonin, I. Korf, D. Kulp, D. Lancet, T. M. Lowe, A. McLysaght, T. Mikkelsen, J. V. Moran, N. Mulder, V. J. Pollara, C. P. Ponting, G. Schuler, J. Schultz, G. Slater, A. F. Smit, E. Stupka, J. Szustakowski, D. Thierry-Mieg, J. Thierry-Mieg, L. Wagner, J. Wallis, R. Wheeler, A. Williams, Y. I. Wolf, K. H. Wolfe, S. P. Yang, R. F. Yeh, F. Collins, M. S. Guyer, J. Peterson, A. Felsenfeld, K. A. Wetterstrand, A. Patrinos, M. J. Morgan, P. de Jong, J. J. Catanese, K. Osoegawa, H. Shizuya, S. Choi, and Y. J. Chen. Initial sequencing and analysis of the human genome. *Nature*, 409(6822):860–921, Feb 2001.

[3] J. C. Venter, M. D. Adams, E. W. Myers, P. W. Li, R. J. Mural, G. G. Sutton, H. O. Smith, M. Yandell, C. A. Evans, R. A. Holt, J. D. Gocayne, P. Amanatides, R. M. Ballew, D. H. Huson, J. R. Wortman, Q. Zhang, C. D. Kodira, X. H. Zheng, L. Chen, M. Skupski, G. Subramanian, P. D. Thomas, J. Zhang, G. L. Gabor Miklos, C. Nelson, S. Broder, A. G. Clark, J. Nadeau, V. A. McKusick, N. Zinder, A. J. Levine, R. J. Roberts, M. Simon, C. Slayman, M. Hunkapiller, R. Bolanos, A. Delcher, I. Dew, D. Fasulo, M. Flanigan, L. Florea, A. Halpern, S. Hannenhalli, S. Kravitz, S. Levy, C. Mobarry, K. Reinert, K. Remington, J. Abu-Threideh, E. Beasley, K. Biddick, V. Bonazzi, R. Brandon, M. Cargill, I. Chandramouliswaran, R. Charlab, K. Chaturvedi, Z. Deng, V. Di Francesco, P. Dunn, K. Eilbeck, C. Evangelista, A. E. Gabrielian, W. Gan, W. Ge, F. Gong, Z. Gu, P. Guan, T. J. Heiman, M. E. Higgins, R. R. Ji, Z. Ke, K. A. Ketchum, Z. Lai, Y. Lei, Z. Li, J. Li, Y. Liang, X. Lin, F. Lu, G. V. Merkulov, N. Milshina, H. M. Moore, A. K. Naik, V. A. Narayan, B. Neelam, D. Nusskern, D. B. Rusch, S. Salzberg, W. Shao, B. Shue, J. Sun, Z. Wang, A. Wang, X. Wang, J. Wang, M. Wei, R. Wides, C. Xiao, C. Yan, A. Yao, J. Ye, M. Zhan, W. Zhang, H. Zhang, Q. Zhao, L. Zheng, F. Zhong, W. Zhong, S. Zhu, S. Zhao, D. Gilbert, S. Baumhueter, G. Spier, C. Carter, A. Cravchik, T. Woodage, F. Ali, H. An, A. Awe, D. Baldwin, H. Baden, M. Barnstead, I. Barrow, K. Beeson, D. Busam, A. Carver, A. Center, M. L. Cheng, L. Curry, S. Danaher, L. Davenport, R. Desilets, S. Dietz, K. Dodson, L. Doup, S. Ferriera, N. Garg, A. Gluecksmann, B. Hart, J. Haynes, C. Haynes, C. Heiner, S. Hladun, D. Hostin, J. Houck, T. Howland, C. Ibegwam, J. Johnson, F. Kalush, L. Kline, S. Koduru, A. Love, F. Mann, D. May, S. McCawley, T. McIntosh, I. McMullen, M. Moy, L. Moy, B. Murphy, K. Nelson,

C. Pfannkoch, E. Pratts, V. Puri, H. Qureshi, M. Reardon, R. Rodriguez, Y. H. Rogers, D. Romblad, B. Ruhfel, R. Scott, C. Sitter, M. Smallwood, E. Stewart, R. Strong, E. Suh, R. Thomas, N. N. Tint, S. Tse, C. Vech, G. Wang, J. Wetter, S. Williams, M. Williams, S. Windsor, E. Winn-Deen, K. Wolfe, J. Zaveri, K. Zaveri, J. F. Abril, R. Guigo, M. J. Campbell, K. V. Sjolander, B. Karlak, A. Kejariwal, H. Mi, B. Lazareva, T. Hatton, A. Narechania, K. Diemer, A. Muruganujan, N. Guo, S. Sato, V. Bafna, S. Istrail, R. Lippert, R. Schwartz, B. Walenz, S. Yooseph, D. Allen, A. Basu, J. Baxendale, L. Blick, M. Caminha, J. Carnes-Stine, P. Caulk, Y. H. Chiang, M. Coyne, C. Dahlke, A. Mays, M. Dombroski, M. Donnelly, D. Ely, S. Esparham, C. Fosler, H. Gire, S. Glanowski, K. Glasser, A. Glodek, M. Gorokhov, K. Graham, B. Gropman, M. Harris, J. Heil, S. Henderson, J. Hoover, D. Jennings, C. Jordan, J. Jordan, J. Kasha, L. Kagan, C. Kraft, A. Levitsky, M. Lewis, X. Liu, J. Lopez, D. Ma, W. Majoros, J. McDaniel, S. Murphy, M. Newman, T. Nguyen, N. Nguyen, M. Nodell, S. Pan, J. Peck, M. Peterson, W. Rowe, R. Sanders, J. Scott, M. Simpson, T. Smith, A. Sprague, T. Stockwell, R. Turner, E. Venter, M. Wang, M. Wen, D. Wu, M. Wu, A. Xia, A. Zandieh, and X. Zhu. The sequence of the human genome. *Science*, 291(5507):1304–1351, Feb 2001.

[4] A. Bernal, U. Ear, and N. Kyrpides. Genomes OnLine Database (GOLD): a monitor of genome projects world-wide. *Nucleic Acids Res*, 29(1):126–127, Jan 2001.

[5] W. Gilbert. Why genes in pieces? *Nature*, 271(5645):501, Feb 1978.

[6] P. F. R. Little. Structure and function of the human genome. *Genome Res*, 15(12):1759–1766, Dec 2005.

[7] J. A. Shapiro and R. von Sternberg. Why repetitive DNA is essential to genome function. *Biol Rev Camb Philos Soc*, 80(2):227–250, May 2005.

[8] E. E. Eichler. Repetitive conundrums of centromere structure and function. *Hum Mol Genet*, 8(2):151–155, Feb 1999.

[9] E. E. Eichler. Recent duplication, domain accretion and the dynamic mutation of the human genome. *Trends Genet*, 17(11):661–669, Nov 2001.

[10] International Human Genome Sequencing Consortium. Finishing the euchromatic sequence of the human genome. *Nature*, 431(7011):931–945, Oct 2004.

[11] X. She, G. Liu, M. Ventura, S. Zhao, D. Misceo, R. Roberto, M. F. Cardone, M. Rocchi, E. D. Green, N. Archidiacano, and E. E. Eichler. A preliminary comparative analysis of primate segmental duplications shows elevated substitution rates and a great-ape expansion of intrachromosomal duplications. *Genome Res*, 16(5):576–583, May 2006.

[12] J. A. Bailey, D. M. Church, M. Ventura, M. Rocchi, and E. E. Eichler. Analysis of segmental duplications and genome assembly in the mouse. *Genome Res*, 14(5):789–801, May 2004.

[13] E. Tuzun, J. A. Bailey, and E. E. Eichler. Recent segmental duplications in the working draft assembly of the brown Norway rat. *Genome Res*, 14(4):493–506, Apr 2004.

[14] P. Stankiewicz, C. J. Shaw, M. Withers, K. Inoue, and J. R. Lupski. Serial segmental duplications during primate evolution result in complex human genome architecture. *Genome Res*, 14(11):2209–2220, Nov 2004.

[15] Rizzon C, Ponger L, and Gaut BS. Striking Similarities in the Genomic Distribution of Tandemly Arrayed Genes in Arabidopsis and Rice. *PLoS Comput Biol*, 2(9), Sep 2006. JOURNAL ARTICLE.

[16] S. B. Cannon, A. Mitra, A. Baumgarten, N. D. Young, and G. May. The roles of segmental and tandem gene duplication in the evolution of large gene families in Arabidopsis thaliana. *BMC Plant Biol*, 4:10, Jun 2004.

[17] J. M. Hancock. Gene factories, microfunctionalization and the evolution of gene families. *Trends Genet*, 21(11):591–595, Nov 2005.

[18] J. M. Raser and E. K. O'Shea. Noise in gene expression: origins, consequences, and control. *Science*, 309(5743):2010–2013, Sep 2005.

[19] A. M. Tomas and J. M. Kelly. Stage-regulated expression of cruzipain, the major cysteine protease of Trypanosoma cruzi is independent of the level of RNA1. *Mol Biochem Parasitol*, 76(1-2):91–103, Feb 1996.

[20] M. Parniske, K. E. Hammond-Kosack, C. Golstein, C. M. Thomas, D. A. Jones, K. Harrison, B. B. Wulff, and J. D. Jones. Novel disease resistance specificities result from sequence exchange between tandemly repeated genes at the Cf-4/9 locus of tomato. *Cell*, 91(6):821–832, Dec 1997.

[21] B. C. Meyers, S. Kaushik, and R. S. Nandety. Evolving disease resistance genes. *Curr Opin Plant Biol*, 8(2):129–134, Apr 2005.

[22] P. M. R. Aldred, E. J. Hollox, and J. A. L. Armour. Copy number polymorphism and expression level variation of the human alpha-defensin genes DEFA1 and DEFA3. *Hum Mol Genet*, 14(14):2045–2052, Jul 2005.

[23] S. Taudien, P. Galgoczy, K. Huse, K. Reichwald, M. Schilhabel, K. Szafranski, A. Shimizu, S. Asakawa, A. Frankish, I. F. Loncarevic, N. Shimizu, R. Siddiqui, and M. Platzer. Polymorphic segmental duplications at 8p23.1 challenge the determination of individual defensin gene repertoires and the assembly of a contiguous human reference sequence. *BMC Genomics*, 5(1):92, 2004.

[24] G. Santoyo and D. Romero. Gene conversion and concerted evolution in bacterial genomes. *FEMS Microbiol Rev*, 29(2):169–183, Apr 2005.

[25] N. M. El-Sayed, P. J. Myler, D. C. Bartholomeu, D. Nilsson, G. Aggarwal, A. N. Tran, E. Ghedin, E. A. Worthey, A. L. Delcher, G. Blandin, S. J. Westenberger, E. Caler, G. C. Cerqueira, C. Branche, B. Haas, A. Anupama, E. Arner, L. Aslund, P. Attipoe, E. Bontempi, F. Bringaud, P. Burton, E. Cadag, D. A. Campbell, M. Carrington, J. Crabtree, H. Darban, J. F. da Silveira, P. de Jong, K. Edwards, P. T. Englund, G. Fazelina, T. Feldblyum, M. Ferella, A. C. Frasch, K. Gull, D. Horn, L. Hou, Y. Huang, E. Kindlund, M. Klingbeil, S. Kluge, H. Koo, D. Lacerda, M. J. Levin, H. Lorenzi, T. Louie, C. R. Machado, R. McCulloch, A. McKenna, Y. Mizuno, J. C. Mottram, S. Nelson, S. Ochaya, K. Osoegawa, G. Pai, M. Parsons, M. Pentony, U. Pettersson, M. Pop, J. L. Ramirez, J. Rinta, L. Robertson, S. L. Salzberg, D. O. Sanchez, A. Seyler, R. Sharma, J. Shetty, A. J. Simpson, E. Sisk, M. T. Tammi, R. Tarleton, S. Teixeira, S. Van Aken, C. Vogt, P. N. Ward, B. Wickstead, J. Wortman, O. White, C. M. Fraser, K. D. Stuart, and B. Andersson. The genome sequence of Trypanosoma cruzi, etiologic agent of Chagas disease. *Science*, 309(5733):409–415, Jul 2005.

[26] S. Callejas, V. Leech, C. Reitter, and S. Melville. Hemizygous subtelomeres of an African trypanosome chromosome may account for over 75 *Genome Res*, 16(9):1109–1118, Sep 2006.

[27] P. Stankiewicz and J. R. Lupski. Genome architecture, rearrangements and genomic disorders. *Trends Genet*, 18(2):74–82, Feb 2002.

[28] Y. Ji, E. E. Eichler, S. Schwartz, and R. D. Nicholls. Structure of chromosomal duplicons and their role in mediating human genomic disorders. *Genome Res*, 10(5):597–610, May 2000.

[29] J. W. Gray, S. Suzuki, W. L. Kuo, D. Polikoff, M. Deavers, K. Smith-McCune, A. Berchuck, D. Pinkel, D. Albertson, and G. B. Mills. Specific keynote: genome copy number abnormalities in ovarian cancer. *Gynecol Oncol*, 88(1 Pt 2):16–21, Jan 2003.

[30] O. Kabbarah and L. Chin. Revealing the genomic heterogeneity of melanoma. *Cancer Cell*, 8(6):439–441, Dec 2005.

[31] S. M. Mirkin. DNA structures, repeat expansions and human hereditary disorders. *Curr Opin Struct Biol*, 16(3):351–358, Jun 2006.

[32] F. Sanger, S. Nicklen, and A. R. Coulson. DNA sequencing with chain-terminating inhibitors. *Proc Natl Acad Sci U S A*, 74(12):5463–5467, Dec 1977.

[33] F. Sanger, A. R. Coulson, B. G. Barrell, A. J. Smith, and B. A. Roe. Cloning in single-stranded bacteriophage as an aid to rapid DNA sequencing. *J Mol Biol*, 143(2):161–178, Oct 1980.

[34] R. W. Holley, J. Apgar, G. A. Everett, J. T. Madison, M. Marquisee, S. H. Merrill, J. R. Penswick, and A. Zamir. STRUCTURE OF A RIBONUCLEIC ACID. *Science*, 147:1462–1465, Mar 1965.

[35] H. Peltola, H. Soderlund, and E. Ukkonen. SEQAID: a DNA sequence assembling program based on a mathematical model. *Nucleic Acids Res*, 12(1 Pt 1):307–321, Jan 1984.

[36] J. K. Bonfield, K. F. Smith, and R. Staden. A new DNA sequence assembly program. *Nucleic Acids Res*, 23(24):4992–4999, Dec 1995.

[37] X. Huang. A contig assembly program based on sensitive detection of fragment overlaps. *Genomics*, 14(1):18–25, Sep 1992.

[38] X. Huang. An improved sequence assembly program. *Genomics*, 33(1):21–31, Apr 1996.

[39] X. Huang and A. Madan. CAP3: A DNA sequence assembly program. *Genome Res*, 9(9):868–877, Sep 1999.

[40] X. Huang, J. Wang, S. Aluru, S. P. Yang, and L. Hillier. PCAP: a whole-genome assembly program. *Genome Res*, 13(9):2164–2170, Sep 2003.

[41] S. Batzoglou, D. B. Jaffe, K. Stanley, J. Butler, S. Gnerre, E. Mauceli, B. Berger, J. P. Mesirov, and E. S. Lander. ARACHNE: a whole-genome shotgun assembler. *Genome Res*, 12(1):177–189, Jan 2002.

[42] D. B. Jaffe, J. Butler, S. Gnerre, E. Mauceli, K. Lindblad-Toh, J. P. Mesirov, M. C. Zody, and E. S. Lander. Whole-genome sequence assembly for mammalian genomes: Arachne 2. *Genome Res*, 13(1):91–96, Jan 2003.

[43] E. W. Myers, G. G. Sutton, A. L. Delcher, I. M. Dew, D. P. Fasulo, M. J. Flanigan, S. A. Kravitz, C. M. Mobarry, K. H. Reinert, K. A. Remington, E. L. Anson, R. A. Bolanos, H. H. Chou, C. M. Jordan, A. L. Halpern, S. Lonardi, E. M. Beasley, R. C. Brandon, L. Chen, P. J. Dunn, Z. Lai, Y. Liang, D. R. Nusskern, M. Zhan, Q. Zhang, X. Zheng, G. M. Rubin, M. D. Adams, and J. C. Venter. A whole-genome assembly of Drosophila. *Science*, 287(5461):2196–2204, Mar 2000.

[44] J. C. Mullikin and Z. Ning. The phusion assembler. *Genome Res*, 13(1):81–90, Jan 2003.

[45] J. Wang, G. K. Wong, P. Ni, Y. Han, X. Huang, J. Zhang, C. Ye, Y. Zhang, J. Hu, K. Zhang, X. Xu, L. Cong, H. Lu, X. Ren, X. Ren, J. He, L. Tao, D. A. Passey, J. Wang, H. Yang, J. Yu, and S. Li. RePS: a sequence assembler that masks exact repeats identified from the shotgun data. *Genome Res*, 12(5):824–831, May 2002.

[46] P. A. Pevzner, H. Tang, and M. S. Waterman. An Eulerian path approach to DNA fragment assembly. *Proc Natl Acad Sci U S A*, 98(17):9748–9753, Aug 2001.

[47] B. Ewing, L. Hillier, M. C. Wendl, and P. Green. Base-calling of automated sequencer traces using phred. I. Accuracy assessment. *Genome Res*, 8(3):175–185, Mar 1998.

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

[49] D. Walther, G. Bartha, and M. Morris. Basecalling with LifeTrace. *Genome Res*, 11(5):875–888, May 2001.

[50] K. M. Chao, W. R. Pearson, and W. Miller. Aligning two sequences within a specified diagonal band. *Comput Appl Biosci*, 8(5):481–487, Oct 1992.

[51] T. F. Smith and M. S. Waterman. Identification of common molecular subsequences. *J Mol Biol*, 147(1):195–197, Mar 1981.

[52] S. F. Altschul, W. Gish, W. Miller, E. W. Myers, and D. J. Lipman. Basic local alignment search tool. *J Mol Biol*, 215(3):403–410, Oct 1990.

[53] J. K. Gallant. The complexity of the overlap method for sequencing biopolymers. *J Theor Biol*, 101(1):1–17, Mar 1983.

[54] E. L. Anson and E. W. Myers. ReAligner: a program for refining DNA sequence multi-alignments. *J Comput Biol*, 4(3):369–383, Fall 1997.

[55] E. S. Lander and M. S. Waterman. Genomic mapping by fingerprinting random clones: a mathematical analysis. *Genomics*, 2(3):231–239, Apr 1988.

[56] A. Edwards, H. Voss, P. Rice, A. Civitello, J. Stegemann, C. Schwager, J. Zimmermann, H. Erfle, C. T. Caskey, and W. Ansorge. Automated DNA sequencing of the human HPRT locus. *Genomics*, 6(4):593–608, Apr 1990.

[57] M. Pop, D. S. Kosack, and S. L. Salzberg. Hierarchical scaffolding with Bambus. *Genome Res*, 14(1):149–159, Jan 2004.

[58] W. J. Kent and D. Haussler. Assembly of the working draft of the human genome with GigAssembler. *Genome Res*, 11(9):1541–1548, Sep 2001.

[59] L. D. Stein. Human genome: end of the beginning. *Nature*, 431(7011):915–916, Oct 2004. Comment.

[60] D. Gordon, C. Desmarais, and P. Green. Automated finishing with autofinish. *Genome Res*, 11(4):614–625, Apr 2001.

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

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

[63] L. Frangeul, P. Glaser, C. Rusniok, C. Buchrieser, E. Duchaud, P. Dehoux, and F. Kunst. CAAT-Box, Contigs-Assembly and Annotation Tool-Box for genome sequencing projects. *Bioinformatics*, 20(5):790–797, Mar 2004.

[64] L. Herron-Olson, J. Freeman, Q. Zhang, E. F. Retzel, and V. Kapur. MGView: an alignment and visualization tool to enhance gap closure of microbial genomes. *Nucleic Acids Res*, 31(17):e106, Sep 2003.

[65] J. C. Venter, M. D. Adams, G. G. Sutton, A. R. Kerlavage, H. O. Smith, and M. Hunkapiller. Shotgun sequencing of the human genome. *Science*, 280(5369):1540–1542, Jun 1998.

[66] J. L. Weber and E. W. Myers. Human whole-genome shotgun sequencing. *Genome Res*, 7(5):401–409, May 1997.

[67] J. C. Venter, H. O. Smith, and L. Hood. A new strategy for genome sequencing. *Nature*, 381(6581):364–366, May 1996.

[68] P. Green. Against a whole-genome shotgun. *Genome Res*, 7(5):410–417, May 1997.

[69] E. E. Eichler. Masquerading repeats: paralogous pitfalls of the human genome. *Genome Res*, 8(8):758–762, Aug 1998.

[70] R. H. Waterston, E. S. Lander, and J. E. Sulston. On the sequencing of the human genome. *Proc Natl Acad Sci U S A*, 99(6):3712–3716, Mar 2002.

[71] E. W. Myers, G. G. Sutton, H. O. Smith, M. D. Adams, and J. C. Venter. On the sequencing and assembly of the human genome. *Proc Natl Acad Sci U S A*, 99(7):4145–4146, Apr 2002. Comment.

[72] P. Green. Whole-genome disassembly. *Proc Natl Acad Sci U S A*, 99(7):4143–4144, Apr 2002.

[73] R. H. Waterston, E. S. Lander, and J. E. Sulston. More on the sequencing of the human genome. *Proc Natl Acad Sci U S A*, 100(6):3022–3024, Mar 2003. Comment.

[74] M. D. Adams, G. G. Sutton, H. O. Smith, E. W. Myers, and J. C. Venter. The independence of our genome assemblies. *Proc Natl Acad Sci U S A*, 100(6):3025–3026, Mar 2003. JOURNAL ARTICLE.

[75] J. Cheung, X. Estivill, R. Khaja, J. R. MacDonald, K. Lau, L. C. Tsui, and S. W. Scherer. Genome-wide detection of segmental duplications and potential assembly errors in the human genome sequence. *Genome Biol*, 4(4):R25, 2003.

[76] S. Istrail, G. G. Sutton, L. Florea, A. L. Halpern, C. M. Mobarry, R. Lippert, B. Walenz, H. Shatkay, I. Dew, J. R. Miller, M. J. Flanigan, N. J. Edwards, R. Bolanos, D. Fasulo, B. V. Halldorsson, S. Hannenhalli, R. Turner, S. Yooseph, F. Lu, D. R. Nusskern, B. C. Shue, X. H. Zheng, F. Zhong, A. L. Delcher, D. H. Huson, S. A. Kravitz, L. Mouchard, K. Reinert, K. A. Remington, A. G. Clark, M. S. Waterman, E. E. Eichler, M. D. Adams, M. W. Hunkapiller, E. W. Myers, and J. C. Venter. Whole-genome shotgun assembly and comparison of human genome assemblies. *Proc Natl Acad Sci U S A*, 101(7):1916–1921, Feb 2004. Evaluation Studies.

[77] X. She, Z. Jiang, R. A. Clark, G. Liu, Z. Cheng, E. Tuzun, D. M. Church, G. Sutton, A. L. Halpern, and E. E. Eichler. Shotgun sequence assembly and recent segmental duplications within the human genome. *Nature*, 431(7011):927–930, Oct 2004.

[78] S. Li, G. Cutler, J. J. Liu, T. Hoey, L. Chen, P. G. Schultz, J. Liao, and X. B. Ling. A comparative analysis of HGSC and Celera human genome assemblies and gene sets. *Bioinformatics*, 19(13):1597–1605, Sep 2003. Evaluation Studies.

[79] D. H. Huson, K. Reinert, S. A. Kravitz, K. A. Remington, A. L. Delcher, I. M. Dew, M. Flanigan, A. L. Halpern, Z. Lai, C. M. Mobarry, G. G. Sutton, and E. W. Myers. Design of a compartmentalized shotgun assembler for the human genome. *Bioinformatics*, 17 Suppl 1:132–139, 2001.

[80] P. Havlak, R. Chen, K. J. Durbin, A. Egan, Y. Ren, X. Z. Song, G. M. Weinstock, and R. A. Gibbs. The Atlas genome assembly system. *Genome Res*, 14(4):721–732, Apr 2004.

[81] R. L. Warren, D. Varabei, D. Platt, X. Huang, D. Messina, S. P. Yang, J. W. Kronstad, M. Krzywinski, W. C. Warren, J. W. Wallis, L. W. Hillier, A. T. Chinwalla, Jacqueline E. Schein, A. S. Siddiqui, M. A. Marra, R. K. Wilson, and S. J. M. Jones. Physical map-assisted whole-genome shotgun sequence assemblies. *Genome Res*, 16(6):768–775, Jun 2006.

[82] P. A. Pevzner and H. Tang. Fragment assembly with double-barreled data. *Bioinformatics*, 17 Suppl 1:225–233, 2001.

[83] P. A. Pevzner, H. Tang, and G. Tesler. De novo repeat classification and fragment assembly. *Genome Res*, 14(9):1786–1796, Sep 2004.

[84] I. M. Dew, B. Walenz, and G. Sutton. A tool for analyzing mate pairs in assemblies (TAMPA). *J Comput Biol*, 12(5):497–513, Jun 2005.

[85] D. Bartels, S. Kespohl, S. Albaum, T. Druke, A. Goesmann, J. Herold, O. Kaiser, A. Puhler, F. Pfeiffer, G. Raddatz, J. Stoye, F. Meyer, and S. C. Schuster. BACCardI–a tool for the validation of genomic assemblies, assisting genome finishing and intergenome comparison. *Bioinformatics*, 21(7):853–859, Apr 2005. Evaluation Studies.

[86] S. J. Lindsay, J. K. Bonfield, and M. E. Hurles. Shotgun haplotyping: a novel method for surveying allelic sequence variation. *Nucleic Acids Res*, 33(18):e152, 2005.

[87] G. T. Marth, I. Korf, M. D. Yandell, R. T. Yeh, Z. Gu, H. Zakeri, N. O. Stitziel, L. Hillier, P. Y. Kwok, and W. R. Gish. A general approach to single-nucleotide polymorphism discovery. *Nat Genet*, 23(4):452–456, Dec 1999.

[88] J. Wang and X. Huang. A method for finding single-nucleotide polymorphisms with allele frequencies in sequences of deep coverage. *BMC Bioinformatics*, 6:220, 2005.

[89] J. Kececioglu and J. Yu. Separating repeats in DNA sequence assembly. *Proceedings of the Fifth Annual International Conference on Computational Biology (RECOMB)*, 99(7):176–183, Apr 2001.

[90] M. Roberts, B. R. Hunt, J. A. Yorke, R. A. Bolanos, and A. L. Delcher. A preprocessor for shotgun assembly of large genomes. *J Comput Biol*, 11(4):734–752, 2004.

[91] P. Gajer, M. Schatz, and S. L. Salzberg. Automated correction of genome sequence errors. *Nucleic Acids Res*, 32(2):562–569, 2004.

[92] B. Chevreux, T. Pfisterer, B. Drescher, A. J. Driesel, W. E. G. Muller, T. Wetter, and S. Suhai. Using the miraEST assembler for reliable and automated mRNA transcript assembly and SNP detection in sequenced ESTs. *Genome Res*, 14(6):1147–1159, Jun 2004.

[93] S. J. Westenberger, N. R. Sturm, and D. A. Campbell. Trypanosoma cruzi 5S rRNA arrays define five groups and indicate the geographic origins of an ancestor of the heterozygous hybrids. *Int J Parasitol*, 36(3):337–346, Mar 2006.

[94] S. Thomas, S. J. Westenberger, D. A. Campbell, and N. R. Sturm. Intragenomic spliced leader RNA array analysis of kinetoplastids reveals unexpected transcribed region diversity in Trypanosoma cruzi. *Gene*, 352:100–108, Jun 2005.

[95] P. C. Ng and S. Henikoff. SIFT: Predicting amino acid changes that affect protein function. *Nucleic Acids Res*, 31(13):3812–3814, Jul 2003.

[96] C. Hertz-Fowler, C. S. Peacock, V. Wood, M. Aslett, A. Kerhornou, P. Mooney, A. Tivey, M. Berriman, N. Hall, K. Rutherford, J. Parkhill, A. C. Ivens, M. A. Rajandream, and B. Barrell. GeneDB: a resource for prokaryotic and eukaryotic organisms. *Nucleic Acids Res*, 32(Database issue):339–343, Jan 2004.

[97] M. L. Cremona, D. O. Sanchez, A. C. Frasch, and O. Campetella. A single tyrosine differentiates active and inactive Trypanosoma cruzi transsialidases. *Gene*, 160(1):123–128, Jul 1995.

[98] C. A. Buscaglia, V. A. Campo, A. C. C. Frasch, and J. M. Di Noia. Trypanosoma cruzi surface mucins: host-dependent coat diversity. *Nat Rev Microbiol*, 4(3):229–236, Mar 2006.

[99] L. Kall, A. Krogh, and E. L. L. Sonnhammer. A combined transmembrane topology and signal peptide prediction method. *J Mol Biol*, 338(5):1027–1036, May 2004.

[100] S. L. Salzberg and J. A. Yorke. Beware of mis-assembled genomes. *Bioinformatics*, 21(24):4320–4321, Dec 2005. Letter.

[101] E. E. Eichler. Segmental duplications: what's missing, misassigned, and misassembled–and should we care? *Genome Res*, 11(5):653–656, May 2001.

[102] K. Chen and L. Pachter. Bioinformatics for whole-genome shotgun sequencing of microbial communities. *PLoS Comput Biol*, 1(2):106–112, Jul 2005.

[103] J. P. Vinson, D. B. Jaffe, K. O'Neill, E. K. Karlsson, N. Stange-Thomann, S. Anderson, J. P. Mesirov, N. Satoh, Y. Satou, C. Nusbaum, B. Birren, J. E. Galagan, and E. S. Lander. Assembly of polymorphic genomes: algorithms and application to Ciona savignyi. *Genome Res*, 15(8):1127–1135, Aug 2005.

[104] M. Pop, A. Phillippy, A. L. Delcher, and S. L. Salzberg. Comparative genome assembly. *Brief Bioinform*, 5(3):237–248, Sep 2004.

[105] M. Margulies, M. Egholm, W. E. Altman, S. Attiya, J. S. Bader, L. A. Bemben, J. Berka, M. S. Braverman, Y. J. Chen, Z. Chen, S. B. Dewell, L. Du, J. M. Fierro, X. V. Gomes, B. C. Godwin, W. He, S. Helgesen, C. H. Ho, G. P. Irzyk, S. C. Jando, M. L. I. Alenquer, T. P. Jarvie, K. B. Jirage, J. B. Kim, J. R. Knight, J. R. Lanza, J. H. Leamon, S. M. Lefkowitz, M. Lei, J. Li, K. L. Lohman, H. Lu, V. B. Makhijani, K. E. McDade, M. P. McKenna, E. W. Myers, E. Nickerson, J. R. Nobile, R. Plant, B. P. Puc, M. T. Ronan, G. T. Roth, G. J. Sarkis, J. F. Simons, J. W. Simpson, M. Srinivasan, K. R. Tartaro, A. Tomasz, K. A. Vogt, G. A. Volkmer, S. H. Wang, Y. Wang, M. P. Weiner, P. Yu, R. F. Begley, and J. M. Rothberg. Genome sequencing in microfabricated high-density picolitre reactors. *Nature*, 437(7057):376–380, Sep 2005.

[106] S. T. Bennett, C. Barnes, A. Cox, L. Davies, and C. Brown. Toward the 1,000 dollars human genome. *Pharmacogenomics*, 6(4):373–382, Jun 2005.

[107] M. Chaisson, P. Pevzner, and H. Tang. Fragment assembly with short reads. *Bioinformatics*, 20(13):2067–2074, Sep 2004. Evaluation Studies.

[108] S. M. D. Goldberg, J. Johnson, D. Busam, T. Feldblyum, S. Ferriera, R. Friedman, A. Halpern, H. Khouri, S. A. Kravitz, F. M. Lauro, K. Li, Y. H. Rogers, R. Strausberg, G. Sutton, L. Tallon, T. Thomas, E. Venter, M. Frazier, and J. C. Venter. A Sanger/pyrosequencing hybrid approach for the generation of high-quality draft assemblies of marine microbial genomes. *Proc Natl Acad Sci U S A*, 103(30):11240–11245, Jul 2006.

[109] Y. Ponty, M. Termier, and A. Denise. GenRGenS: software for generating random genomic sequences and structures. *Bioinformatics*, 22(12):1534–1535, Jun 2006.

[110] M. L. Engle and C. Burks. GenFrag 2.1: new features for more robust fragment assembly benchmarks. *Comput Appl Biosci*, 10(5):567–568, Sep 1994.

[111] H. Shatkay, J. Miller, C. Mobarry, M. Flanigan, S. Yooseph, and G. Sutton. ThurGood: evaluating assembly-to-assembly mapping. *J Comput Biol*, 11(5):800–811, 2004.

[112] S. L. Salzberg, D. Church, M. DiCuccio, E. Yaschenko, and J. Ostell. The genome Assembly Archive: a new public resource. *PLoS Biol*, 2(9):E285, Sep 2004.

[113] E. W. Myers. The fragment assembly string graph. *Bioinformatics*, 21 Suppl 2:ii79–ii85, Sep 2005.

Part II

Reports