Otto-von-Guericke-Universität Magdeburg

Fakultät für Informatik

Studienarbeit

Mapping of WDLPS neochromosomes

Kristian Löwe

September 15, 2010

Academic Supervisors:

Dr. Arthur Hsu
Walter and Eliza Hall Institute of Medical Research
Parkville Victoria 3052, Australia

Prof. Dr. Saman Halgamuge
University of Melbourne
Department of Mechanical Engineering
Parkville Victoria 3010, Australia

Prof. Dr. Rudolf Kruse, Dipl.-Inf. Georg Ruß
Universität Magdeburg
Fakultät für Informatik
Postfach 4120, 39016 Magdeburg, Germany

Löwe, Kristian:
Mapping of WDLPS neochromosomes
Studienarbeit,
Otto-von-Guericke-Universität Magdeburg,
2010

Acknowledgements

I would like to thank Georg Ruß for his support and advice and for involving me in this project in the first place.
Special thanks go to Dr. Arthur Hsu, who guided me through this project and spent a lot of his time enabling me to develop an understanding of the subject and discussing ideas.
I'd like to say thank you to Prof. Saman Halgamuge and Prof. Rudolf Kruse, who afforded me the opportunity for this internship and provided advise and support wherever necessary.
I'd like to thank all the members of the research group, who made me feel welcome from day one.
Special thanks go to my father Peter Löwe for his help in improving the manuscript and to both my parents and grandparents for supporting my stay in Australia.
Thanks also to my friend Sascha Peilicke for proofreading this report.

Contents

1 Introduction **1**

2 Basics **3**

2.1 DNA 3

2.2 Sequencing 4

2.2.1 Shotgun sequencing 5

2.2.2 Sequencing technologies 5

2.3 Assembly 7

2.3.1 A greedy approach for the shortest common superstring problem 11

2.3.2 Graph theory approaches 12

2.3.2.1 Overlap-layout-consensus 12

2.3.2.2 De Bruijn graph approach 13

2.3.3 Scaffolding 15

3 Mapping of WDLPS neochromosomes **17**

3.1 Cytogenetic features of WDLPS 17

3.2 Data 19

3.3 Analysis strategy 19

3.3.1 Read-contig mapping 21

3.3.2 Contig-reference mapping 25

3.3.3 Test data generation 27

3.3.4 Implementation 28

4 Results 29

5 Concluding remarks 37

List of Figures 41

List of Tables 43

List of Abbreviations 45

Bibliography 47

1 Introduction

As first established by Boveri in his pioneering work on malignant tumours [Boveri, 1914], chromosomal rearrangements within the genome are substantially involved in carcinogenesis. The resulting changes in gene expression cause the abnormal and malignant behaviour of the affected cells.

Well-differentiated liposarcoma (WDLPS) is a type of cancer occuring in fat cells of soft-tissue. While most cancer genomes show alterations of the normal chromosomes, WDLPS is typically characterized by one or more neochromosomes that occur in addition to the otherwise mostly unaltered karyotype [Sandberg, 2004]. The malignancy of the cells is therefore believed to arise from the neochromosomes which are composed of various fragments derived from the normal chromosomes [Garsed et al., 2009].

While the composition of individual tumors varies, commonly occuring components of WDLPS neochromosomes - such as 12q14–15 - have been identified [Garsed et al., 2009]. Less is known, however, about the spatial order of the incorporated fragments. Their breakpoints are of particular interest as they are likely to comprise gene alterations such as gene fusions or truncations, that are widely believed to play a central role in the progression of WDLPS.

Next-generation sequencing (NGS) platforms provide a cost-effective and high-resolution tool when aiming to characterize structural rearrangements within cancer genomes as showcased e.g. by Campbell et al. [2008]. Provided with short-read sequencing data of the neochromosome of a WDLPS patient, it is the aim of this work to identify the incorporated donor regions and especially the sites of fusions between them.

The further characterization of such breakpoints will provide valuable information, not only aiding diagnostics and therapeutics, but also helping to understand the underlying mechanisms of cancer development. However, the biological analysis necessary to make sense of the identified breakpoints is beyond the scope of this thesis.

In order to tackle the tasks outlined above, some theoretical knowledge had to be acquired as to the concepts, techniques, and methodologies applied in or applicable to genetic research. Since a notable fraction of the time allotted to this study had to be spent on such preparatory work, it is deemed appropriate to elaborate on some such 'tools of the trade' here. Beyond that, this compilation of basics (provided in chapter 2) is intended to make the body of this study more easily accessible to a readership from outside the genetic field. In particular, current DNA sequencing technologies employing the shotgun sequencing strategy and different assembly approaches are discussed.

In chapter 3, the material and methods used in order to identify donor regions and breakpoints on the sequenced neochromosome are presented.

Starting out from the cytogenetic characteristics of WDLPS and a concise description of the neochromosomal dataset the general analysis strategy is layed out. Then, a more extensive description of self-developed modules of the full chain of analysis is given. The implementation of the analysis pipeline and its general feasibility were validated using synthetic datasets modelled after realistic data properties.

Next, some results on the liposarcoma neochromosome are presented, before this thesis is concluded in its final chapter.

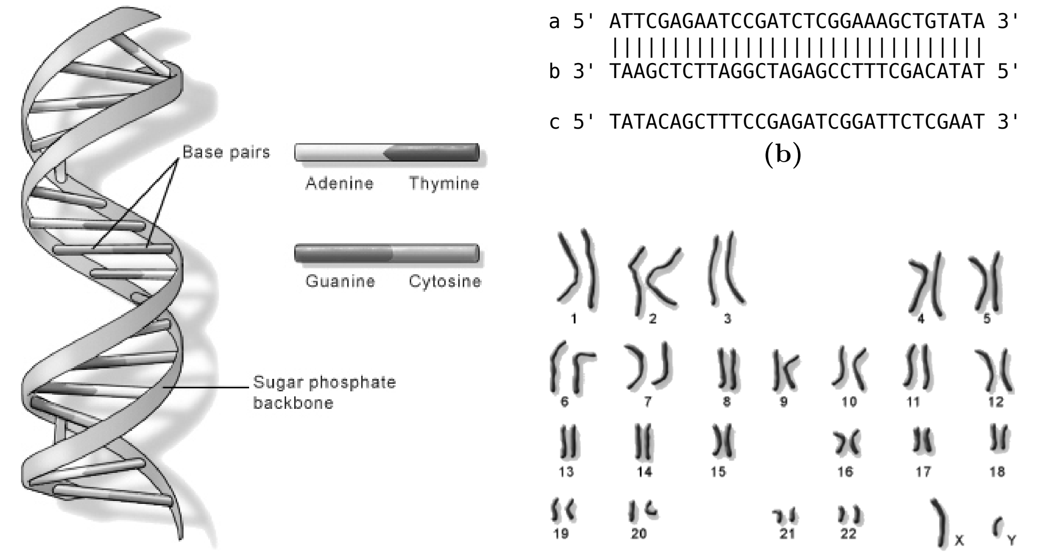
2 Basics

2.1 DNA

Deoxyribonucleic acid (DNA) contains the genetic information in almost all living organisms, including humans. Chemically, DNA consists of two antiparallel strands of nucleotides forming a spiral structure (Fig. 2.1a). The two ends of each strand are distinguished by the chemical groups that terminate them. One end, the 5' end, has a terminal phosphate group while the other one, the 3' end, has a terminal hydroxyl group (Fig. 2.1b).

Each nucleotide is made up of a sugar, a phospate and one of the four bases adenine (A), guanine (G), cytosine (C), and thymine (T). Sugar and phosphate build the backbone of each strand while each base on the one strand forms a hydrogen bond with one base on the other strand. These units formed by two bases are called base pairs (bp), with A bonding only to T and G bonding only to C. As a result, the two nucleotide strands are complementary and the entire genetic information – which is incorporated in the sequence of the bases along the sugar phosphate backbone of each strand – is present twice. This property is utilized in the process of DNA replication in the context of cell division.

Within cells, DNA is organized in chromosomes. Humans have 22 different autosomes occurring in pairs and two sex chromosomes forming another pair, which adds up to a total of 46 chromosomes (Fig. 2.1c). Together, the 46 chromosomes form the human genome.



```
a 5' ATTCGAGAATCCGATCTCGGAAAGCTGTATA 3'
     |||||||||||||||||||||||||||||||
b 3' TAAGCTCTTAGGCTAGAGCCTTTCGACATAT 5'

c 5' TATACAGCTTTCCGAGATCGGATTCTCGAAT 3'
```

(b)

(a) **(c)**

Figure 2.1: DNA structure and organisation. A: A stretch of DNA. B: The figure exemplifies the representation of DNA sequences using the IUPAC Nucleotide Code. a and b form a piece of double-stranded DNA. b is the complement of a. As DNA sequences are generally read in direction from 5' to 3', b can also be represented by its reverse c which is the reverse complement of a. C: DNA is organized in chromosomes. The figure shows the set of chromosomes of a male human. The 22 different autosomes come in pairs. Additionally, there are two sex chromosomes (X and Y or two X) forming another pair. A and C adopted from the Genetics Home Reference on the National Library of Medicine's web site (`http://www.nlm.nih.gov/`).

2.2 Sequencing

In the context of genome analysis sequencing refers to the process of determining the exact sequence of nucleotides in a sample of DNA. However, the length of a DNA molecule that can be sequenced directly using current sequencing technology is quite limited. The maximum length is around 2000bp, while chromosomes – e.g. the human choromosome 1 at 247 million nucleotide base pairs – are much longer. Still, the determination of very long stretches of DNA is possible by use of the shotgun strategy. In this, the original sequence is split into smaller, directly sequencable fragments, the information of which is to be pieced together through assembly.

In the following, we will first describe the shotgun sequencing strategy (section 2.2.1) and proceed to a comparison of current sequencing technologies that utilize it (section 2.2.2). Different approaches to assembly are discussed separately in section 2.3.

2.2.1 Shotgun sequencing

Again, by use of current technologies one can directly sequence DNA only up to a certain length. Invariably, this length is much smaller than complete genomes or chromosomes. A strategy to resolve this problem is shotgun sequencing [Staden, 1979], which consists of four steps: First, the original DNA sample (Fig. 2.2a) is copied many times (Fig. 2.2b). These copies are then randomly sheared into smaller fragments, e.g. by ultrasound (Fig. 2.2c). Subsequently, a subset of these fragments is selected by size[1] (Fig. 2.2d). Finally, from each fragment a read[2] is obtained.

In the double-barrel variant of shotgun sequencing a read is obtained from both ends of each fragment, which leads to a collection of read pairs instead of just single reads. The pairs provide valuable information, because not only the sequence of the two reads is known, but – due to the size-selection of the fragments – the approximate distance between them as well (Fig. 2.2e). The collection of reads resulting from a shotgun sequencing experiment is subject to the assembly process (section 2.3) aiming to reconstruct as much as possible of the original source sequence.

2.2.2 Sequencing technologies

When comparing current sequencing technologies there are many factors to consider such as read length, accuracy, speed and cost effectiveness – both of the latter being highly related.

Since its introduction in 1977 Sanger sequencing [Sanger et al., 1977] has

[1] The term shotgun sequencing derives from the analogy to the firing of a shotgun whereupon the pellets spread randomly towards the target. Similarly, the selected fragments are obtained as if from random locations on the source sequence.

[2] The nucleotide order is determined beginning at the end of a fragment. The obtained sequence is called a read. A read is shorter than the fragment, i.e. only the end fraction of the fragment is sequenced.

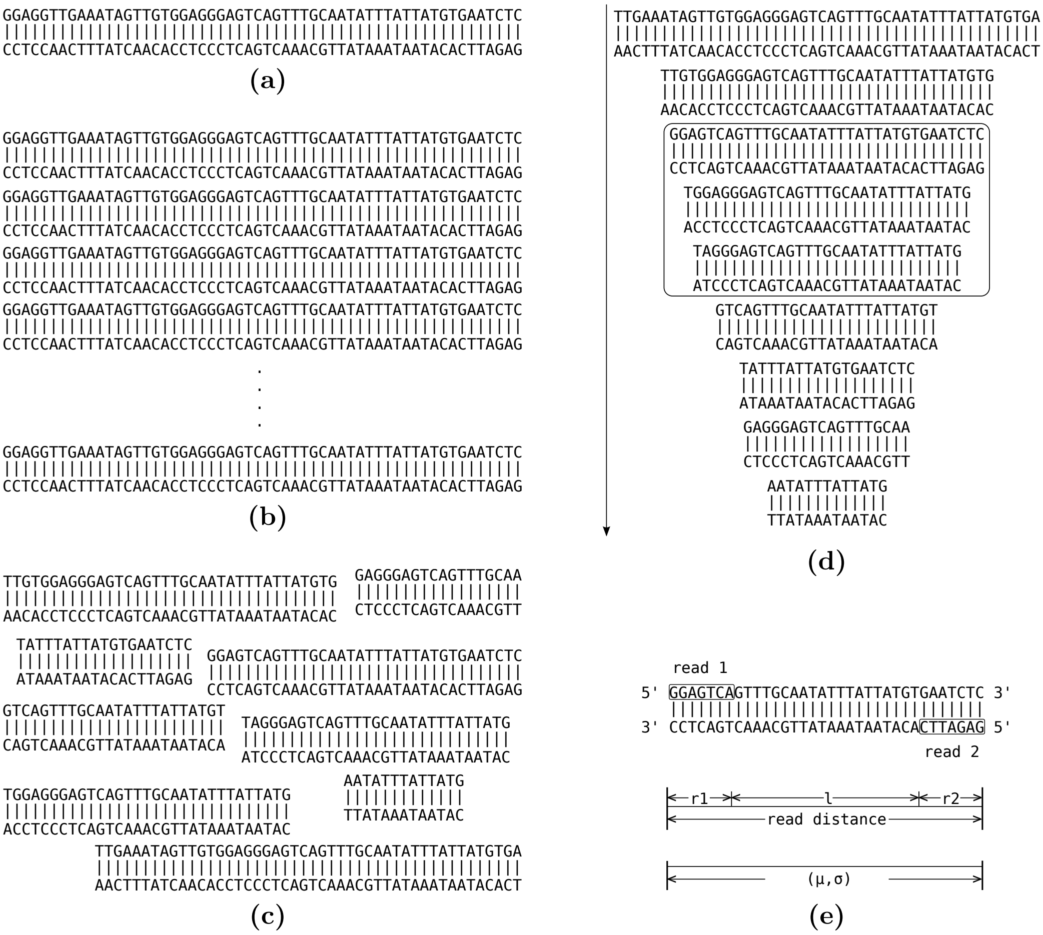

GGAGGTTGAAATAGTTGTGGAGGGAGTCAGTTTGCAATATTTATTATGTGAATCTC
CCTCCAACTTTATCAACACCTCCCTCAGTCAAACGTTATAAATAATACACTTAGAG
(a)
(b)
(c)
TTGAAATAGTTGTGGAGGGAGTCAGTTTGCAATATTTATTATGTGA
AACTTTATCAACACCTCCCTCAGTCAAACGTTATAAATAATACACT
(d)
read 1
5' GGAGTCAGTTTGCAATATTTATTATGTGAATCTC 3'
3' CCTCAGTCAAACGTTATAAATAATACACTTAGAG 5'
read 2
r1
l
r2
read distance
(μ,σ)
(e)

Figure 2.2: Shotgun sequencing. For details see text.

dominated the field, due to its efficiency and reliability compared to other methods. Gradually improved over the years, it currently affords read-lengths of up to ~2000bp and per-base accuracies as high as 99.999%. With automation and parallelization of the process the cost effectiveness went up significantly. The price per kilo base (kb) dropped from $10000 in 1985 to $1 in 2005, when the potential of increasing the throughput seemed mostly exhausted, making a further major decrease in cost unlikely. Thus, intense efforts towards the development of alternative methods were made.

The first of these next-generation sequencing (NGS) technologies, the Genome-

Sequencer by 454 Life Sciences[3], was introduced in 2005 [Margulies et al., 2005], shortly followed by the Illumina/Solexa Genome Analyzer[4] in 2006 and the ABI/SOLiD system[5] in 2007. The new platforms afford fast and cheap generation of an abundance of data. This main advantage over traditional Sanger-based methods is due to a far higher throughput resulting from a very high degree of parallelization of the sequencing process. Strongly reduced cost per mega base (Mb) in comparison to Sanger sequencing puts "large-scale sequencing within the reach of many scientists" [Pop and Salzberg, 2008]. On the downside are shorter read-lengths and lower accuracies. While paired-end data is available with both Sanger-based and NGS platforms, the possible read distance is much smaller with NGS [Kingsford et al., 2010]. An overview of the properties of current sequencing technologies is provided in Table 2.1. For more detailed descriptions of these technologies the reader is referred to the reviews by Mardis [2008] or Ansorge [2009].

	method	cost ($/Mb)	read-length (bp)	accuracy (%)
Sanger	chain-termination	1000	up to 2000	99.0 to > 99.999
454	pyrosequencing	60	250 bp	96.0 to 97.0
Illumina	sequencing-by-synthesis	2	35/75 bp	96.2 to 99.7
SOLiD	sequencing-by-ligation	2	35 bp	99.0 to > 99.9

Table 2.1: Comparison of current sequencing technologies. Estimates of costs, read-lengths and accuracies may quickly be outdated due to ongoing research and rapid development in the field. Accuracies from Chan [2009].

2.3 Assembly

As current sequencing methods produce reads of limited length, longer stretches of DNA can only be determined by assembling – i.e. aligning and merging – a collection of reads R previously obtained from shotgun sequencing. The reads, just as the source sequence, are words over the alphabet Σ. The sequence of each read is known, while the corresponding location on the source sequence S is not.

[3] `http://www.454.com/`

[4] `http://www.solexa.com/`

[5] `http://www.appliedbiosystems.com/`

Formally described in (2.1), this situation constitutes the fragment assembly problem (FAP), which is to reconstruct the original sequence S given the collection of reads R.

$$\begin{gathered}\Sigma = \{A, C, G, T\} \\ S = s_1 s_2 ... s_m, s_i \in \Sigma \text{ oder } S \in \Sigma^* \\ R = \{r_1, r_2, ..., r_n\}, r_i \in \Sigma^*\end{gathered} \tag{2.1}$$

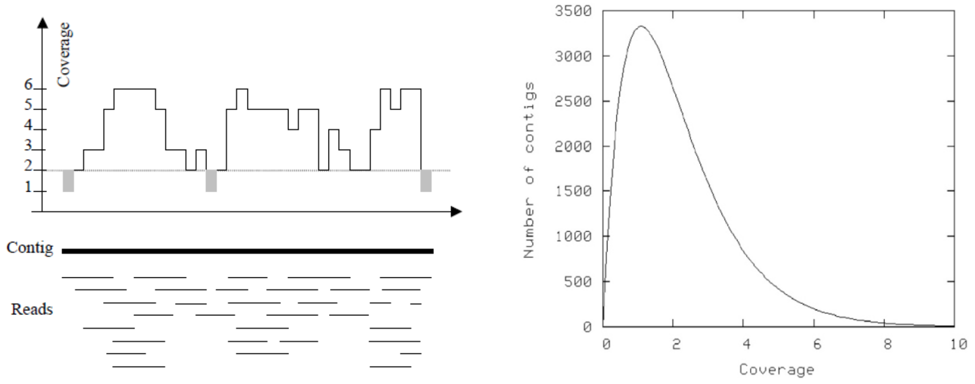
This is a difficult (NP-hard) combinatorial optimization problem as, a priori, there are $2^n n!$ possible combinations to assemble n reads [Alba and Dorronsoro, 2008]. Thus, feasible approaches involve sophisticated algorithms that take into account the characteristics of the data and preferably additional constraints, e.g., paired-end information or a reference assembly. While the former can be used in any case, it is inappropriate to use the latter, if substantial rearrangements are to be expected. For this reason, only de novo assembly, i.e., assembly without the use of a reference, is considered in the remainder of this section.

As a consequence of random shearing during shotgun sequencing the source sequence is not uniformly covered by reads (Fig. 2.3a). The local coverage at a given position of the source sequence is defined as the number of reads that cover it. Accordingly, the overall coverage of the source sequence S is the ratio of the size of the read set and the size of S:

$$coverage(S) = \frac{\sum_{i=1}^{n} |r_i|}{|S|} \tag{2.2}$$

Stretches of S not covered by reads cannot be reconstructed by assembly. If such stretches occur, the assembly is bound to be fragmented, i.e., its result will consist of multiple contigs[6]. Lander and Waterman [1988] analyzed the relationship between coverage and the number of contigs of an assembly (Fig. 2.3b) on the supposition that the reads are distributed along the source sequence according to a Poisson process. Drops of rain cover the ground bit by bit; many spots are hit repeatedly before, eventually, the entire surface

[6] A contig is a contiguous sequence obtained by merging overlapping reads or sequences.



(a) (b)

Figure 2.3: Coverage distribution. A: Coverage distribution of a contig reconstructed from overlapping reads displayed underneath. The coverage varies substantially due to random shearing during the shotgun process. (adopted from `http://www.cbcb.umd.edu/confcour/CMSC423-materials/Genome_assembly.pdf`) B: Dependence of the number of contigs on coverage for a 1Mbp source sequence due to the Lander-Waterman equation [1988]. Evidently, a too low coverage would result in a considerably fragmented assembly. (adopted from `http://www.cbcb.umd.edu/research/assembly_primer.shtml`)

is wet. By analogy, the source sequence has to be oversampled several times to make sure (or highly likely) that any base is covered by at least one read [Pop, 2004].

One assumption, that assembly algorithms rely on, is that every read corresponds to a fraction of the source sequence. While this is generally justified, incorrect reads do occur for which the above is only partially true. The distortion of such reads originates in the false detection of one or more bases in the course of the sequencing process. There is a variety of sequencing errors, the simplest of which are base insertion, base deletion and base substitution. The error probability can be estimated and is recorded in terms of quality scores. Some sequencing methods are more reliable than others, which is reflected by their quality score height. The problem of incorrect reads can be fixed, when the affected stretch of the source sequence is covered by multiple reads (Fig. 2.4). The required coverage in order to correct most errors varies with the sequencing platform used. For instance, with Sanger sequencing 3-fold coverage is usually sufficient even for diploid genomes, while for Illumina reads (at lengths of around 35bp) 20-fold coverage is necessary to assemble bacterial genomes [Chan, 2009; Dohm et al., 2008].

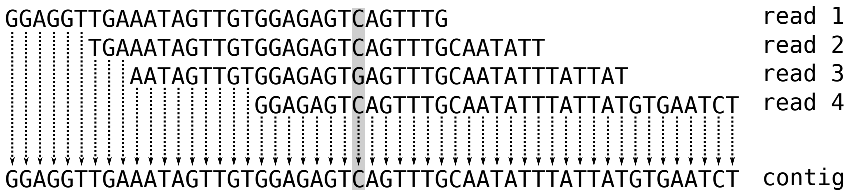

GGAGGTTGAAATAGTTGTGGAGAGTCAGTTTG read 1
TGAAATAGTTGTGGAGAGTCAGTTTGCAATATT read 2
AATAGTTGTGGAGAGTGAGTTTGCAATATTTATTAT read 3
GGAGAGTCAGTTTGCAATATTTATTATGTGAATCT read 4
GGAGGTTGAAATAGTTGTGGAGAGTCAGTTTGCAATATTTATTATGTGAATCT contig

Figure 2.4: Error-correction. Sequencing errors can be detected and often corrected if multiple reads cover the relevant stretch of the source/the contig. Thus, sufficient coverage is an essential requirement of error-correction. The shaded base in read 3 is identified as incorrect and can be corrected by means of the other aligned reads (1,2 and 4). Fig. based on Pop [2004].

Another assumption is that overlapping reads come from the same location on the source sequence. However, this is not necessarily the case [Pop, 2004]. Repeats, i.e., sequences that occur more than once within the source sequence, can induce overlaps between reads that originate from different regions within the source. Such reads cannot always be assembled unambiguously. Obviously, the difficulty is to differentiate between 'true' and repeat-induced overlaps. If a repeat is longer than the read length, paired-end information – if available – can be used to resolve ambiguity by bridging the repeat. Repeats longer than the distance between the mates are still not solvable. Also coverage estimates are taken into account when assembling repetitive stretches (see also Fig. 2.6 and Fig 2.7). Unresolved repeats yield a fragmented assembly at best and a misassembly in the worst case. In practice, sequencing errors and repeats render a correct reconstruction of

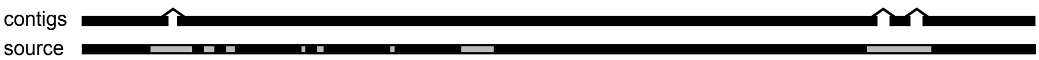


Figure 2.5: Sketch of the fragmentation of an assembly due to repeats. The source sequence (lower line) contains eight repeats (light grey). The upper line represents the assembly consisting of four contigs. The four short repeats are resolved by coverage estimates and paired-end information, while the longer ones cause the gaps in the assembly. The order of the contigs is inferred from pairs that span the gaps (cf. scaffolding, section 2.3.3). Fig. based on Pop and Salzberg [2008].

an entire source sequence next to impossible without additional information [Pop, 2004]. Given sufficient coverage and paired-end data, an assembly consisting of a small number of contigs is a more realistic objective in most cases (Fig. 2.5).

2.3.1 A greedy approach for the shortest common superstring problem

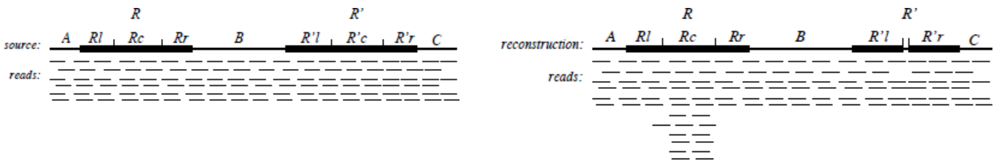
The fragment assembly problem can be modeled by the shortest common superstring problem (SCS), i.e., given a set of strings $\Sigma = \{s_1, s_2, ..., s_n\}$, to find the shortest string S that holds: s_i is a substring of S for all i, $1 \leq i \leq n$. When considering a graph representation of this problem, it becomes obvious that the problem is equivalent to the Travelling Salesman Problem (TSP), i.e., to find the shortest path that visits every node of a graph at least once. The TSP is known to be NP-complete and so is the SCS [Huson, 2010].

The SCS does not model repeats and sequencing errors and is thus an idealized setting of the FAP. While the SCS can be modified to include errors [Huson, 2010], it is not adaptable to model repeats. Thus, it proved suitable to only a limited extent when trying to solve the FAP in practice, but it remains useful for a better understanding of the situation [Pop, 2004].

In the context of the SCS, greedy algorithms provide a straight forward strategy to assembly: Beginning with individual reads as single contigs, merge the two contigs that have the maximum overlap and iteratively continue to do so until there is nothing left to join.

1. Compute all pairwise overlaps of the set of reads R.
2. Considering all reads as contigs, merge the two contigs that overlap best.
3. Repeat step 2 until no more joins possible.

This greedy approach is relatively efficient and quite easy to implement. However, as mentioned above, the approach is not feasible to handle repeats. Here, the outlined greedy strategy can lead to misassemblies, e.g., overcompressions because it takes only local information into account (Fig. 2.6).



(a) **(b)**

Figure 2.6: Repeat-induced overcompression using the greedy strategy outlined above. The source sequence (a) contains a two-copy R=R'. In the assembly (b) the part Rc=R'c is overcompressed. From Huson [2010].

2.3.2 Graph theory approaches

As seen in the previous section, the SCS model is suboptimal when the shortest solution is not the correct one. Due to repeats, this is the case often enough. Accordingly, the corresponding greedy-algorithm – inherently optimizing a local objective function (here the overlap-quality of two reads/contigs) – does not necessarily lead to a globally optimal solution [Huson, 2010; Pop, 2004; 2009].

However, there are two graph-based approaches to assembly that pursue different strategies: The Overlap-layout-consensus (OLC) and the de Bruijn graph approach.

2.3.2.1 Overlap-layout-consensus

The overlap layout consensus approach divides the assembly into three basic steps:

1. Overlap phase: the reads are compared to each other resulting in a list of pairwise overlaps, which is then used to construct the overlap graph. This is a graph containing each read as a node and an edge between every two nodes whose corresponding reads overlap.
2. Layout phase: The reads are grouped based on their alignments/overlaps and ordered relatively within their group. This corresponds to identifying (sub)paths in the graph that correspond to sections of the original source sequence.
3. Consensus phase: The (sub)paths found in the layout step are converted into contigs or (in the optimum case) just one contig.

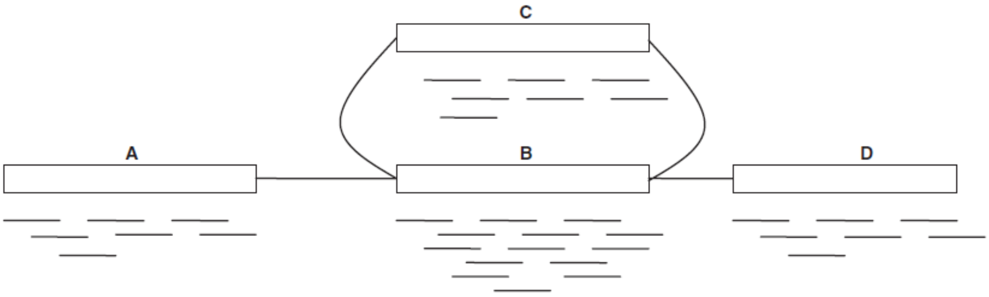


Figure 2.7: Schematic overlap graph of a partially repetitive source sequence. According to their overlaps, the reads were assigned to the connected groups A,B,C and D. By means of coverage comparisons, group B is recognized as a two-copy repeat and the original source sequence is infered to be ABCBD. Fig. from Pop [2009].

Ideally, the result of the consensus phase is a single contig equal to the original source sequence that corresponds to a single path through the overlap graph visiting each node exactly once. To find such a path is the Hamiltonian path problem which is known to be NP-hard.
However, unlike the greedy approach the OLC approach is appropriate to represent repeats (cf. Fig. 2.7; compare to Fig. 2.6).
In the context of Sanger-based sequencing OLC-based assembly tools were used with great success. When applied to NGS data, however, the huge number of reads severely scales up the graph and thus aggravates the contig calculation. Additionally, due to the short reads the calculated overlaps need to incorporate most of the participating reads [MacLean et al., 2009].

2.3.2.2 De Bruijn graph approach

A string of length k is called a k-mer. The k-mer spectrum of a string s is the set of all k-mers that are substrings of s.
Given a set of reads, the de Bruijn graph is constructed as follows (see also Fig. 2.8):

1. Construct the k-mer spectra of all reads.
2. Construct the (k-1)-spectrum of each unique k-mer, i.e., two (k-1)-mers per k-mer.

3. Create a node for each (k-1)-mer.
4. Create an edge between two nodes, if the corresponding (k-1)-mers can be merged into one of the k-mers.

In this setting, the assembly problem is to find an Eulerian path in the de Bruijn graph, i.e., a path that traverses every edge in the graph.

```
source sequence: ACCATTCCAA

reads:           ACCATTC, ATTCCAA

k-mers:          ACCA, CCAT, CATT, ATTC, TTCC, TCCA, CCAA

(k-1)-mers:      ACC, CCA, CAT, ATT, TTC, TCC, CAA
```

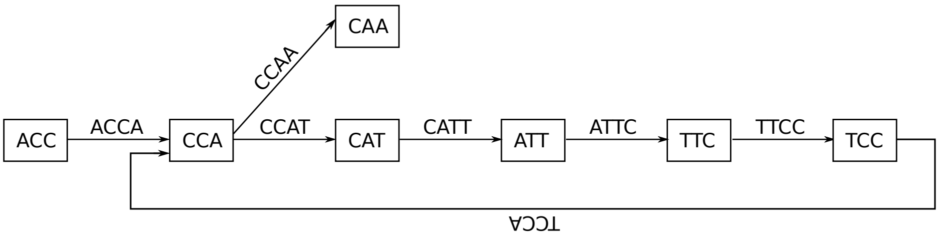


Figure 2.8: De Bruijn graph example. The source sequence is covered by two reads. The unique k-mers and (k-1)-mers are incorporated in the corresponding de Bruijn graph as described in the text.

Unlike the Hamiltonian path problem, the Eulerian path problem is generally solvable in polynomial time. However, there are potentially multiple Eulerian paths in a graph, and to find the best one – with respect to the given constraints – is not as easy. Medvedev et al. showed, e.g., that to find the shortest Eulerian path in a de Bruijn graph is NP-hard [2007].
Furthermore, the approach is very sensitive to read errors, as each error leads to the creation of additional k-mers. In consequence, the approach and implementations employing it, e.g., the Euler assembler [Pevzner et al., 2001], were not as popular at first.
However, the advent of NGS platforms led to the establishment of the strategy as the abundance of short reads generated here can be handled better than with the OLC approach. In the OLC approach the construction of the overlap graph is very demanding when confronted with an abundance of short

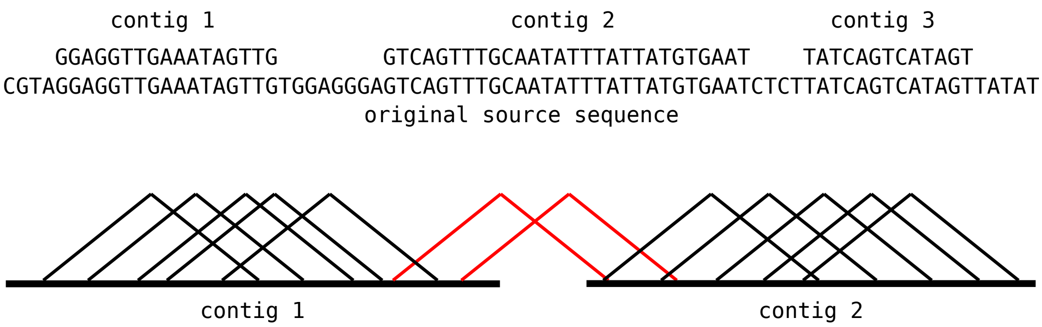

contig 1
contig 2
contig 3
GGAGGTTGAAATAGTTG
GTCAGTTTGCAATATTTATTATGTGAAT
TATCAGTCATAGT
CGTAGGAGGTTGAAATAGTTGTGGAGGGAGTCAGTTTGCAATATTTATTATGTGAATCTCTTATCAGTCATAGTTATAT
original source sequence
contig 1
contig 2

Figure 2.9: Contigs and scaffolds. A: Due to insufficient coverage an assembly of the reads acquired from the original source sequence resulted in the three contigs 1, 2 and 3. They correspond to consecutive sections of the original source sequence. However this is initially not known. B: The order of the contigs can often be inferred by use of paired-end information. In this example two pairs indicate the order, distance and relative orientation of contig 1 and 2.

reads. By contrast, in a de Bruijn graph only unique k-mers/(k-1)-mers are incorporated instead of all the reads.

2.3.3 Scaffolding

The output of any assembler – independent of its underlying algorithmic paradigm – consists of more than just one contig in the vast majority of cases. The relative orientation, order and distance with respect to each other is initially unknown. Paired-end data – if available – can be used to deduce some of this information. If a pair links to contigs, it indicates their relative orientation and – except in case of quite short contigs – order as well as an approximate distance. The information of all relevant pairs can be incorporated in a graph with contigs as nodes and linking pairs as edges. A preferably small number of coherent arrangements of contigs – called scaffolds (Fig. 2.9) – can then be extracted from the graph [Pop, 2009].

3 Mapping of WDLPS neochromosomes

The main overall objective here is to determine the structure of the sequenced well-differentiated liposarcoma (WDLPS) neochromosome.

Liposarcoma is a malignant tumor occurring in fat cells of soft-tissue. With an annual incidence of 2.5 per million population, it is the most common type of soft-tissue sarcoma in adults, the 5-year survival rate is below 50% [Schwartz et al., 2010].

Starting out from the most important cytogenetic and molecular aspects of WDLPS, a concise description is given of the data to be analyzed. An outline of the general analysis strategy is followed by a more comprehensive elaboration of the techniques employed here. To test its implementation and evaluate its feasibility, the strategy is applied to a synthetic dataset, the structural properties of which are known.

3.1 Cytogenetic features of WDLPS

Most WDLPS tumors are cytogenetically characterized by one or more cancer-associated neochromosomes that occur in addition to the 46 normal chromosomes. These neochromosomes are found in both, a ring and linear-shaped ('giant rod') topology (Fig. 3.1; compare Fig. 2.1c), with the linear form supposedly emerging from the ring through linearisation [Garsed et al., 2009; Sandberg, 2004].

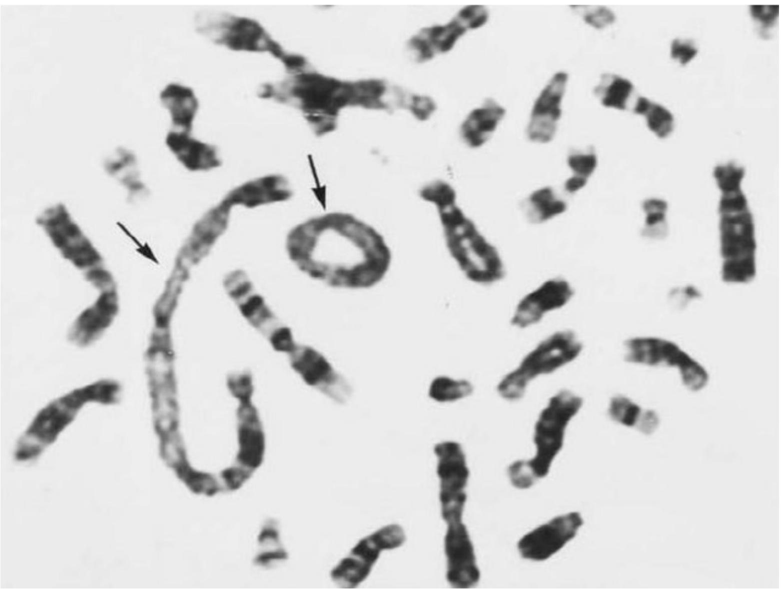


Figure 3.1: Common cytogenetic features of WDLPS. In this segment of the metaphase of a WDLPS both a ring and a linear-shaped neochromosome are present. Figure extracted from Sandberg [2004].

The genetic material of neochromosomes consists of "donated" fragments from different normal chromosomes (Fig. 3.2). The incorporated parts are subject to substantial rearrangement, amplification, and modifications of one sort or another. While it is assumed that WDLPS is driven by these genomic alterations contained within neochromosomes, the exact mechanism is still unknown. The involved donor sites have already been identified to a great extent, however, their order and the fusion sites are not as well characterized, yet [Garsed et al., 2009].

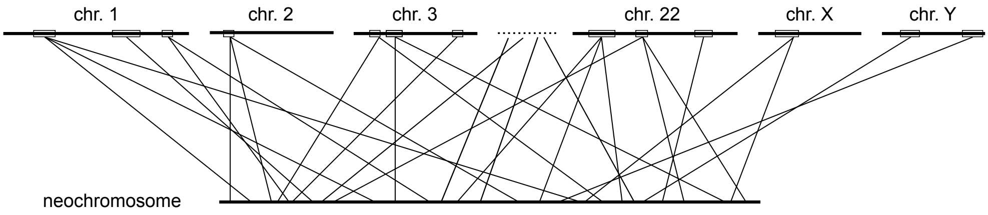


Figure 3.2: Conceptual sketch of neochromosome structure. The cancer chromosome is presumably composed of substantially rearranged and modified genetic material from normal chromosomes.

3.2 Data

We were provided with three sets of reads from a giant rod chromosome found in cells of a patient suffering from WDLPS.

Due to its size the giant rod is heavier than normal chromosomes. Hence, it could be successfully separated in the laboratory using float-sorting. Prior to our analysis three runs of sequencing using the Illumina platform were conducted, resulting in three sets of reads, viz.:

1. ∼12 million pairs of 35bp length
2. ∼12 million pairs of 75bp length
3. ∼12 million single-end reads of 75bp length.

The read data adds up to a total of $(24000000*(35+75)+12000000*75)$bp $\approx$ 3500Mbp corresponding to ∼3.6 GB of disc space when stored as FASTA-files[1].

The giant rod has a size of approximately 650Mbp. Thus, the overall coverage is approximately 5-fold (3500Mbp/650Mbp).

3.3 Analysis strategy

The analysis of the data is essentially conducted in three steps (Fig. 3.3). Because the use of a reference genome to support the assembly process is inappropriate in the face of significant genomic rearrangements, the first step consists in de novo assembly of the read data (section 2.3). Due to the sizeable amount of short reads, the de Bruijn graph approach would appear better suited than the OLC. Hence, for data assembly here we chose Velvet [Zerbino and Birney, 2008], which is one of the most widely used assembly tools based on the de Bruijn graph approach (section 2.3.2.2).

It seemed more efficient to analyze the contigs resulting from the assembly separately from the de Bruijn graph itself, which at a size exceeding 10 GB is quite difficult to handle. Though a closer look at the graph would appear

[1] A FASTA-file contains sequences in the following format: For each sequence a one line description beginning with '>' is followed by one or more lines containing the sequence.

worthwhile, it was deemed necessary, also on account of time constraints as to the internship, to opt for a further analysis of the contigs only.
In the second step, the short read alignment tool Bowtie [Langmead et al., 2009] is utilized to map the individual reads back to the contigs. In a scaffolding manner we try to detect and characterize connections between the contigs by analyzing the paired-end data. In the resulting contig-graph, each contig is represented by a node and each detected connection between contigs is mapped as an edge connecting corresponding nodes. The detected connections differs in the number of pairs supporting them. If the support is below an adjustable threshold, the connection is flagged as unreliable.

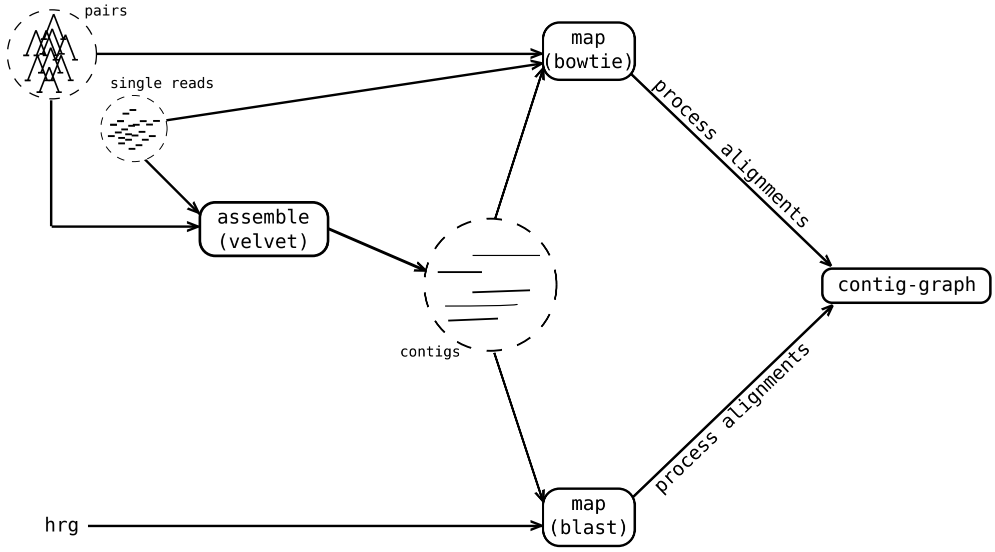


Figure 3.3: Overview of analysis strategy.

In the third step the contigs are mapped to the human reference genome using BLAST [Altschul et al., 1990]. The BLAST-results are postprocessed to select and, if necessary, to combine the alignments in order to identify the donor sites – i.e. the location of contigs or parts of them on the human reference genome – with highest possible confidence. Subsequently, this information is added to the contig-graph, which is then searched for the most interesting nodes and connections. The nodes containing regions from different locations of the human reference genome and the edges connecting such nodes are indicative of fusion sites on the neochromosome.

The strategy was first implemented and tested using synthetic data (section 3.3.3). Finally, the analysis chain was applied to the real data.

3.3.1 Read-contig mapping

The contigs obtained through assembly need to be analyzed as to their relative placement with respect to each other. For this purpose we use an approach that follows the idea of scaffolding, briefly described in section 2.3.3. In case the distance of two contigs is less than the expected distance of paired-end reads, it can be assumed – given sufficient coverage – that there are pairs that 'bridge' the gap between the contigs (Fig. 2.9): Finding one read of a pair on the first contig and the other on the second, indicates that the two contigs involved were connected on the original sequence, i.e., the neochromosome here.

We use Bowtie [Langmead et al., 2009] in single read mode to map the reads to the contigs. The alignments are structured as shown in Table 3.1.

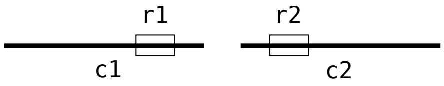
The post-processing of the Bowtie-alignments is conducted using the most reliable information. Out of all reads those having just one alignment can be assigned to the aligned location with highest confidence provided there are no mismatches. Similarly, when both reads of a pair are mapped uniquely, the pair can be used to identify connections between contigs with highest confidence.

read id	strand	contig id	position	sequence	mismatches
17	−	409	768	TGGGAAGGTAAGTTATTTTTTATG	0
52	+	7156	1126	ATTTTGTCACATCTACCAATACTG	0
53	−	7156	1344	TTGTTGAACGTTCTTTAGTTCAGA	0
76	+	24429	2816	TTTTGGTGAGAAGAATATGTATTT	0
77	−	24429	3047	AGAGAGGGCACTTCTGTTGTGCCC	0
⋮	⋮	⋮	⋮	⋮	⋮

Table 3.1: Structure of Bowtie alignments. The table exemplifies the structure of the Bowtie alignments. Each line corresponds to an alignment. In each alignment the following fields are specified: read id, strand of the contig to which the read was aligned, contig id, position, i.e., offset with respect to the start of the contig, nucleotide sequence and number of mismatches.

All fully mapped - i.e. unambiguously aligned - read pairs are subject to further analysis. The initial conditions are formally described through eq. (2.1) (cf. section 2.3). Moreover, each two reads r_k and r_{k+1} form a pair, if $k \bmod 2 = 0$. Let c_1 and c_2 be contigs. We say c_1 and c_2 are connected by the pair formed by r_k and r_{k+1}, if r_k is aligned to c_1 and r_{k+1} is aligned to c_2 or vice versa.

Let $k = 1$. If c_1 and c_2 are connected by r_1 and r_2 there are several scenarios to consider. To begin with, there are the cases $c1 = c2$ and $c1 \neq c2$. Either of them ramifies further due to side conditions as to read orientation in that r_1 and r_2 may have equal (strand(r_1) = strand(r_2)) or opposing (strand(r_1) $\neq$ strand(r_2)) directions. As illustrated in Fig 3.4, this leads to four scenarios altogether.



A: $c_1 = c_2 \land \text{str}(r_1) \neq \text{str}(r_2)$

B: $c_1 = c_2 \land \text{str}(r_1) = \text{str}(r_2)$

C: $c_1 \neq c_2 \land \text{str}(r_1) \neq \text{str}(r_2)$

D: $c_1 \neq c_2 \land \text{str}(r_1) = \text{str}(r_2)$

Figure 3.4: Analysis of pairs. Each fully mapped pair is subject to further analysis. There are basically four scenarios to consider, labelled A, B, C and D.

The relative placement of two contigs on the original source sequence, i.e., whether c_1 is followed by c_2 or vice versa, may be established through a case-by-case analysis. The decision as to which contig sequence is most probably correct is based on contig distance estimates to be performed for all possible cases A through D. Distance estimation in either case involves the positions (offsets) of a read pair found on any two contigs. In the following, these distances are specified in terms of contig length $|c|$, read position, read length $|r|$, and unknown gap length x between contigs.

Depending on the relative positioning of c_1 and c_2 with respect to each other there are two possible contig distances for case C (cf. Fig. 3.5), viz.:

$$\text{C1: } d = |c_1| - \text{position}(r_1) + x + \text{position}(r_2) + |r_2|$$

$$\text{C2: } d = |c_2| - \text{position}(r_2) + x + \text{position}(r_1) + |r_1|$$

In case D both reads are mapped to the same strand, which violates the complementary strand constraint of paired-end reads (cf. Fig. 2.2e). This implies that one of the contigs needs to be flipped in order to meet this stipulation. The flipping would appear to cause two additional distance measures. However, since two contig arrangements are equivalent to one another, there are effectively two possibilities. As can be inferred from Fig. 3.5:

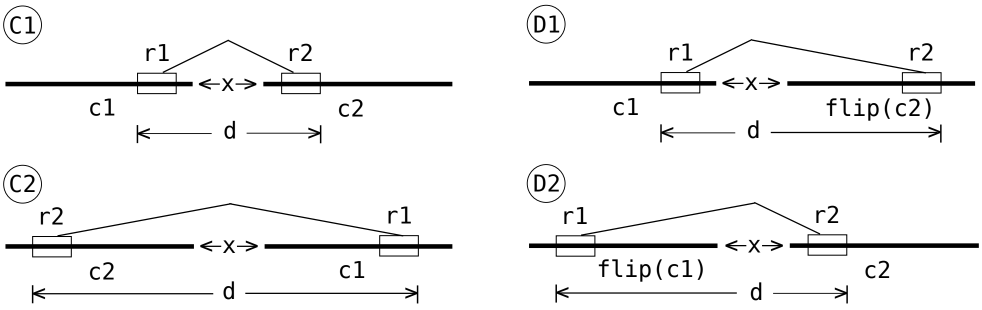


Figure 3.5: Analysis of pairs - different contigs.

D1: c_1 followed by flip(c_2) $\Leftrightarrow$ c_2 followed by flip(c_1)

$$d = |c_1| - \text{position}(r_1) + x + |c_2| - \text{position}(r_2)$$

D2: $flip(c_1)$ followed by (c_2) $\Leftrightarrow$ $flip(c_2)$ followed by (c_1)

$$d = \text{position}(r_1) + |r_1| + x + \text{position}(r_2) + |r_2|$$

Case B is analogous to D. Instead of two different contigs there are two instances of the same contig. Observing that $c_1 = c_2 = c$, the distances may be evaluated as with D1 and D2.

The same correspondence exists between cases A and C, such that the distances in the cases A1 and A2 may be determined as with C1 and C2, except for $c_1 = c_2 = c$. However, in addition to cases A1 and A2, which involve two instances of the same contig c, a third case A3 (Fig. 3.6) arises from the fact that both reads may be found on the same instance of c:

A3: $d = \text{position}(r_2) + |r_2| - \text{position}(r_1)$

Even though A3 appears irrelevant as to contig order assessment, it plays a key-role in threshold specification (see below). Note also that an equivalent case B3 is rendered impossible by the complementary strand constraint of paired-end reads.

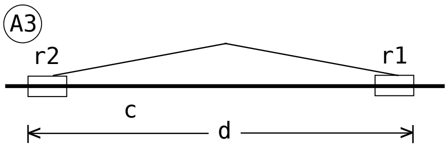


Figure 3.6: Analysis of pairs - same contig.

The estimated distances for all possible arrangements (A1 through D2) of each read pair are compared to the expected distance resulting from the average read distance (also known as insert size) used during the sequencing process (cf. Fig. 2.2e). A threshold defined as the maximum allowable difference between expected and estimated distance can be inferred from distance variations in read pairs from a case A3-only analysis. This threshold, subsequently, serves in selecting the most probable distance estimate from the distance ensemble corresponding to A1 through D2. Read pairs corresponding to A3 are used in estimating the coverage of the contig they are found on, while pairs associated with any other case indicate the type of contig connection as well as the approximate distance of the contigs involved. The information gained from connection-inducing pairs is utilized to construct a contig-graph. In this, nodes and edges between nodes represent contigs and detected contig connections, respectively.

Let e be an edge connecting two contigs. Then, e has two options of binding at each end. It can bind to the positive or the negative strand of each contig. In other words, it connects either to the contig or to the flipped version of the contig. This results in two distinct types of edges. The first type corresponds to all connections that are induced by pairs whose mates are in correct orientation with respect to each other. This type represents connections of the kind c1 – c2 and c2 – c1. The second type represents connections induced by pairs whose mates where mapped to the same strand, viz. c1 – flip(c2) and flip(c1) – c2. An example is given by Fig. 3.7. Type 1/2 connections are

depicted as straight/curvilinear lines.
Furthermore, each edge is attributed the number of pairs supporting it. The more pairs support a connection the more reliable it is. The supporting pairs also indicate the approximate distance of the connected contigs. The number of pairs and the indicated connection are denoted in the format 'number of pairs/ approximate distance' as edge label.

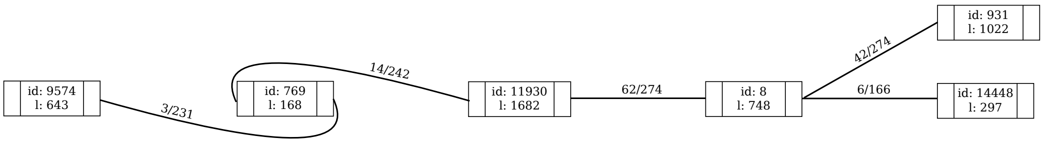


Figure 3.7: Contig-graph example.

3.3.2 Contig-reference mapping

As described in the previous section, the outcome of read-contig-mapping and processing of alignments is a graph containing the contigs and their connections. In this section we explain how the graph is extended by adding information obtained by mapping the contigs to the human reference genome using BLAST [Altschul et al., 1990]. This way, it is aimed at identifying the contigs' origin(s) on the normal chromosomes.

An example of the BLAST-results is given in Table 3.2. For each contig that has alignments on the human reference genome there is a result of the displayed structure. Each alignment has the following properties: the query id (which correponds to the contig id) and the subject id (corresponding to the id of the chromosome from the human reference genome); identity, length, number of mismatches and number of gap openings; the location on the query and on the subject; the e-value and the bit score. Each line corresponds to an individual alignment.
As with the example of Table 3.2, there are many obvious cases as to what is the best, i.e., most probable, deduction that can be drawn from the alignment. However, there are also other cases in which the situation is not assessable as easily. Then, combinations of alignments need to be considered in order to find the most probable deduction.

q.id	s.id	id. (%)	length	mm	gaps	q.s	q.e	s.s	s.e	e-val.	bit sc.
871	12	100.00	3572	0	0	1	3572	73852464	73856035	0.0	7081
871	9	93.02	86	6	0	1	86	31673785	31673700	$8e^{-25}$	123
871	2	92.31	91	4	1	1	88	214304525	214304435	$3e^{-24}$	121
871	11	92.05	88	7	0	1	88	108622082	108622169	$1e^{-23}$	119
871	14	92.05	88	7	0	1	88	51000788	51000875	$1e^{-23}$	119
871	1	91.01	89	8	0	3	91	143900301	143900389	$8e^{-22}$	113
871	1	91.76	85	7	0	1	85	33488862	33488946	$8e^{-22}$	113
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

Table 3.2: Structure of Blast alignments. The table exemplifies the structure of the Blast alignments. From left to right the columns contain: query id, subject id, identity (%), alignment length, number of mismatches, number of gap openings, query start, query end, subject start, subject end, e-value and bit score. Each line corresponds to one alignment.
In the example the contig 871 was most probably on chromosome 12 at position 73852464. This can be infered from the comparison of the alignments: The first alignment has an identity value of 100% and is also a lot longer than the other alignments. This leads to a superior e-value and bitscore.

A simple heuristics is applied to process and classify the obtained alignments: First, find the most obvious cases and assign a reference-location to the corresponding contigs. Then, look at the alignments of 'neighbours' of already assigned contigs in the graph and assign reference-locations to them, too, if possible. Build combinations of alignments starting with the most reliable ones and choose the best option where necessary. The retrieved information is used to extend the contig-graph (see example in Fig. 3.8).

The contig-graph can now be searched for connections of interest. Most importantly, edges between nodes corresponding to contigs assigned to distant regions on the reference genome represent the breakpoints, that are likely to contain gene modifications like gene fusions or truncations.

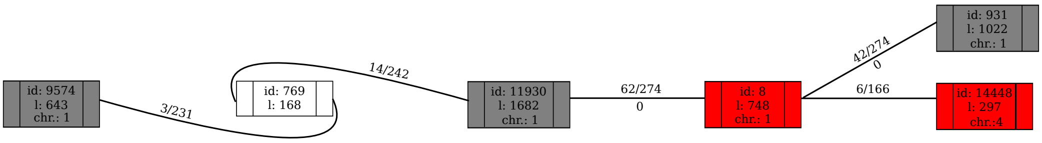


Figure 3.8: Contig-graph example after extension.

3.3.3 Test data generation

During the process of development and for the purpose of testing the implementation synthetic data were generated whose structural properties are known. It was the aim to model the assumed situation described in section 3.1 to the necessary extent (see Fig. 3.2).
The generation of test data sets was conducted in three steps.
First, a reference sequence corresponding to the 24 different chromosomes was randomly created. For the sake of simplicity, only one reference (instead of 24) was used here. In the second step, a 'cancer sequence' was assembled by using fragments randomly extracted from 'donor sites' on the reference. Finally, a double barrel shotgun sequencing experiment was simulated by randomly extracting paired-end reads from the cancer sequence.
The implementation of the data generator along these lines features a variety of adjustable parameters that are summarized in Table . The implementation of the described steps comprises different parameters, which are outlined in Table 3.3.

step	parameter	description
reference generation	alphabet	e.g. {A,C,G,T}
	reference length (bp)	e.g. 100000
cancer sequence generation	fragment length $\sim \mathcal{N}(\mu, \sigma^2)$	the fragment length is modelled as a gaussian-distributed variable, the mean μ and standard deviation σ of which can be specified
	# extracted fragments	number of different fragments extracted from the reference
	# composing fragments	number of fragments randomly taken from the set of extracted fragments used to compose the cancer sequence
read extraction	read length (bp)	e.g. 75
	read distance $\sim \mathcal{N}(\mu, \sigma^2)$	the read distance is modelled as a gaussian-distriubted variable, too. (see fragment length)
	overall coverage	the number of reads that need to be extracted is determined by the term $\frac{\lvert\text{cancer sequence}\rvert * \text{overall coverage}}{\text{read length}}$.

Table 3.3: Parameters of test data generation.

3.3.4 Implementation

The strategies described above were implemented in the JAVA programming language. Its almost-independence of platform and operating system as well as the availability of libraries usable for visualization and GUI were all-important in choosing this language.
The necessary input of the program is the paired-end sequencing data in form of fasta files. Furthermore, the implementation depends on external tools for assembly and alignment as well as the human reference genome. The external applications Velvet, Bowtie, and BLAST can be called from within the program provided the paths to the installation directories are given. Alternatively, the assembly and alignment results can be given as additional input files. While the tools named above were chosen here, it is generally possible to use others.
The workflow of the implementation can be summarized as follows:

1. Import the reads from the specified FASTA-file(s).
2. Call Velvet and/or import contigs from FASTA-file (Velvet-output).
3. Call Bowtie and/or import read-contig alignments (Bowtie-output).
4. Analyze paired-end information and construct contig-graph (section 3.3.1).
5. Call BLAST and/or import contig-reference alignments (BLAST-output).
6. Analyze contig-reference alignments and enhance contig-graph (section 3.3.2).
7. Search contig-graph for interesting connections and write output.

The output consists of a text-file containing information on the extracted connections. Also on output are the contig-graph as a dot-file[2], and a text-file summarizing some statistics. Among the latter are the number of input reads located on contigs or the number of contigs assigned to a position on the human reference genome.
The program may be run in command line mode or from a simple graphical user interface (GUI). Display of status messages and interim results as well as saving and loading options for discrete intermediate stages of the applications are among the features of the GUI.

[2] By means of the dot-file the graph can be visualized using Graphviz, an open-source collection of graph drawing tools [Ellson et al., 2002].

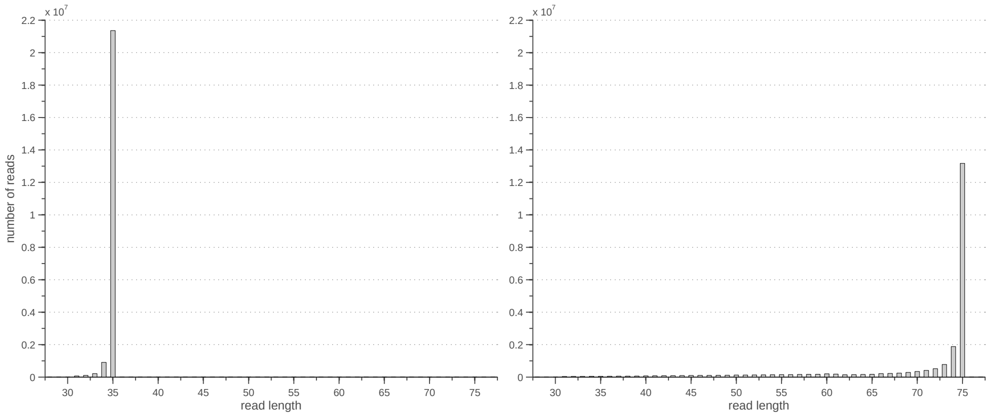
4 Results

Paired-end information is essential to the detection of connections between contigs and thus to the generation of the contig-graph. Hence, it is deemed sensible to present some characteristics of the two sets of paired-end reads to begin with.

The distributions of read lengths for set 1 and set 2 are shown in Fig. 4.1. The target read lengths were 35bp and 75bp, respectively. While most of the reads of either set match the target length, an exponential drop off to shorter read lengths is evident from Fig. 4.1. Typically the quality scores tend to decrease towards the end of a read. If quality falls short of some minimum score, the associated read is trimmed. As may be inferred from Fig. 4.1, the reads of set 1 were trimmed by five bases at most, while those of set 2 were truncated by up to 35 bp or half the nominal read length.

A rather low, about 5-fold coverage of the sequenced neochromosome made a considerably fragmented assembly expectable. In consequence, 30463 contigs with lengths of at least 100bp passed the internal coverage cutoff of Velvet. About 80% of the contigs fall in the range 100 – 1000 bp. As illustrated by Fig. 4.2, contig lengths are approximately exponentially distributed.

By way of read-contig-mapping (section 3.3.1) it was found that the majority of read pairs, viz. 87% (set 1) and 91% (set 2), could not be aligned. The fraction of pairs that could be fully mapped was 6% and 3%, respectively. As for the rest, only one of the mates of a pair could be aligned (Fig. 4.3). In essence, there are two reasons for utilization ratios such low. On the one hand, reads belonging to contigs that fell short of the coverage cutoff used in Velvet cannot possibly be aligned. On the other hand, many reads failed to



(a) (b)

Figure 4.1: Distribution of read lengths for set 1 (a) and set 2 (b).

align on account of the abundance of short contigs. More specifically, reads not entirely belonging to a contig, but to some extent to a breakpoint inbetween contigs, cannot be found by the alignment as conducted here. Small though it may be, the read utilization rate still proved sufficient to produce useful results.

Prior to analyzing the mapped pairs and inferring the connections they are indicative of, the threshold – previously defined as the maximum allowable difference between expected and estimated read distance (cf. section 3.3.1) – had to be adjusted to fit with the data at hand. Hence, in a first

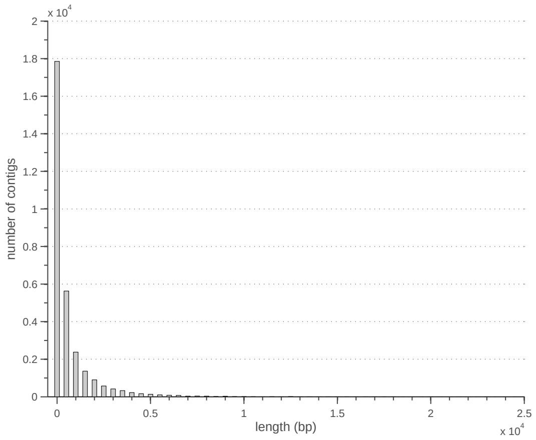


Figure 4.2: Distribution of contig lengths.

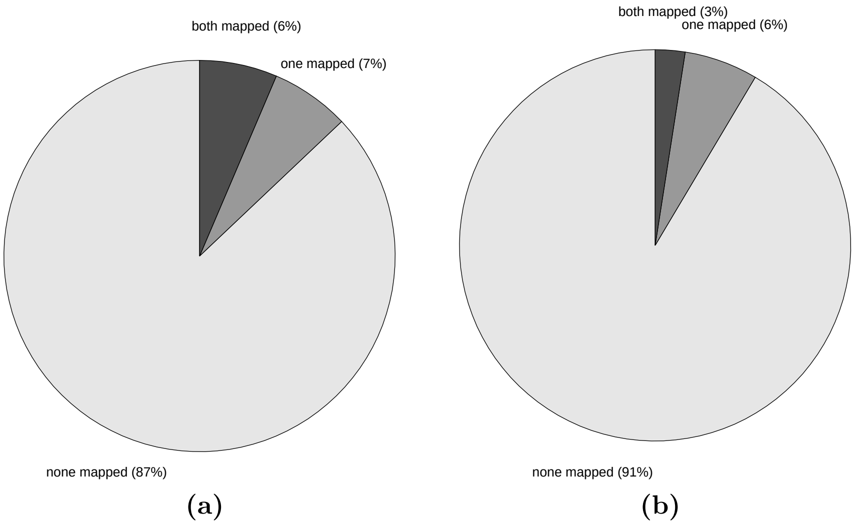
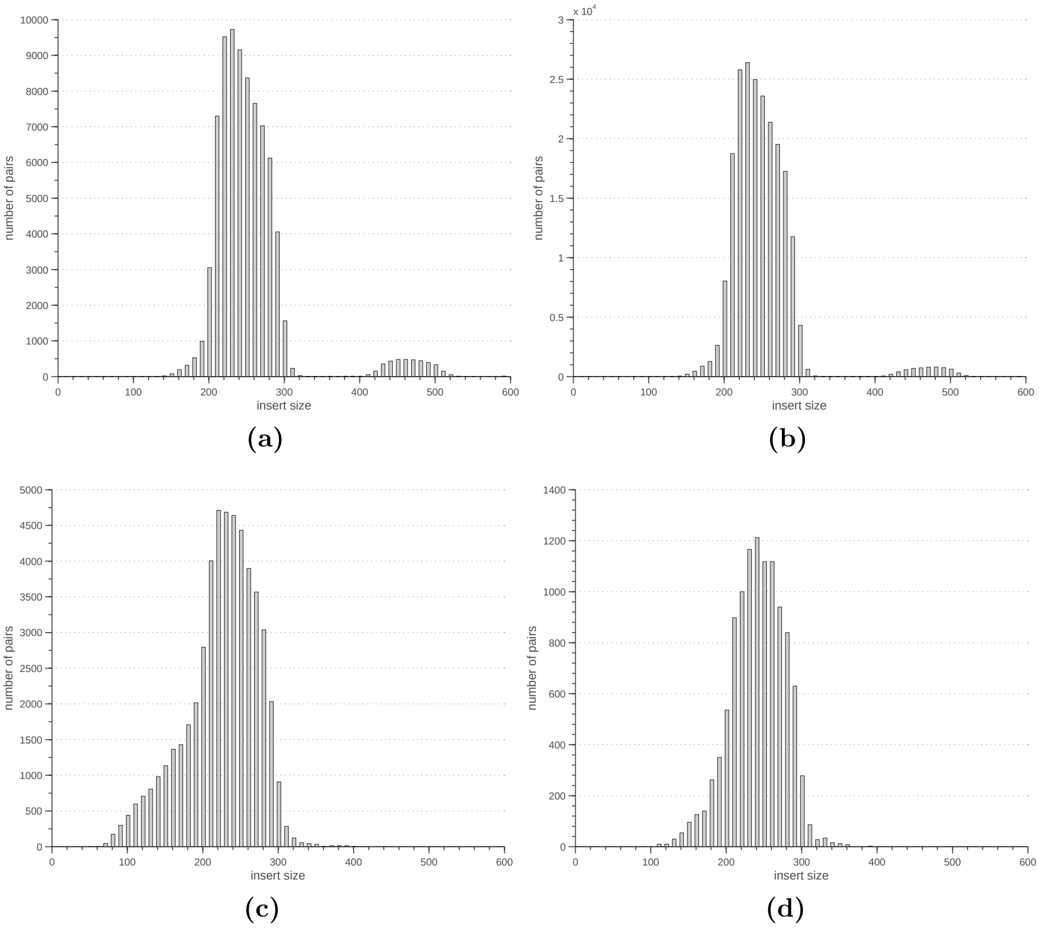


Figure 4.3: Utilization of read pairs in read-contig-mapping for set 1 (a) and set 2 (b).

run, a high threshold of 800 bp was chosen in order to permit very large deviations from the expected read distance or rather insert size. The insert size distribution of all pairs that could be unambiguously assigned to case A3 – viz. the same instance of the same contig (cf. section 3.3.1) – is displayed in Fig. 4.4 (top) for both read sets. These distributions are regarded representative of the complete read sets. The majority of pairs is halfway symmetrically distributed within a range of ±150bp around the expected insert size of 250bp. Surprisingly, a secondary mode is observed between 400 and 530bp. Bimodal distributions, which may be indicative of different populations, never occurred in application testing. Hence, it would appear not entirely fallacious to presume that the occurrence of the minor mode is associated with the sequencing step carried out in the laboratory. A definite cause, however, is not obvious. In consequence, the threshold was set to 150bp, corresponding to difference between expected distance and the lower limit of the spurious mode.

Thereupon the adjusted threshold was used in conducting the actual read-contig mapping as described in section 3.3.1. The distribution of insert sizes of read pairs linking different contigs is presented in Fig. 4.4 (bottom) for both read sets. As compared to the high threshold distributions (Fig. 4.4, top) they appear right-shifted by some 30bp. This reflects the k-mer size of Velvet, as adjacent contigs overlap by up to k-1. Moreover, the number of pairs with insert sizes ranging from twice the read length to 200bp increased. This range corresponds to pairs connecting contigs that do not border on one another. In these cases the estimated distance is too small as the distance inbetween contigs is not accounted for.

Just about 5% or 1 out of 22 million pairs comprising the two sets of



(a) **(b)**

(c) **(d)**

Figure 4.4: Distribution of insert sizes. Left: set 1, right: set 2; top: pairs assigned to one contig (same instance), threshold 800bp, bottom: pairs linking different contigs, threshold 150bp.

paired-end reads were successfully mapped to the contigs of the assembly. The analysis of order and orientation of the contigs with respect to one another is thus based on just this fraction of reads.

In spite of this, the combined strategy of read-contig mapping and contig-reference mapping revealed 188 fusion sites on the neochromosome. In the contig-graph (for an example cf. Fig. 4.5), these fusion sites correspond to connected contigs which were mapped to far apart locations or rather donor sites on the human reference genome. Among these were 94 (intrachromosomal) breakpoints between distant locations on the same chromosome and 94 (interchromosomal) breakpoints between donor sites on different chromosomes. These results are detailed in Table 4.1 and Fig. 4.6.

	a	*b*	*c*	*d*	*e*	*f*	*g*	*h*	*i*	*j*	*k*	*l*	*m*	*n*	*o*	*p*	*q*	*r*	*s*
a	11		1	5	3	1			1	3	24	1	5			1			
b		1									1								
c											1								
d				5							6		1						
e					11					1	1						1		
f																			
g							7												
h								15											
i									2		4								
j										1	6	2	3						
k											16	4	5	1		3	5		1
l													1	1					
m													3						
n														4					
o															2				
p																			1
q																	1		
r																		10	
s																			5

Table 4.1: Identified fusion sites on the neochromsome. The entries give the number of breakpoints between regions on chromosome i and chromosome j. Zero entries were omitted for clarity, as were entries below the main diagonal since N(i,j)=N(j,i).

Obviously, some chromosomes act more frequently as donor sites than others. This is particularly true for chromosomes a and k, which were heavily involved in both, intra- (a:11, k:16) and interchromosomal breakpoints (a: 45, k: 62). The results hint to such translocation hotspots.

A biological interpretation is beyond the scope of this work. Moreover, on account of a non-disclosure agreement the identified fusions and their locations must not be revealed here in a more precise fashion. In particular, the chromosomes encoded here by characters a through x do not match their natural numerical order.

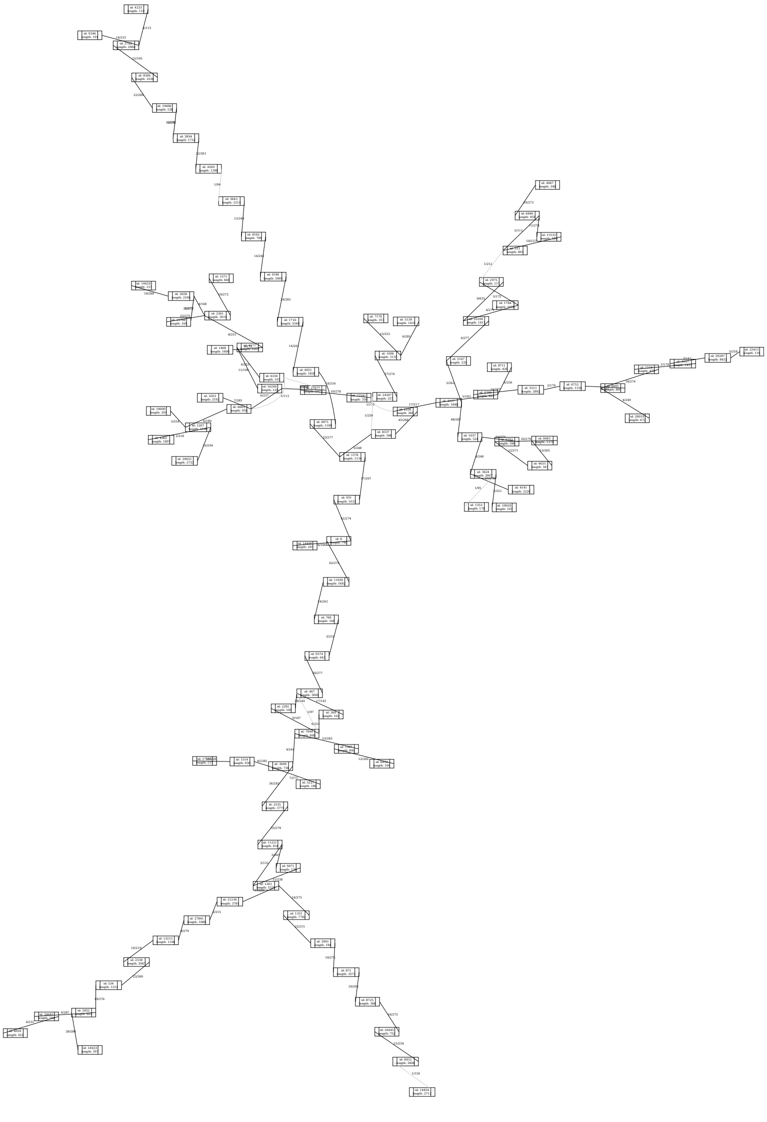


Figure 4.5: A subgraph of the contig-graph layed out using Graphviz.

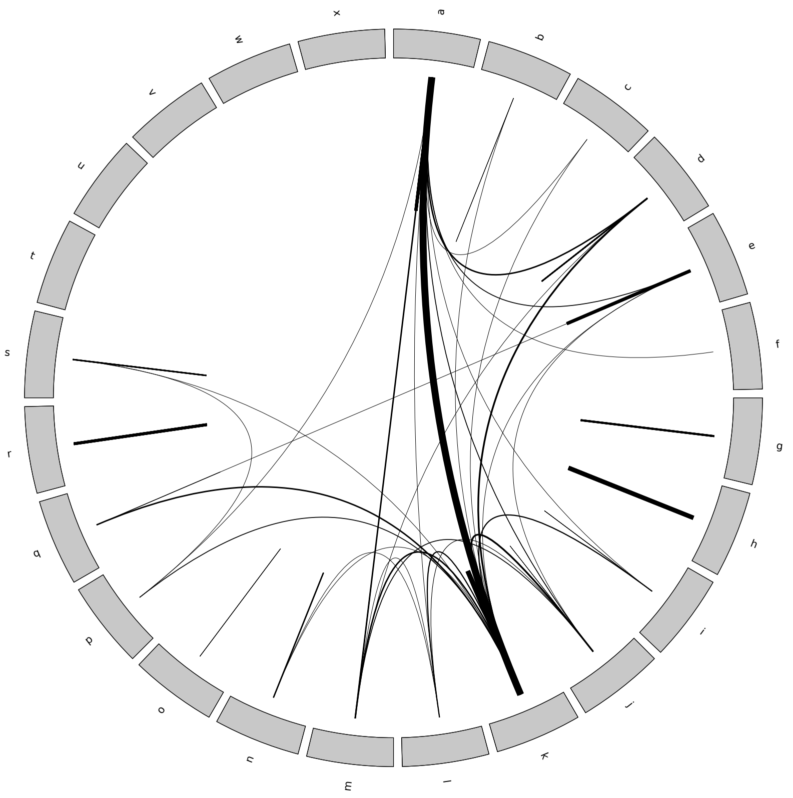


Figure 4.6: Indentified fusion sites on the neochromosome. The chromosomes are denoted a through x and displayed in disturbed order as to their natural numerical sequence. Lines connecting different chromosomes represent interchromosomal breakpoints, while dead-end lines stand for intrachromosal breakpoints. Line thickness is proportional to the number of breakpoints. The underlying data are detailed in Tab. 4.1. Fig. generated using Circos [Krzywinski et al., 2009].

5 Concluding remarks

In this study we proposed a strategy to identify and characterize structural rearrangements of genomes in comparison to a given reference. This strategy represents a chain of analysis composed of, not only public domain applications (Velvet, Bowtie, BLAST), but also self-developed tools, in order to postprocess, interrelate, and interpret intermediate results of the former. The pipeline of analysis was implemented in Java, validated through synthetic datasets with known properties, and eventually applied to paired-end sequencing data from a WDLPS neochromosome.

After an assembly of the sequencing data, the paired-end reads are aligned to the resulting contigs, followed by an alignment of the contigs to the reference. The contigs, read-contig alignments and contig-reference alignments are then utilized in the following way:

Based on an analysis of the read-contig alignments a contig-graph is constructed representing the different connections between contigs. The contig-reference alignments are analyzed in order to locate the origin of contigs or fractions of them on the human reference genome. The obtained information is added to the contig-graph. Finally, interesting connections, viz. those between contigs aligned to distant locations on the human reference genome, are extracted from the contig-graph.

This chain of analysis was designed and implemented to investigate the structure of WDLPS neochromosomes. Synthetic datasets were modelled after realistic data properties and used, not only in debugging and validating the implemetation during the phase of development, but also in assessing the feasibility of the chain or its components. After successful completion of several test runs as to its structural and functional integrity the full chain of analysis was applied to sequencing data of a WDLPS neochromosome.

Most importantly, a search of the final contig-graph revealed 188 breakpoints between components of different origin on the normal chromosomes. Among these 94 were associated with different regions on the same chromosome. The remainder of 94 breakpoints occurred between regions from different chromosomes.

These breakpoints are likely to comprise gene alterations that may contribute to the abnormal and malignant behaviour of the tumour cells. Thus, the further characterization of the breakpoints can provide valuable information, not only aiding diagnostics and therapeutics, but also helping to understand the underlying mechanisms of cancer development. However, the biological analysis needed in this regard is beyond the scope of this thesis.

Even though the strategy of analysis was developed and implemented on the background of WDLPS, it is applicable to any situation in which the structure of a target sequence is to be analyzed as to differences to a reference genome. Among such applications are, for instance, other cancer genomes that feature rearrangements contained inside the normal chromosomes.

Some improvements are conceivable, but had to be disregarded in the current setup on account of time considerations. In view of the fact that about 90% of the read pairs had no alignment on the contigs, the data utilization ratio is no doubt quite low, but not unusual on the background of the large number of short contigs. A higher coverage to be realized in the laboratory would reduce this number towards fewer and longer contigs, which in turn would make for a better utilization ratio. Another measure towards this objective would consist in accounting for breakpoints between contigs when mapping the reads. Specifically, by defining breakpoint regions of twice the read length for at least such contigs that are identified as adjacent to one another, a significant increase in data utilization could be realized. Moreover, low coverage contigs, which presently are filtered out through Velvet's automatic coverage-cutoff, could be accounted for.

Similarly, a considerable fraction of contigs could not be assigned to their origin on the human reference genome. The reason here is, that the heuristics postprocessing the BLAST alignments is not yet capable of solving more intricate cases.

Finally, it would appear worthwhile to extract information from Velvet's full de Bruin graph instead of analyzing the contigs file, only. Such additional information, however, is not easily accessible in a PC environment, and, hence, was reconstructed here not in whole, but in part, after all.

List of Figures

2.1 DNA structure and organisation 4
2.2 Shotgun sequencing 6
2.3 Coverage distribution 9
2.4 Error-correction 10
2.5 Sketch of the fragmentation of assemblies due to repeats . . . 10
2.6 Repeat-induced overcompression. 12
2.7 Schematic overlap graph of a partially repetitive source sequence 13
2.8 De Bruijn graph example. 14
2.9 Contigs and scaffolds 15

3.1 Common cytogenetic features of WDLPS 18
3.2 Conceptual sketch of neochromosome structure. 18
3.3 Overview of analysis strategy 20
3.4 Analysis of pairs. 22
3.5 Analysis of pairs - different contigs. 23
3.6 Analysis of pairs - same contig. 24
3.7 Contig-graph example 25
3.8 Contig-graph example 26

4.1 Distribution of read lengths. 30
4.2 Distribution of contig lengths. 30
4.3 Utilization of read pairs in read-contig-mapping. 31
4.4 Distribution of insert sizes. 32
4.5 A subgraph of the contig-graph. 34

4.6 Indentified fusion sites on the neochromosome. 35

List of Tables

2.1 Comparison of current sequencing technologies 7

3.1 Structure of Bowtie alignments 21
3.2 Structure of Blast alignments 26
3.3 Parameters of test data generation. 27

4.1 Identified fusion sites on the neochromosome. 33

List of Abbreviations

A	Adenine
BLAST	Basic Local Alignment Search Tool
bp	Base pairs
C	Cytosine
DNA	Deoxyribonucleic acid
FAP	Fragment assembly problem
G	Guanine
IUPAC	International Union of Pure and Applied Chemistry
NGS	Next-generation sequencing
OLC	overlap layout consensus
SCS	shortest common superstring problem
T	Thymine
TSP	Travelling Salesman Problem
WDLPS	Well-differentiated liposarcoma

Bibliography

454 Life Sciences (2010). http://www.454.com/.

Alba, E. and Dorronsoro, B. (2008). Bioinformatics: The DNA Fragment Assembly Problem. In Cellular Genetic Algorithms vol. 42, of Operations Research/Computer Science Interfaces Series pp. 203–210. Springer US.

Altschul, S., Gish, W., Miller, W., Myers, E. and Lipman, D. (1990). Basic local alignment search tool. Journal of molecular biology *215*, 403–410.

Ansorge, W. (2009). Next-generation DNA sequencing techniques. New biotechnology *25*, 195–203.

Applied Biosystems (2010). http://www.appliedbiosystems.com/.

Boveri, T. (1914). Zur frage der entstehung maligner tumoren. G. Fischer.

Campbell, P. J., Stephens, P. J., Pleasance, E. D., O'Meara, S., Li, H., Santarius, T., Stebbings, L. A., Leroy, C., Edkins, S., Hardy, C., Teague, J. W., Menzies, A., Goodhead, I., Turner, D. J., Clee, C. M., Quail, M. A., Cox, A., Brown, C., Durbin, R., Hurles, M. E., Edwards, P. A., Bignell, G. R., Stratton, M. R. and Futreal, P. A. (2008). Identification of somatically acquired rearrangements in cancer using genome-wide massively parallel paired-end sequencing. Nat Genet *40*, 722–729.

Chan, E. Y. (2009). Next-generation sequencing methods: impact of sequencing accuracy on SNP discovery. Methods Mol Biol *578*, 95–111.

Dohm, J. C., Lottaz, C., Borodina, T. and Himmelbauer, H. (2008). Substantial biases in ultra-short read data sets from high-throughput DNA sequencing. Nucleic Acids Res *36*.

Ellson, J., Gansner, E., Koutsofios, L., North, S. and Woodhull, G. (2002). Graphviz—open source graph drawing tools. In Graph Drawing pp. 594–597, Springer.

Garsed, D., Holloway, A. and Thomas, D. (2009). Cancer-associated neochromosomes: a novel mechanism of oncogenesis. BioEssays *31*, 1191–1200.

Huson, D. (2010). Lecture notes: Bioinformatics I. `http://www-ab.informatik.uni-tuebingen.de/teaching/ws09/bioinformatics-i`.

Kingsford, C., Schatz, M. C. and Pop, M. (2010). Assembly complexity of prokaryotic genomes using short reads. BMC Bioinformatics *11*, 21–21.

Krzywinski, M., Schein, J., Birol, İ., Connors, J., Gascoyne, R., Horsman, D., Jones, S. and Marra, M. (2009). Circos: an information aesthetic for comparative genomics. Genome research *19*, 1639.

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

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

MacLean, D., Jones, J. and Studholme, D. (2009). Application of'next-generation'sequencing technologies to microbial genetics. Nature Reviews Microbiology *7*, 287–296.

Mardis, E. (2008). The impact of next-generation sequencing technology on genetics. Trends in Genetics *24*, 133–141.

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

Medvedev, P., Georgiou, K., Myers, G. and Brudno, M. (2007). Computability of models for sequence assembly.

National Library of Medicine (2010). `http://www.nlm.nih.gov/`.

Pevzner, P., Tang, H. and Waterman, M. (2001). An Eulerian path approach to DNA fragment assembly. Proceedings of the National Academy of Sciences of the United States of America *98*, 9748.

Pop, M. (2004). Shotgun Sequence Assembly. vol. 60, of Advances in Computers pp. 193 – 248. Elsevier.

Pop, M. (2009). Genome assembly reborn: recent computational challenges. Briefings in bioinformatics *10*, 354.

Pop, M. and Salzberg, S. L. (2008). Bioinformatics challenges of new sequencing technology. Trends Genet *24*, 142–149.

Sandberg, A. (2004). Updates on the cytogenetics and molecular genetics of bone and soft tissue tumors: liposarcoma. Cancer genetics and cytogenetics *155*, 1–24.

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

Schwartz, R. A., Trovato, M. J. and Centurion, S. A. (2010). Liposarcoma. `http://emedicine.medscape.com/article/1102007-print`.

Solexa (2010). `http://www.solexa.com/`.

Staden, R. (1979). A strategy of DNA sequencing employing computer programs. Nucleic Acids Res *6*, 2601–2610.

Zerbino, D. and Birney, E. (2008). Velvet: algorithms for de novo short read assembly using de Bruijn graphs. Genome research *18*, 821.

Selbständigkeitserklärung

Hiermit erkläre ich, dass ich die vorliegende Arbeit selbständig und nur mit erlaubten Hilfsmitteln angefertigt habe.

Magdeburg, den 15. September 2010

Kristian Löwe