Chapter 3

New approaches to assemble *var* genes from short reads

3.1 Introduction

As described in Chapter 2, *de novo* assembly of *var* genes is challenging mainly due to high A+T content, shared sequence blocks and uneven read coverage. The polymorphic and mosaic nature of *var* genes adds further complexity to the problem of short read assembly. Despite the high sequence polymorphism in *var* genes, the presence of highly conserved short motifs was reported (Bull et al., 2007; Rask et al., 2010). For example the motif LARSFADIG is located on the DBLα region and found in nearly all *var* genes. Although the frequent recombination events that shuffle sequence blocks play an important role in the evolution of *var* genes (Frank et al., 2008; Kraemer et al., 2007; Rask et al., 2010; Taylor et al., 2000b), they also make use of existing short read assembly approaches inadequate. This chapter focuses on two major challenges in assembling *var* genes: identifying reads that belong to *var* genes and performing a targeted assembly on the collated reads.

Currently, there are no established methods for a targeted assembly of a gene, a group of genes or gene families. One approach could be to run a whole genome assembly followed by identifying contigs that resemble the genes of interest. However, in chapter 2, I demonstrated that this approach had

serious limitations; whole genome *de novo* assemblies of *P. falciparum* are highly fragmented, with the potential to collapse highly similar regions. Due to the high A+T content and low complexity, the uniqueness of most regions is also extremely low, resulting in false joins during the assembly or scaffolding stages. The problem becomes complicated while dealing with sequences from clinical isolates due to the presence of multiple infections, poor data quality and uneven read coverage. Although the blood-stage parasites are haploid, presence of multiple infections results in multiple haplotypes in individual patient samples. As a result, the complexity of the assembly graph increases for example due to bubbles and chimeric connections leading to a highly fragmented assembly. A new approach is thus required to rapidly identify short reads that come from the *var* gene family in order to perform a targeted assembly of short reads.

The problem of genome assembly could be illustrated with the analogy of solving a jigsaw puzzle where short reads are the pieces of the puzzle that need to be organized in order to reconstruct a complete genome or region of the genome. Assembling the *var* gene family could therefore be seen as solving $\sim$60 related puzzles from a mixture of millions of pieces including pieces from non-*var* puzzles (*var* genes are $\sim$0.2% of the genome). Thus a plausible approach would be to first identify the pieces that belong to the $\sim$60 puzzles as a whole and then to find a way to solve each puzzle from the mix.

This chapter aims to develop an alternative assembly approach for *var* genes that:

- addresses limitations of existing assembly approaches, specifically the two challenges described in the previous section:
  - rapidly identifying reads that belong to the *var* gene family, and
  - reconstructing members of the family
- is scalable to thousands of parasite isolates
- builds on existing methods already developed in our laboratory when applicable.

3.2 Methods: Proposed assembly approach

The assembly approach proposed in this chapter has two components. Firstly, identical regions of the *var* mosaic blocks were identified as shared motifs and used to assist with identification of reads that belong to the *var* genes. Secondly, an iterative assembly approach, that takes advantage of de Bruijn graph-based and overlap-layout-consensus assembly approaches, was introduced. The three main stages: pre-processing, generating seed contigs and iterative scaffolding/extension (Figure 3.1A-C) and six processes (1-6) comprise the new approach developed to address the assembly problem of *var* genes.

3.2.1 Preprocessing sequence data

3.2.1.1 BAM to preFasta

The purpose of this stage (Figure 3.1A. 1) is to reduce the dataset by excluding sequence reads that do not come from defined "regions of interest". In addition to minimizing the physical storage requirements (i.e. disk space), this step will eventually improve assembly quality by reducing data complexity as a result of the removal of reads from unwanted regions.

Input file 1: BAM files
Initially, raw FASTQ files of the samples will be stored in the BAM (Li et al., 2009a) file format. BAM files could be obtained as a result of an alignment process to a reference genome or alternatively, in the absence of a reference genome, BAM files will only store raw reads. Although the methods developed in this chapter are applicable in the absence of a reference genome, here, availability of a reference is assumed.

Input file 2: A file with regions of interest
A tab delimited file representing regions of interest is required to identify regions of the genome that will be included in the raw data. The format is shown below:

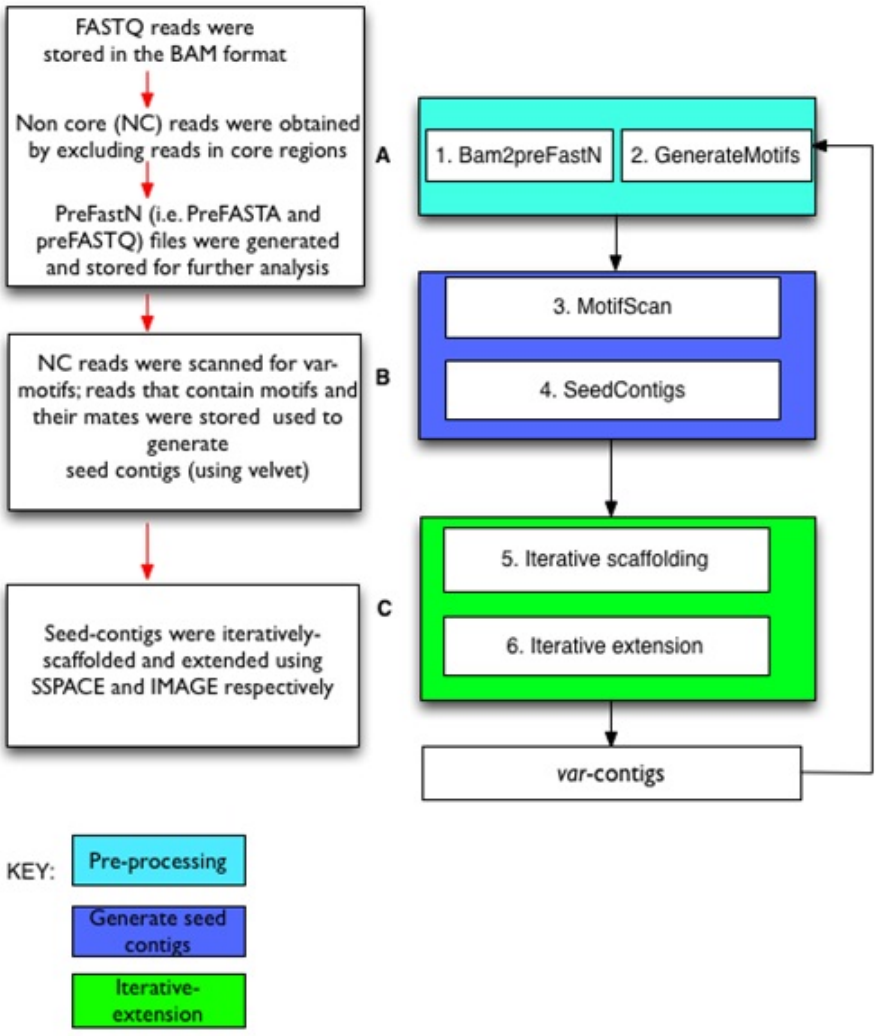

Figure 3.1: A work flow diagram of the iterative assembly approach developed to assemble *var* genes. The three stages and six processes of the iterative assembly approach developed to assemble *var* genes are shown. Conserved regions of *var* genes that were found in a minimum of two culture adapted samples were identified as initial motifs. Similarly, motifs for consecutive iterations were generated by finding conserved regions that were common in *var*-contigs of two or more samples. Core-regions were defined as central regions of the genome except the central *var*-clusters. Non-core reads were thus defined as reads that align to non-core regions of the genome and reads that did not align to the reference genome. See text for further details.

Ch1_Telo.01	MAL1	1	91653
Ch1_Telo.02	MAL1	565425	643292
Ch2_Telo.01	MAL2	1	67545
Ch2_Telo.02	MAL2	860464	947102

where columns 1 to 4 represent regionID, Chromosome, Start and End positions, respectively.

Defining regions of interest

First, a working definition of Subtelomeric regions was obtained by taking regions of the genome from chromosome-ends to the last subtelomeric copy of a *var*, *rif*, or *stevor* gene family. The genome was then divided into subtelomeric regions, central *var* gene clusters and the remaining core regions of the genome.

Regions of interest for assembly of the *var* gene family were defined by combining subtelomeric regions and central *var* clusters resulting in ∼2 Mb (10% of the genome) of non-core regions. In addition, reads that did not align to the genome were also included as they contain potentially novel sequences and highly polymorphic regions. The reads from non-core regions of the genome (i.e. subtelomeric reads, reads in central *var* regions and unmapped reads) are required during the iterative assembly process. It was therefore necessary to define a new file format that allowed efficient data storage.

Defining the preFasta and preFastq file formats

The FASTQ file format used to represent short read sequences from the Illumina platforms is shown below:

```
@Root_ID/1
Forward_read
+
Forward_quality
@Root_ID/2
Reverse_read
+
```

```
Reverse_quality
```

The FASTA file format has no quality values and is defined as:

```
>Root_ID/1
Forward_read
>Root_ID/2
Reverse_read
```

A file format suitable for storing large number samples was adopted in this thesis to store reads from non-core regions of the genome. In order to increase efficiency in data storage, *preFasta* and *preFastq* files contain the minimal sequence information required for subsequent storage and iterative assembly stages. The quality information in FASTQ files is still not used by assembly tools and could be discarded in the preFasta files. In addition, the standard FASTQ and FASTA indicators such as "@", ">", "/1", "/2" could also be discarded during storage and generated in real time while processing the data.

The preFasta file was therefore defined with three columns containing the minimum required information for a FASTA file:

```
Root_ID Forward_read Reverse_read
```

Similarily, a preFastq fill will contain additional two columns to include quality values for the forward and reverse reads:

```
Root_ID Forward_read Reverse_read Forward_Q Reverse_Q
```

This format ensured a significant saving in storage compared to the original FASTA and FASTQ files, which required four and eight lines respectively compared to the single line representation of preFasta and preFastq files. Finally, preFasta/preFastq files were compressed using the Unix command "gzip -9" to ensure additional savings in storage space.

3.2.1.2 Generate motifs

The next step in the pre-processing stage of the assembly pipeline was to generate a list of conserved or shared sequence elements, also known as motifs (Figure 3.1A.2). Initially, a BLAST search was done on a union file of *var* genes from three laboratory-adapted samples (i.e. all against all). However, this approach was not scalable with respect to the number of genes and size of input files.

A *k-mer* based hashing approach was developed to count the frequency of exact matches of sequences with a length of 'k' (k=10 for amino acids and k=30 for nucleotides). Motifs were then grouped according to the number of samples (or genomes) they came from. A 'shared motif' was defined as a motif found in a minimum of two samples. In order to avoid motifs from low-complexity regions that could potentially cause spurious matches, shared motifs that were also present in core regions of the genome were discarded. The information on motifs and motif sharing was represented in the following format:

```
ID  #Populations  #Samples  #Genes  GeneInfo SampleInfo PopInfo
```

GeneInfo, SampleInfo and PopInfo contain comma-separated lines listing the names and frequency of genes, samples and populations that share the motif.

The process of identifying motifs was repeated after each iteration of assembly and extension of contigs (described in sections 3.2.2 and 3.2.3). Addition of new motifs was expected to increase the motif database, enhancing the potential of capturing new sequences and novel variants of the *var* gene family.

3.2.2 Generating seed contigs

3.2.2.1 Scanning raw reads for motifs

Once a set of shared motifs (nucleotide or amino acid) were identified from members of the *var* gene family, raw reads were examined for the presence of the motifs. An exact match of a motif to either the forward or reverse read of a read pair was required prior to storing both reads for the initial stages of generating seed contigs (see next section).

Initially, identifying reads that contained a motif was done via a BLAST search using motifs and raw reads as database and query respectively. However, it became obvious that a BLAST-based scanning approach was too slow and not scalable as the number of motifs increased from a few thousands to millions.

A motifs scanning tool, *MotifScan*, that rapidly identified exact matches was developed to address this limitation. *MotifScan* was written in C++ and could be used to quickly scan for the presence of motifs in a large number of FASTA or FASTQ reads. Furthermore, motifs could be in amino acid or nucleotide formats. For amino acid motifs, reads were translated in all the six reading frames during the scanning process. *MotifScan* slides by one position over the length of each read or its amino acid translation until a match to the motif database is found. It is important to note that scanning until the sixth frame is the worst case scenario as a match to the motif database could be identified earlier. The result of this step was a FASTA/FASTQ formatted list of reads and their mates where the forward, reverse or both reads contained a motif. Including a read where its mate has a motif is an important aspect of the motif-scanning process as it provided additional information that could be used to extend and join seed regions in subsequent stages of the assembly process.

3.2.2.2 Generating seed contigs

Using the reads obtained from the motif scanning process, seed contigs were produced that could then be extended in the next steps. Velvet (Zerbino and Birney, 2008) was used to generate initial contigs as it was shown to have a better assembly quality when compared with other short read assemblers (Chapter 2). The scaffolding option was not used to minimise the risk of potential false joins. As described in chapter 2, choice of a *k-mer* size is an important parameter that needs optimisation for each iteration of assembly. Although Velvet was used in this thesis, the assembly approach described here is able to use a different assembler to generate seed contigs.

3.2.3 Iterative scaffolding and extension

The final stages of the pipeline (Figure 3.1C) involved sub-iterations of joining and extension of seed contigs.

3.2.3.1 Scaffolding

Seed contigs generated in the previous step were expected to be highly fragmented as the initial set of reads represented a very small fraction of the total. The scaffolding step was therefore important to join contigs that had strong read-pair support. At the beginning of this project, there was no stand-alone scaffolder that could be used independently. Short read assembly tools have built-in scaffolding modules that have a very limited flexibility. In order to address these limitations, a scaffolding tool was developed that took advantage of read pair information from standard sequencing libraries. The distribution of fragment sizes was used to estimate gap sizes and join contigs into scaffolds where there was sufficient and unambiguous evidence. SSPACE (Boetzer et al., 2011), a scaffolding software with similar principles became available during the course of development. After testing the performance, I decided to optimize SSPACE instead of continuing working on our scaffolder.

3.2.3.2 Iterative extension

Contigs and scaffolds generated in the previous steps account for less than the total nucleotide content of the gene family. In addition, scaffolds are expected to have gaps with unknown bases (Ns) that need to be filled during an iterative extension step described here. This step is intended to close gaps in scaffolds and extend contig-ends primarily using read pair information (Figure 3.1C.6).

Reads obtained from non-core regions of the genome were aligned to seed contigs generated in the previous steps and used as raw reads to initiate the extension process. By aligning reads to the seed contigs, it was possible to iteratively walk out of seed regions. The process involved a number of sub-iterations of mapping reads to seed contigs, identifying reads that align to contig-ends and performing a local assembly using Velvet. These principles

were implemented in IMAGE (Tsai et al., 2010), a tool developed in our laboratory by Jason Tsai for the purpose of closing gaps in draft assemblies. It was decided to optimize IMAGE for the iterative extension step of the assembly process. IMAGE begins by aligning short reads to contigs given a list of contigs, scaffolding information and raw FASTQ reads. After a number of optimisation steps on IMAGE's various options, the best extension and gap-closure results were obtained from choosing optimal parameters for the number of iterations, *k-mer* values used for local assembly and the minimum alignment score of reads when mapped to seed-contigs. A *k-mer* value of 41 was used over 5 to 7 iterations to obtain good quality gap closure and contig extension. Decisions to stop at 5 iterations or continue to 7 were made based on the number of gaps closed and the improvement in N50 contig size. A minimum alignment score of 70 for 76 bp reads (i.e. use reads that aligned with a score of above 70) was found to be critical as it minimised the effect of erroneous joins between contigs due to poor quality matches.

3.2.4 Evaluating the assembly approach

3.2.4.1 Testing on culture-adapted samples

Sequence data

In order to evaluate the performance and accuracy of the new assembly approach, a total of four laboratory adapted samples were used. DNA for sequencing of the reference isolate 3D7 and the IT sample was obtained from Prof. Chris Newbold's laboratory in Oxford (Chapter 2, section 2.2). Sequences for DD2 and HB3 were obtained from Prof. Dominic Kwiatkowski's laboratory at the Sanger Institute. The PCR-free library preparation protocol described in chapter 2 was used. Raw reads of the four samples were aligned to the 3D7 reference genome version 2.1.4 using SMALT `http://www.sanger.ac.uk/resources/software/smalt/` (A python script written by Martin Hunt in our laboratory was used to align on the following SMALT parameters: *-i 500 -r 10 -x -k 13 -s 6*). Reads are referred to be mapped in proper-pairs if both the forward and reverse reads align to the reference genome facing each other within the expected fragment size range (200-300 bp).

Var *genes*

Var genes for the 3D7 and IT genomes were obtained from GeneDB (`http://www.genedb.org/`). As described in the previous chapter, 3D7 is the reference isolate with a complete genome whereas the other genomes are still highly fragmented with a limited coverage of the *var* repertoire. Supercontigs of HB3 and DD2 genomes were obtained from the Broad Institute (`http://www.broadinstitute.org/`). Sequences annotated as *VAR*/PfEMP1 were identified resulting in 47 and 25 genes for HB3 and DD2 respectively.

Initial Motifs

Initially, *var* genes from 3D7, IT and HB3 genomes were used to generate motifs using Pmatch. However, due to the minimum length requirement of 14 aa, we decided to develop a *k-mer*-basd hashing approach to generate motifs. For the assembly of culture-adapted samples, initial motifs were generated from *var* genes of HB3 and DD2 genomes for two reasons. First, these samples have incomplete *var* repertoire and motifs generated from them would represent a minimal set of starting motifs. It is thus a good indicator of the approach's success in clinical samples. Second, initiating the process on motifs obtained from the other genomes will provide a means to evaluate the completeness and accuracy of the *var* repertoire produced for the 3D7 genome.

Iterations

The process was run for a total of 10 iterations. New motifs were generated at the end of each iteration and used as input for the next iteration.

Evaluating assembly

Assembly quality was evaluated by four commonly used measures: N50 contig size, sum of contigs and largest contig sizes. The completeness of the *var* repertoire was estimated by counting the number of contigs with the DBLα domain. In order to test the accuracy of the contigs generated by the process, the 3D7 genome was used as a reference. All contigs from the 3D7 assembly were aligned against *var* genes of the 3D7 genome.

3.2.4.2 Testing on clinical samples

Sequence data

A total of 50 samples were randomly selected from the Plasmodium Genome Variation (PGV) project at the Sanger Institute. Clinical samples were initially collected and sequenced by Prof. Dominic Kwiatkowski's lab at the Sanger Institute. The samples were randomly selected from 10 different countries representing West Africa, East Africa and South East Asia.

Initial motifs and iterations

Initial motifs for the assembly of 50 clinical samples were generated from *var* genes of 3D7, HB3 and IT genomes. *Var* genes for the three samples were obtained as described in the previous section. Amino acid motifs of length 10 aa were generated and checked for uniqueness. Motifs shared by a minimum of two samples and that were unique to non-core regions and the flanking upstream and downstream regions of 2 kb of the 3D7 genome were selected to initiate the process. The performance of the iterative assembly was enhanced by generating motifs from a six frame translation of contigs that contain the DBLα tag. In order to determine the number of iterations required to gather an optimal number of motifs (i.e. the iteration where the number of shared motifs reaches a saturation), the assembly was run for a total of 20 iterations. Shared motifs obtained at the end of each iteration were checked for quality and used as inputs for the next iteration.

Assembly statistics and quality check

Contigs that contain the DBLα domain were obtained from the 20^{th} iteration and assessed on how close they are from the expected assembly statistics (based on *var* genes of the 3D7 genome). In addition, the size distribution of open reading frames was examined to evaluate the accuracy of the assembly.

3.2.4.3 Additional evaluation using samples from the Illumina HiSeq platform

In order to further evaluate the assembly process, five clinical samples from the latest runs of the Illumina HiSeq platform were selected. These runs have a higher yield and longer read lengths (100 bp, paired end reads) compared to the previous samples used to test the assembly process.

Comparing with* de novo *assembly

Initially, the five samples were assembled using motifs generated at the end of the 20^{th} iteration. After running the iterative assembly process for three iterations, assembly quality of contigs that contained the DBLα tag were examined and compared with scaffolds of a *de novo* assembly (made by Velvet and obtained from Thomas Otto in our laboratory).

Mixed assembly

In order to test the performance of the new assembly approach in parasite samples that have multiple infections, raw reads from four of the five samples were selected. These samples were shown to have a single infection based on the number of contigs that contain the DBLα domain (i.e. expecting 60 genes per genome). Raw reads from non-core regions of the genome were first individually assembled for each sample (*k=71, cov-cutoff=auto*). The reads were then mixed and assembled using identical assembly parameters as the individual assembly. Open reading frames (ORFs) of contigs that contain the DBLα were obtained by translating to all six frames and choosing the frame with DBLα. The contigs from the two sets of assemblies were compared at the protein level using BLAST (*blastp - F F -m 8*). This provided a better assessment than a nucleotide based comparison as regions of extremely low G+C content such as introns and intragenic regions were excluded.

3.3 Results

3.3.1 Defining regions of interest

Regions of interest were defined using the reference genome 3D7 as described in section 3.2.1.1. A simple definition of subtelomeric regions was sought for the purpose of this thesis resulting in a total of ∼2Mb (10% of the genome) from the 28 subtelomeres. An example of the working definition of subtelomeric regions on chromosome 1 is shown in Figure 3.2. Analysis of the size distribution of subtelomeric regions in the *P. falciparum* genome using this working definition revealed that chromosomes 4 and 7 had the longest subtelomeric regions (Figure 3.3).

Although the working definition of subtelomeric regions adopted for this thesis may be different from that of the original genome annotation (defined using synteny with closely related species), it was possible to capture highly polymorphic regions that are currently being excluded in studies that rely on a unique alignment of short reads to call single nucleotide polymorphisms and copy number variations.

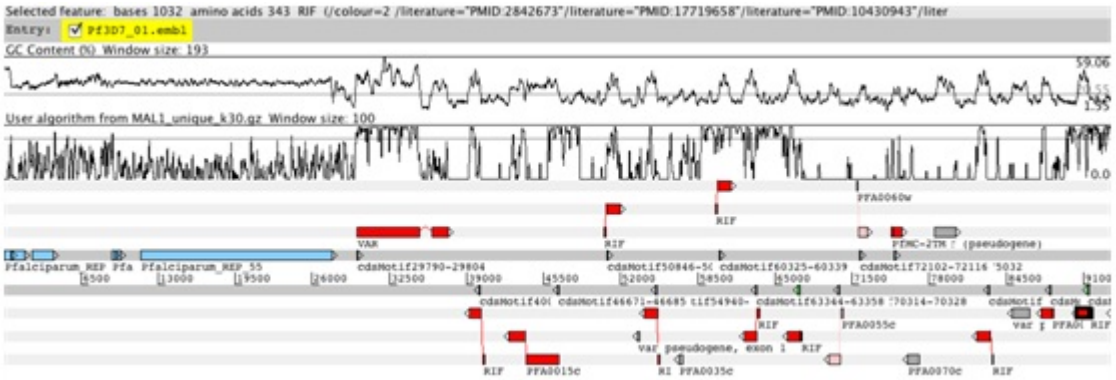

Figure 3.2: A working definition of subtelomeric regions adopted for this thesis: Regions of the genome from the end of each chromosome to the last subtelomeric copy of a *var*, *rif* or *stevor* family were identified as subtelomeric. The red blocks represent coding genes, grey boxes represent pseudo-genes, cyan blocks at the left-end represent repeats. The two plots on the top panel show the G+C content and *k-mer*-based uniqueness plots.

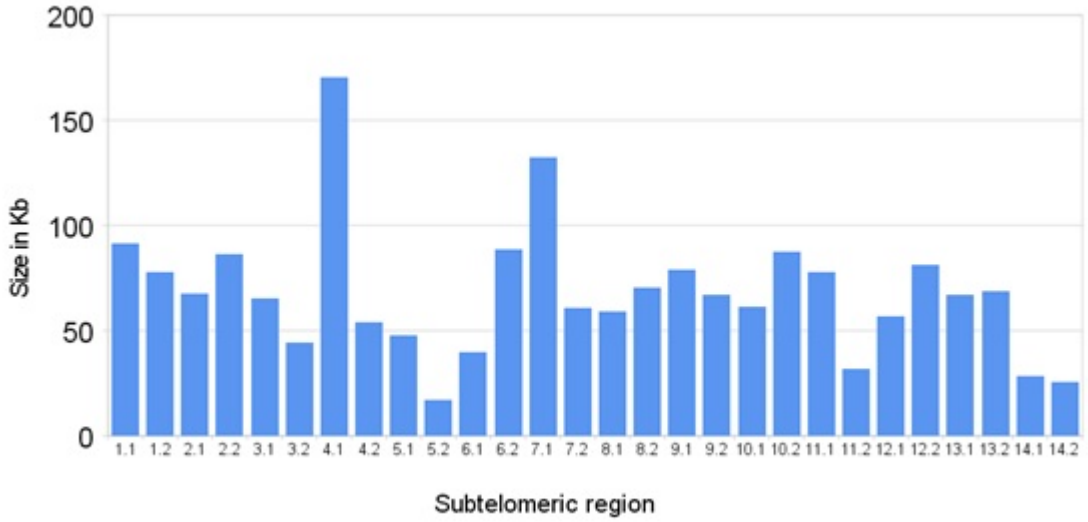

Figure 3.3: Size distribution of subtelomeric regions in the reference genome 3D7 by Chromosome. Sizes range from 17.5 to 170 kb covering ∼10% of the genome.

3.3.2 Evaluating the new assembly approach using culture-adapted samples

3.3.2.1 Sequence data

A summary of raw reads for the four culture-adapted samples are shown in Table 3.1. The number and proportion of non-core reads varied in the four samples from ∼9 to 34% depending on the number of reads aligned to the genome and reads that did not align (eg. due to poor read quality).

The HB3 genome had a significantly higher number of non-core reads due to the higher number of reads that did not align to the 3D7 genome (∼33%). This could be due to a decrease in base quality of the second read after the $\sim 50^{th}$ cycle (Appendix B, Figure B-2).

	Total read pairs	%All reads aligned to 3D7 (Aligned in proper pair)	Non-core read pairs (%total)
3D7	12,488,019	96(93)	3,453,588(28%)
DD2	20,650,235	91(84)	2,327,890 (11%)
HB3	18,501,446	67(73)	6,369,447(34%)
IT	27,013,569	97(86)	2,477,392 (9%)

Table 3.1: A summary of the four culture-adapted samples and non-core reads used to evaluate the iterative assembly approach. Reads that aligned in the correct orientation (facing each other) and within the expected insert size range are defined as 'aligned in proper pair'

3.3.2.2 Motif generation and iterative assembly

A total of 17,719 motifs (10 aa overlapping *k-mer*s) were found to be shared between *var* genes of HB3 and DD2 genomes. After excluding motifs that were also found in the core regions of the 3D7 genome, a total of 7,353 motifs remained to initiate the iterative assembly process. As the number of iterations increased, the number of motifs shared by a minimum of two of the four samples also increased (Figure 3.4). However, the rate of increase in new motifs declined after the $\sim 4^{th}$ iteration due to a potential saturation in the motif space. The highest improvement was found in the first two iterations where the number of motifs increased from the initial ∼7,000 to ∼180,000.

Assembly quality improved with more iterations (Figure 3.5). The most notable improvement was on the 2^{nd} iteration of the process. Overall, while the sum of contigs as well as N50 and largest contigs sizes increased with subsequent iterations, the number of contigs decreased indicating an improvement in assembly quality.

Although 3D7 had the best N50 contig size (∼5 kb), it also had the least number of contigs with DBLα and least value in sum of contigs compared to the other three samples. The N50 contig size is affected by the number of contigs available and thus may give a wrong impression of quality if not taken together with other measures as described here. The sum of contigs ranged from under 300 kb for 3D7 to ∼500 kb in the IT assembly. The number of contigs was comparable between samples during the first iteration (∼150 contigs per sample). However, in subsequent iterations the contig count for 3D7 stayed

below 100 while the other samples generated ∼300 contigs. This variation in the number of contigs is reflected in the N50 contig sizes as the highest N50 values for 3D7 correspond with the fewest contigs. The efficiency of the iterative extension process was also shown by the sizes of the largest contigs which increased from ∼5 kb to ∼12 kb during the course of the iterations. The number of contigs that contained the DBLα tag also showed improvement from the second iteration and converged to ∼40 to 50.

In summary, the iterative assembly generated a substantially higher number of *var*-contigs (i.e. contigs with the DBLα tag) compared to the original number of genes found in the HB3 and DD2 genomes used to initiate the motif generation process. It was possible to recover up to ∼80% of the expected *var* repertoire in the test samples by starting from a very limited set of initial motifs.

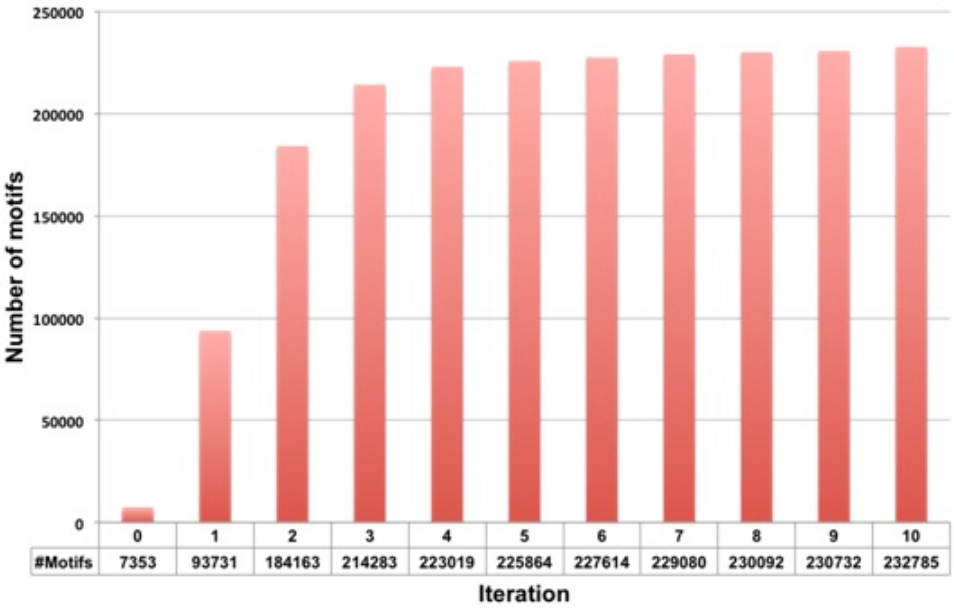

Figure 3.4: The cumulative number of shared motifs for 10 iterations of the four lab adapted samples 3D7, IT, HB3 and DD2. Initial motifs were obtained from HB3 and DD2 in order to test the assembly approach with a limited number of starting motifs.

Evaluating var contigs of the 3D7 genome

Optimal assembly results for 3D7 were obtained at the 5^{th} iteration (Table 3.2).

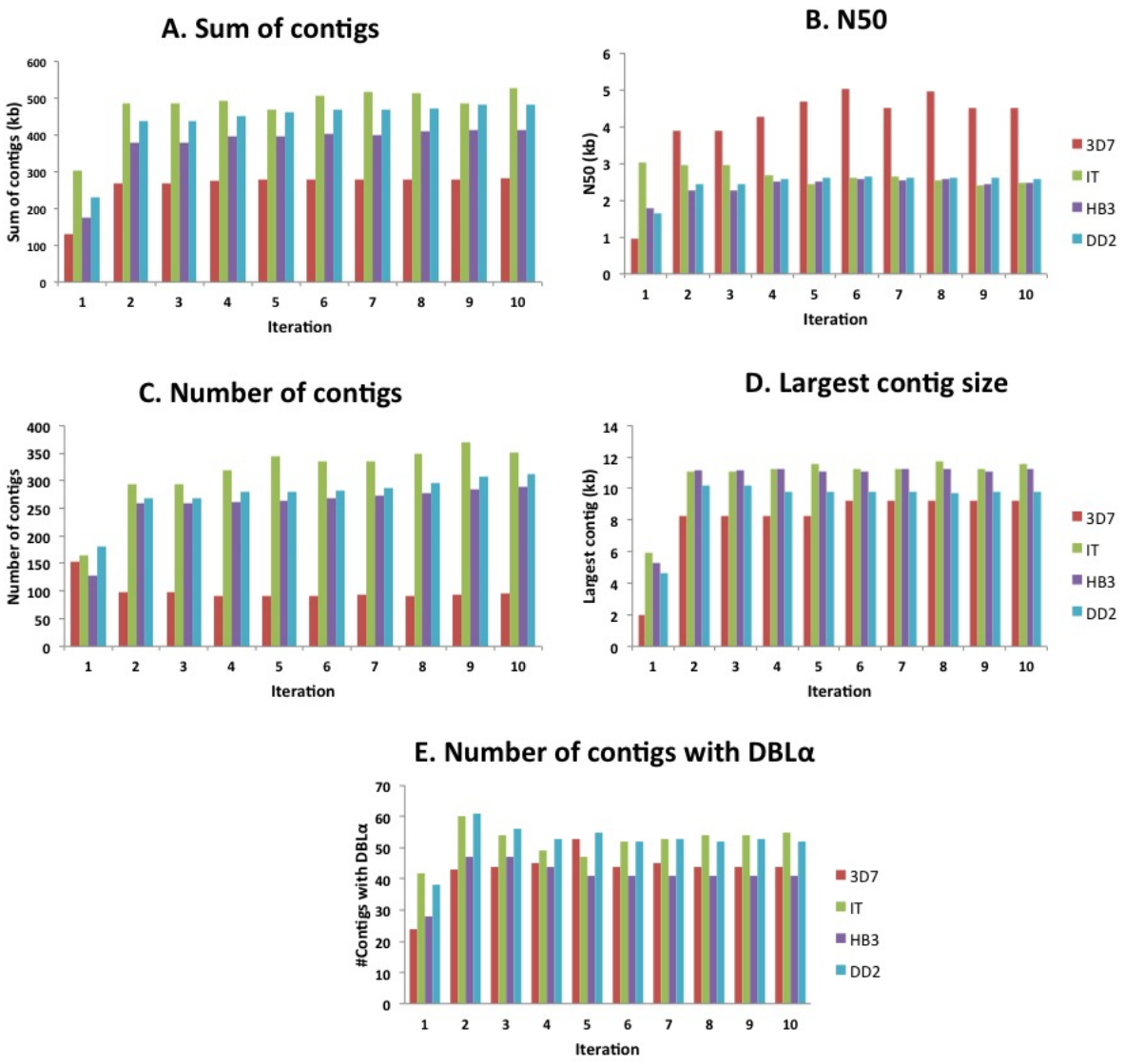

Figure 3.5: Assembly statistics of the four lab adapted samples for 10 iterations. Five assembly measures Sum of contigs, N50, Number of all contigs, largest contig and number of *var*-contigs were used to asses the assembly quality with an increase in iteration. A). Sum of contigs was used to measure the coverage of the *var* repertoire. The N50 contig size (B), the number of contigs (C) and the largest contig size (D) measure assembly contiguity. Completeness of the *var* repertoire is measured by counting contigs that contain the DBLα tag (E). The 3D7 assembly had the highest N50 and fewer contigs, but also had the least contig coverage as shown in A.

Comparing contigs generated from the iterative assembly to the 3D7 genome provided a measure of completeness and accuracy. Of the 46 contigs that contained the DBLα domain, only one misassembly was detected due to a chimeric connection between two genes of the central *var* cluster. Although it does not contain the DBLα domain, var2CSA was assembled in one contig producing a full length (intact) gene. The completeness of the repertoire in the 3D7 genome was therefore estimated to be ∼75% (i.e. expecting 61 *var* genes in the genome). Coverage of the *var* repertoire in 3D7 was evaluated by aligning all 93 contigs to the genome. Inspection of comparisons performed using BLAST and Abacas revealed that 46 of the 61 *var* genes in 3D7 were covered by contigs (Table 3.3). A total of 21 genes were partially covered (minimum coverage of 50% at 99% identity) and the remaining 25 genes were fully covered by one or more contigs. The majority of the genes were fully or partially covered by single contigs (intact).

	All contigs	**Contigs with DBLα**
Sum	277761	196368
N50	5039	5540
Num.	93	45(1*)
Largest	9223	9223

*mis-assembeled contigs

Table 3.2: Iterative assembly results for *var* genes of the 3D7 genome.

	Fully covered	**Partially covered**
Exon1 only	1	7
Exon1(+ Ups)	2	3
Exon1(+ Intron)	7	6
Exon1(+Ups +Intron)	15	5
Total	**25**	**21**
Intact	21	16
Fragmented	4	5

Table 3.3: Coverage of *var* genes in the 3D7 genome. A total of 46 (of the expected 61) genes were covered by one or more contigs.

The iterative assembly approach was able to recover ∼75% of *var* genes in

the 3D7 genome. As described earlier, the initial motifs were obtained from HB3 and DD2. The results were thus very promising as they illustrate the potential of this approach in reconstructing a large proportion of the repertoire in unrelated clinical samples. The missing genes are expected to be due to insufficient seed motifs as the assembly was performed using motifs generated from HB3 and DD2. The single misassembled contig was a result of chimeric join between two genes of the central cluster. These genes share identical regions of larger than 1 kb and could not be resolved using a standard library (200-300 bp). In addition, the smaller fragment sizes in the 3D7 library (quartiles: 143,163,188) may also contribute to poor quality assembly.

3.3.3 Evaluating new assembly approach using clinical samples

The iterative assembly approach was further evaluated using 50 clinical samples from 10 countries (Table 3.4). Samples were chosen from standard PCR-free libraries (insert size 200-300 bp) with a read length of 76 bp.

The three laboratory clones 3D7, HB3 and IT were used to generate initial motifs. A total of 8,766 motifs were shared by at least two of the three samples and also passed quality control steps. The number of motifs increased with each iteration as observed in the lab-adapted samples. However, the rate of increase in motif acquisition was slower after the 10^{th} iteration (Figure 3.6). Each iteration involved sub-iterations of scaffolding and extension that helped improve the quality of assembly. Assembly results of the 50 samples are summarised in Figure 3.7. At the end of the 20^{th} iteration, the average number of contigs that contain the DBLα (n=2,793) was close to the expected value of $\sim$3,000 (i.e. expecting $\sim$60 per genome). In addition, the N50 contig length (6.4 kb) and the largest contig ($\sim$14 kb) sizes were also within the expected range of values for *var* genes with the DBLα tag in the 3D7 genome (sum=428 kb N50=7.7 kb; Number of contigs =54; Largest contig=12.5 kb). Box plots showing the distribution of 2,769 *var*-contigs within the 6 groups (Figure 3.7) show a similar distribution with the 3D7 and other clinical samples studies by Bull and colleagues (Bull et al., 2007).

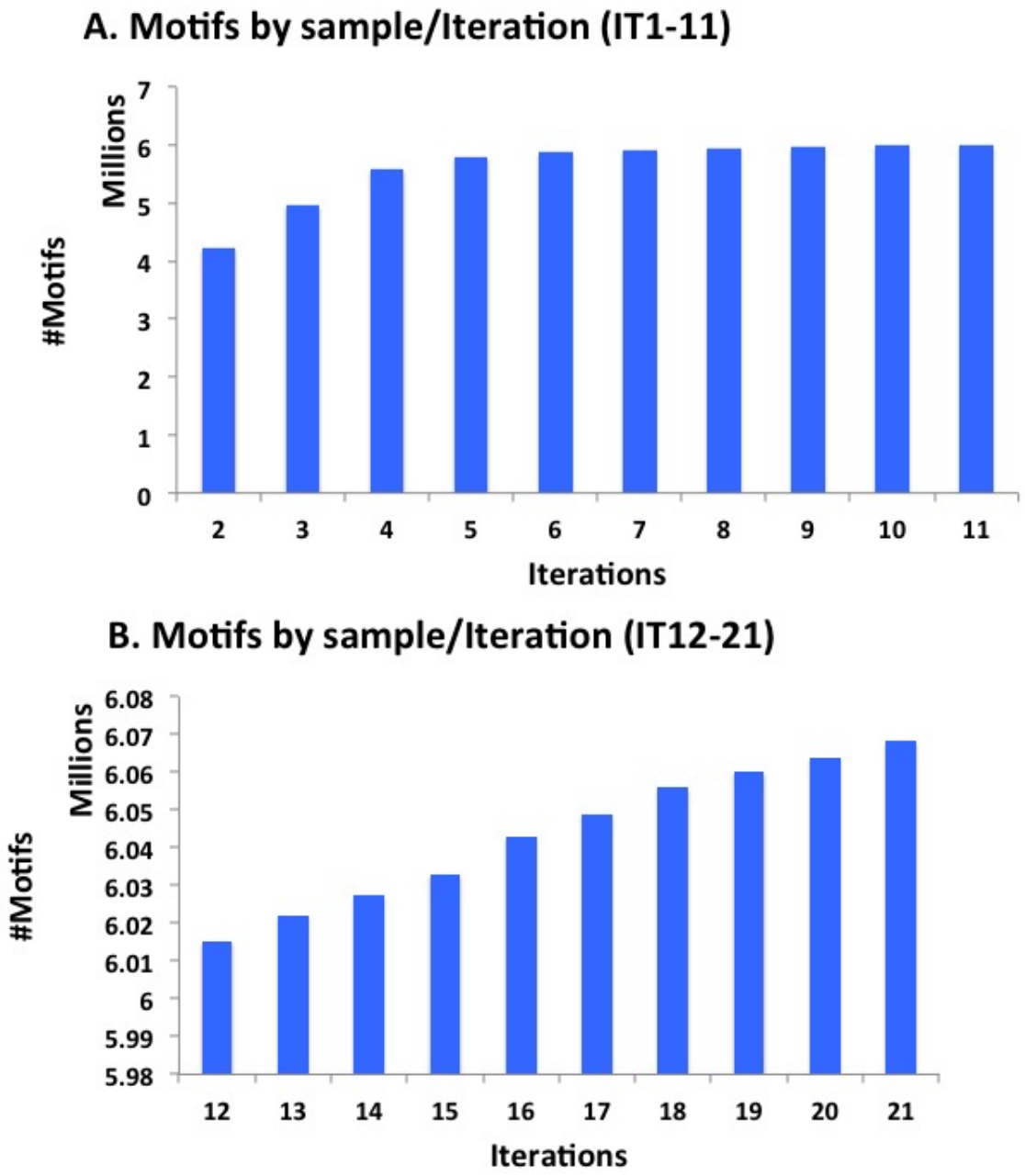

Figure 3.6: Number of motifs shared by at least two samples (i.e. *var*-contigs from two different samples) per iteration for 50 clinical samples. **A)**. The increase in number of shared motifs for the first 11 iterations is shown separately in A. The rate of increase in shared motifs was higher for the first ∼5 iterations. **B)**. The number of shared motifs continued to increase at a slower rate after the 12^{th} iteration as shown by the scale of the y-axis.

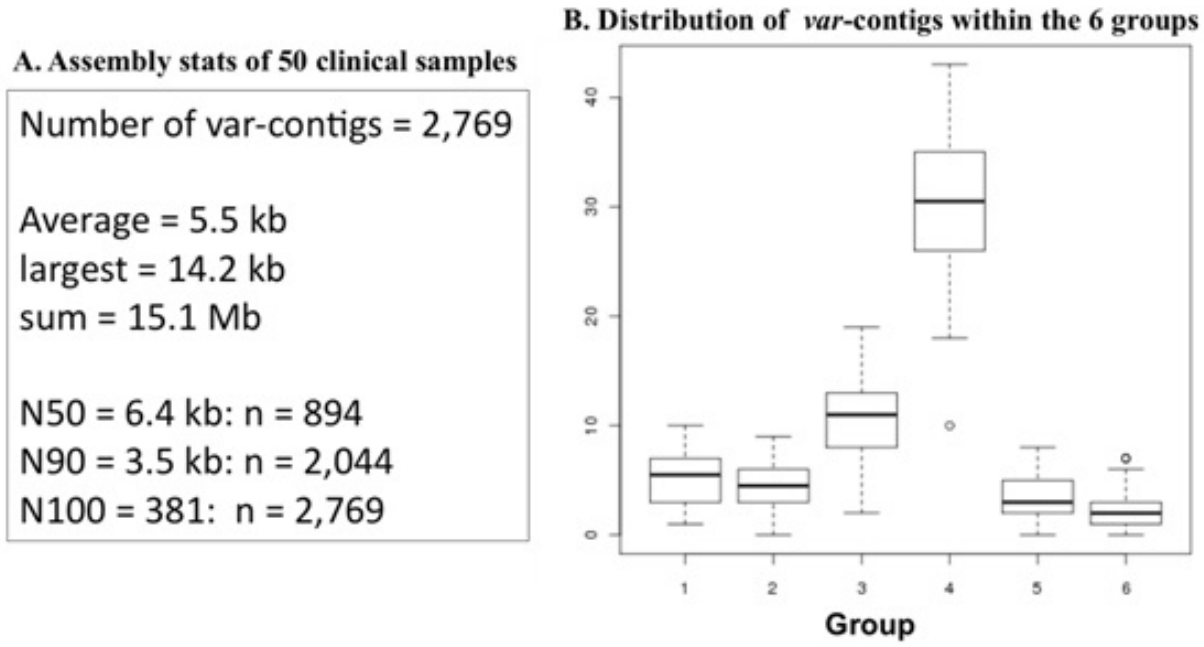

Figure 3.7: Assembly statistics of 50 clinical samples. **A)**. A total of 2,769 contigs had the DBLα tag representing $\sim$92% of the expected *var* repertoire. The number of contigs represented by the N assembly measures were also shown. For example N50 of 6.4 kb; n=894 indicates that a total of 894 contigs are above 6.4 kb in size. The sum of these contigs is equivalent to $\sim$7.5 Mb (i.e. half of the total sum of contigs). **B)**. *Var*-contigs were grouped into one of the six groups using the 'Cys-POLV' grouping method of Bull and colleagues (2007) (Bull et al., 2007).

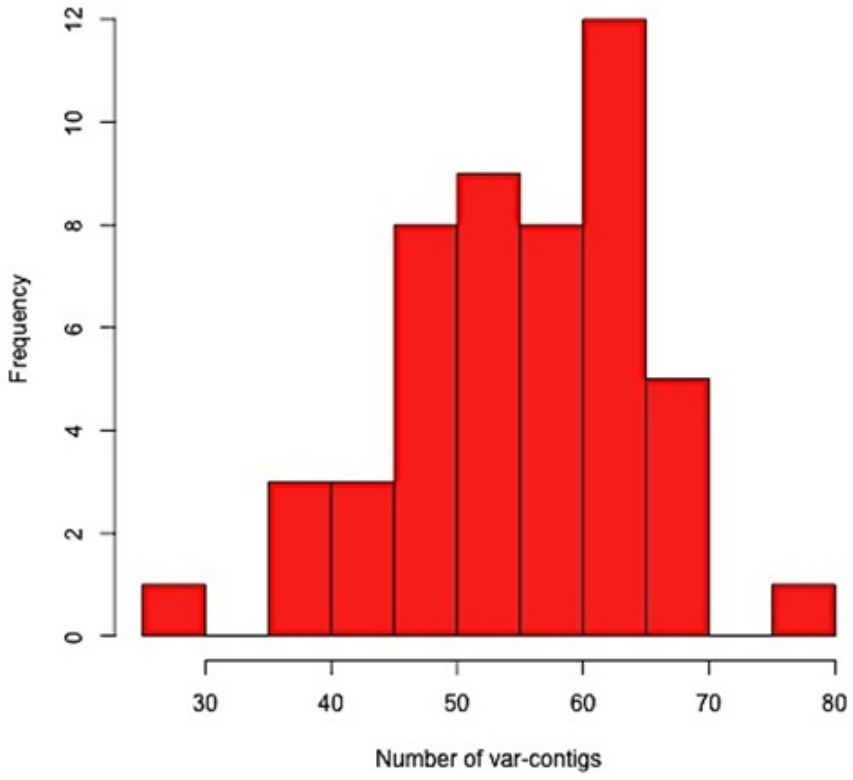

Figure 3.8: A histogram of the number of contigs with DBLα (*var*-contigs) per sample for the 50 clinical isolates. Two samples (PP0011 from Peru and PM0096 from Mali) were found on either end of the distribution with 26 and 78 *var*-contigs respectively.

Sample Origin	Sample IDs				
Gambia	PA0012	PA0032	PA0066	PA0081	PA0091
Kenya	PC0057	PC0070	PC0075	PC0080	PC0083
Thailand	PD0126	PD0127	PD0133	PD0134	PD0138
Ghana	PF0211	PF0231	PF0263	PF0288	PF0290
Cambodia	PH0142	PH0145	PH0380	PH0479	PH0483
Mali	PM0048	PM0090	PM0096	PM0132	PM0162
PNG	PN0027	PN0054	PN0056	PN0057	PN0059
Peru	PP0005	PP0006	PP0010	PP0011	PP0012
Bangladesh	PR0001	PR0002	PR0005	PR0006	PR0008
Uganda	PW0003	PW0009	PW0010	PW0013	PW0016

Table 3.4: Clinical samples used to test the iterative assembly approach. A total of 50 samples were chosen from 10 countries representing Africa, South East Asia and South America.

The average number of contigs with DBLα was $\sim$56 (Standard Deviation/SD=10) indicating that most samples contain the expected number of var genes. Isolates PM0096 and PP0011 were at the two extreme ends of the normal distribution with 78 and 26 *var*-contigs respectively (Figure 3.8). The sample with least number of *var*-contigs (PP0011) had poor data quality that affected the assembly (Appendix B). On the other hand, an increase in the number of *var*-contigs beyond the expected assuming a normal distribution ($\sim$76; considering 56+2SD) may indicate the presence of multiple infections. Additional challenges are envisaged with poor quality sequence data and mixed infections. Further evaluations taken to specifically look at these issues are described in the following sections.

3.3.4 Additional evaluations

3.3.4.1 Comparisons with *de novo* assembly

The sequencing technology and assembly tools have improved since the beginning of this thesis. It was thus important to evaluate if a *de novo* assembly of field isolates is practical due to significant improvements in yield and sequence quality. Although the assembly results were better than found in 2009, the iterative assembly approach described in this chapter generated the greatest

number of *var*-contigs (Table 3.5).

Clinical Sample	DBLα count	
	3 iterations	*De novo*
1	**57**	**48**
2	**56**	**45**
3	**54**	**39**
4	**61**	**48**
5	**97**	**86**

Table 3.5: Comparing iterative assembly with *de novo* assembly using 5 clinical samples (read length =100 bp; Velvet used for *de novo* assembly). The iterative assembly approach generated more *var*-contigs than a simple *de novo* assembly.

The iterative assembly approach was able to generate N50 contig sizes of up to 7.9 kb (Table 3.6) for Sample 1 (57 *var*-contigs) indicating both the efficiency of the method and benefits of long reads in assembling *var* genes.

Sample	Sum(bp)	N50(bp)	*var*-contigs	Largest(bp)
1	423,521	7,902	57	13,813
2	418,299	7,478	56	13,740
3	314,747	6,236	54	10,229
4	401,616	7,139	61	12,715
5	610,588	6,708	97	12,457

Table 3.6: Assembly statistics of the five clinical samples using the iterative assembly approach.

Sample 5 had the highest number of *var*-contigs suggesting presence of multiple infections. It was thus excluded from subsequent comparisons.

3.3.4.2 Mixed assembly

Reads from the first four of the five clinical samples (previous section) were first assembled individually for one iteration resulting in 58 to 67 *var*-contigs (Table 3.7). The assembly results were different from those shown in the previous section (comparisons with *de novo* assembly), as expected from the iterative process. The completeness of the *var* repertoire (i.e. number of contigs) and coverage (the sum of contigs) varies with each iteration due to the increase

in reads that are available to perform the iterative assembly. Concatenating contigs from the four samples resulted in a total of 249 *var*-contigs. Conversely, assembly of mixed reads generated in a total of 230 contigs with DBLα.

As these samples were obtained directly from patients, evaluating the quality of the assembly by comparing to the reference genome was not informative. We therefore compared Open Reading Frames (ORFs) from the two sets of assemblies using BLAST. The results revealed that ORFs from the mixed assembly overlapped with 33 to 84% of ORFs in the individual assembly (99%, match length of 300 aa) (Table 3.8).

Sample	Sum	N50	Num. scaffolds	Largest
1	316,696	5,984	67	10,745
2	323,601	6,263	65	10,990
3	134,458	4,011	58	8,420
4	317,902	6,915	59	11,271
Total	**1,092,657**	**6,163**	**249**	**11,271**
Mixed assembly	**813,249**	**5,348**	**230**	**11,114**

Table 3.7: Comparing individual assembly with mixed assembly of four clinical samples: Assembly statistics. Sample 3 had a relatively poor quality assembly compared to the other three samples.

As these samples were obtained directly from patients, evaluating the quality of the assembly by comparing to the reference genome was not informative. We therefore compared Open Reading Frames (ORFs) from the two sets of assemblies using BLAST. The results revealed that ORFs from the mixed assembly overlapped with 33 to 84% of ORFs in the individual assembly (99%, match length of 300 aa) (Table 3.8).

ORFs with the DBLα domain were first extracted from contigs of the two sets of assemblies (i.e. mixed and individual assemblies). ORFs of the mixed assembly were then compared with ORFs from the four samples resulting in an overlap of 33 to 84% of the total in each sample (99% identity and 300 aa). The fewer ORFs in Sample 3 are indicative of a poor quality assembly. Although 300 aa was a reasonable size to compare the two sets of ORFs, the results presented in Table 3.8 do not reveal the extent of overlap between the ORFs (i.e. the proportion of each ORF aligned at 99% identity, also called the

	Sample1	Sample2	Sample3	Sample4
Found in mixed assembly (total)	48(62)	52(68)	14(42)	48(57)
%Total	77	76	33	84

Table 3.8: Comparing ORFs of individually assembled contigs with ORFs of the mixed assembly. ORFs from the mixed assembly were compared with the ORFs obtained from the four samples. This table shows the count of ORFs of the individual assemblies that overlapped with ORFs of the mixed assembly with a minimum match length of 300 aa and a minimum identity of 99%.

coverage of ORFs). Therefore, additional comparisons were made based on the coverage of ORFs by varying the threshold from 5 to 100% (Figure 3.9). Up to 30% of contigs in the individual assembly were matched to mixed assembly over the full length of their ORFs. The remaining contigs were only partially covered with break points potentially caused by repetitive or shared sequence blocks. Aligning raw reads back to the contigs allowing multiple mapping for non unique reads revealed regions of excess coverage that correlated with contig-ends (i.e. break-points) in the mixed assembly (Figure 3.10).

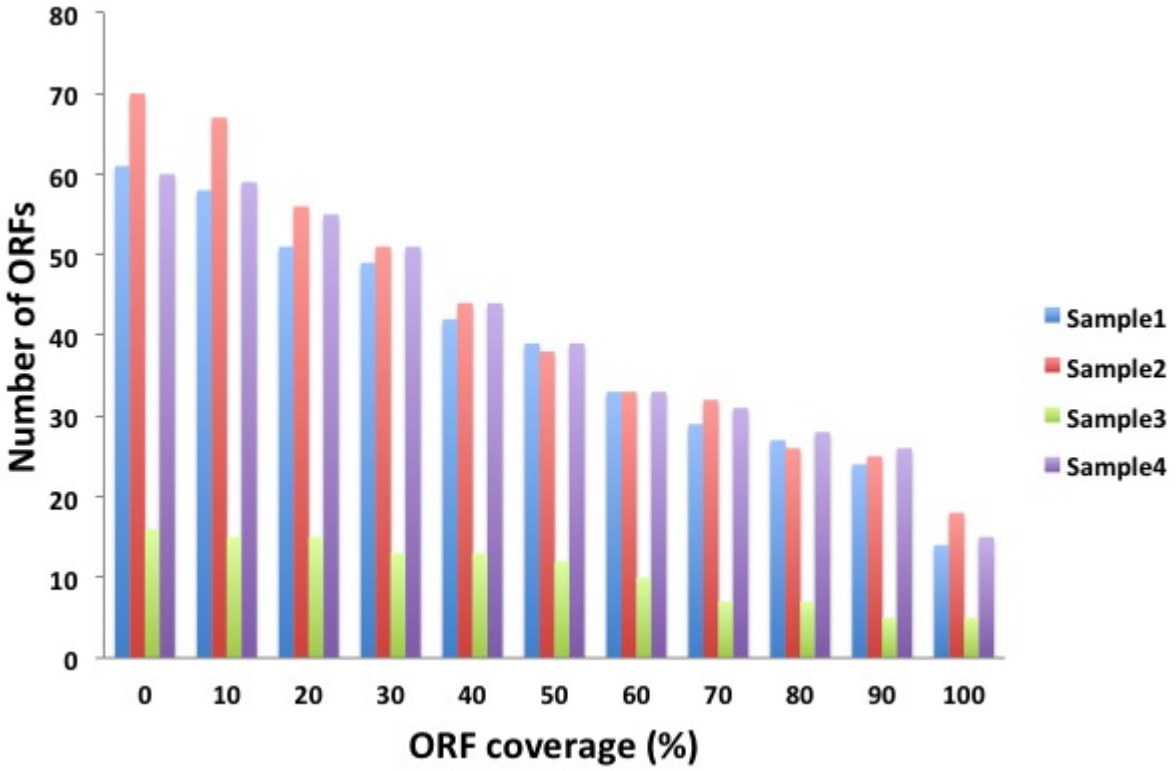

Figure 3.9: Comparing ORFs obtained from individual and mixed assembly by varying the proportion of ORFs covered. ORFs from the mixed assembly had a higher overlap with ORFs from individual assemblies at lower coverage thresholds. As the coverage requirement increased, the number of ORFs (from each individual assembly) that overlapped with ORFs from the mixed assembly decreased.

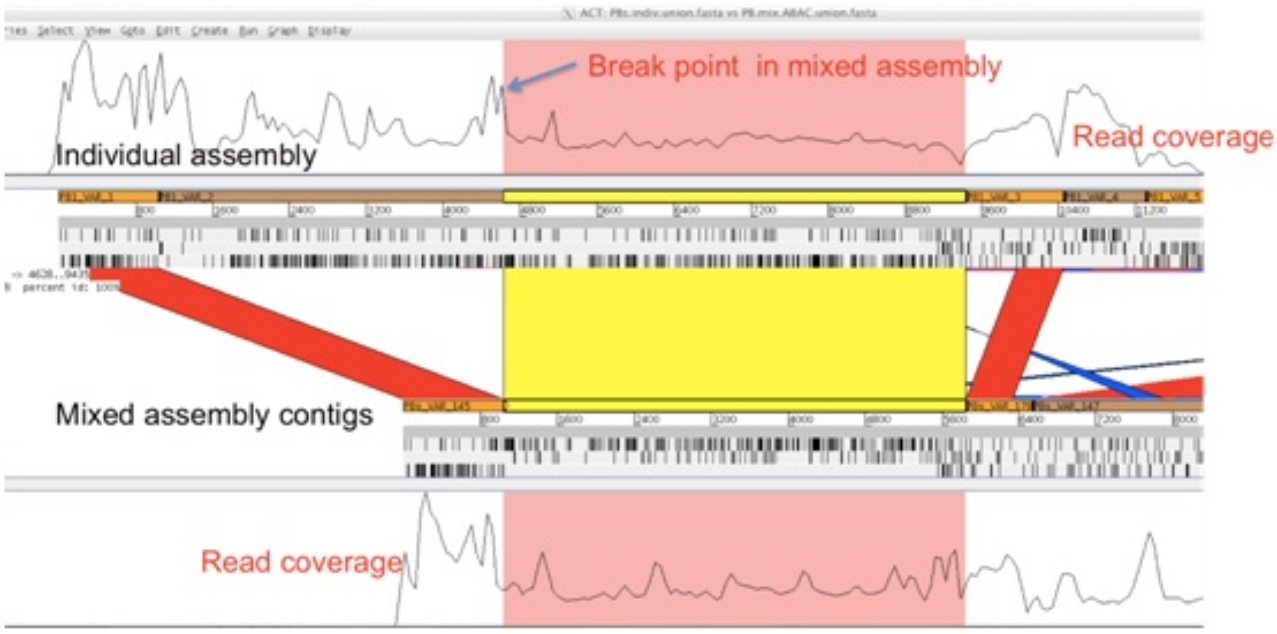

Figure 3.10: An ACT view of read coverage over *var*-contigs generated from individual and mixed assemblies. This example shows a BLAST comparison of *var*-contigs from the individual assemblies (top panel) against the mixed assembly (middle panel). Contigs are shown in alternating orange and brown blocks. The yellow, red and blue blocks (vertical) show synteny matches. Black bars in the middle panel represent stop codons. The contig on the top panel is partially covered (shown by the yellow match) by a contig from the mixed assembly with the breakpoint corresponding to a repetitive region (shown by the increased read coverage).

3.4 Discussion

The aim of this chapter was to develop an alternative approach that will address the limitations of existing assembly approaches in polymorphic gene families. This section presents a discussion of the findings in the following four conclusions.

Conclusion 1: *An iterative assembly approach based on conserved motifs provides a better way of reconstructing the var gene family in P. falciparum*

Gene families have blocks of highly similar and polymorphic regions and continue to evolve by accumulating additional polymorphism as well by recombination within these families. This ongoing microevolution leads to the maintenance of an extremely diverse gene repertoire. The *var* gene family is especially known to contain mosaic blocks that are highly recombinogenic and polymorphic (Bull et al., 2008; Frank et al., 2008; Kraemer et al., 2007). These regions pose significant challenges to standard short read assembly approaches. We have developed an assembly approach that takes advantage of the mosaic nature of *var* genes such that short conserved are used to initiate an iterative assembly. The new approach produced *var*-contigs (contigs that contain the universal DBLα tag) that were accurate and had high repertoire coverage. The efficiency, coverage and accuracy of the approach were demonstrated using sequences from four culture-adapted and 50 clinical samples. The potential of our approach to accurately assemble clinical samples was demonstrated first by assembling *var* genes of the 3D7 genome where initial motifs were generated from unrelated samples with incomplete *var* repertoire. The single misassembly detected in the 3D7 *var* assembly was due to a merge in two highly identical regions of *var* genes on chromosome 12 (central cluster). Such wrong joins are expected as identical segments between *var* genes of the 3D7 genome could be as long as $\sim$5 kb (Chapter 2). Resolving ambiguities in the assembly graph of such long shared segments is not possible with the standard library size of 200 - 300 bp. A fragment size longer than the shared unit is required to accurately assemble the full length of *var* genes that share long sequence stretches. Such cases of misassembly result in frame shifts during aminoacid translations and

were detected as part of the quality control process. Additional quality checks that take advantage of read-mapping coverage of raw reads are also being incorporated in the assembly pipeline. Examination of the read coverage patterns will reveal drops in paired-end coverage as potential signs of false joins.

Conclusion 2: *Motifs are an important part of this approach*

Motifs provided an important part of the assembly process. Identifying conserved (shared) elements across the length of *var* genes and including the mates of reads that contained motifs made it possible to generate sequence islands at the initial stages of the assembly process. These islands were considered as points of initiation for further iterative extension. A combination of iterative scaffolding and extension was used to close gaps between seed regions by walking-in and out of the sequence islands. Assembly quality was affected by data quality, yield and the coverage of motifs used to initiate the process. The increase in motif space at the beginning is characteristic of the initial stages of the assembly process where new motifs are being added to the collection. On the other hand, at later iterations, the majority of the motif would already be in the collection resulting in a decrease in the rate of accumulation.

Conclusion 3: *Iterations provide the mechanism for a controlled extension of the var gene repertoire*

Extremely low and extremely high read coverage are potential reasons for poor quality assembly. An iterative extension approach provided a means for identifying reads that could be used for a gradual extension of seed contigs. As the number of iterations increases the number of motifs identified also increased. A limited number of iterations was required to attain motif saturation ($\sim$5 for culture-samples and $\sim$7-10 for 50 clinical samples). This observation has implications on the number of iterations required to gather motifs in order to assemble a given set of samples. Motifs generated from the 50 clinical samples are representative of different geographical regions. Conditions for terminating iterations are determined based on the assembly quality (i.e. repertoire completeness as measured by the count of *var*-contigs, contiguity of assembled

contigs as measured by N50 and largest contig sizes, and repertoire coverage as measured by sum of *var*-contigs). Once an optimal number of iterations is achieved, assembly quality will stop improving or begin to deteriorate signaling an exit condition from the iterations.

Conclusion 4: *This approach provides a way of reconstructing var genes from clinical samples and get a complete view of the var repertoire for the first time*

Assembly results of 50 clinical samples resulted in the largest collection of *var* genes so far. The total number of contigs (n=2,769) found with the DBLα tag (N50=5.5 kb; Largest=14 kb; sum of contigs = $\sim$15 Mb) represented over 92% of the expected $\sim$3,000 contigs (expecting 60 *var* genes per genome). Samples with below 30 *var*-contigs were associated with poor quality in the raw data. Conversely, samples with over 70 *var*-contigs were shown to have multiple infections by a visual inspection of MSP1 genes. Assembly test of mixed samples resulted in shorter contigs than in the case of individual assemblies. However, within the majority of cases that were visually inspected, the breakpoints corresponded with the regions of high read coverage as expected. These regions are known to cause breaks in assembly due to ambiguities that could not be resolved using standard sized libraries (as discussed in the previous chapter). Samples used for mixed assembly tests were not normalised for coverage in order to represent over-representation of some genotypes in natural populations.

In summary, this chapter presented an alternative assembly approach to effectively reconstruct the *var* gene family from short reads of second-generation sequencing platforms. Applications of the method to perform a targeted assembly of the family were demonstrated using culture-adapted and clinical samples of *P. falciparum*.