CRANFIELD UNIVERSITY

CRANFIELD HEALTH

MSc THESIS

Academic Year 2010-2011

Neil Horner

Creation of a Software Tool for Browsing Genome Variation

Supervisors: Dr Lee Larcombe & Dr Taane Clark

October 2011

This thesis is submitted in partial fulfilment of the requirements for the Degree of Master of Science in Applied Bioinformatics.

Approved by the Sub-Board on Taught Postgraduate Courses at the meeting held on 1st November 2001 and the Sub-Board on Research Students at the meeting held on 22nd October 2001. Amended 29th July 2004, to reflect Crown Copyright requirements.
Updated 16th March 2007

Abstract

The advent of next generation sequencing has led to an explosion of the amount of DNA sequences in public databases. A challenge is now to find tools that are able to make it easier for researchers to browse and make sense of this data. One organism that has recently been subject to extensive sequencing is *Plasmodium falciparum*, a devastating pathogen that infects hundreds of millions of people annually. The first goal of this project was to create a new desktop genome variation browser that can quickly handle large amounts of data from sequencing projects involving numerous isolates. The second aim was to use the new tool to analyse recently-sequenced strains of *P. falciaprum* in order to identify polymorphisms that may be involved in antibiotic resistance.

The variation browser described here was written in C++ and the Qt graphical framework in order to make an easy to use and fast tool that can visualise data from variant call format (VCF) files, which is now a de facto standard for storing polymorphism data. The user is able to browse a VCF file to gain a graphical representation of the variation among multiple samples. For rapid identification of relevant polymorphisms, the user is able to filter variant positions using several criteria including mapping quality, sample group membership, and whether the mutations alter the amino acid sequence of a gene. Some basic statistical analysis was incorporated to help identify selective pressures acting on polymorphic sites.

The usefulness of the program was ascertained by analysing 75 isolates of *P. falciparum* from Africa and Asia. Mutations were identified in the chloroquine resistance marker protein, PI4-K, and a putative ubiquitin carboxyl hydrolase, which are potentially involved in antibiotic resistance.

Acknowledgments

I would like to thank my project supervisors Dr Taane Clark and Dr Lee Larcombe for their excellent guidance. I would also like to thank Dr Mark Preston for helpful discussions and C++ trouble-shooting.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

aCGH	Array comparative genome hybridisation
ASM	anchored split-mapping
BCF	Binary variant call format
CDS	Coding sequence
CNV	Copy
CQ	Chloroquine
G(UI)	Graphical user interface
HTS	High throughput sequencing
PCR	Polymerase chain reaction
PEM	Paired end mapping
Pf	Plasmodium falciparum
QRD	Quinine-related drugs
RD	Read depth
SB	Subterlomeric block
SNP	Single nucleotide polymorphism
SR	Split read
SV	Structural variation

1. Introduction

There are approximately 500 million cases of malaria per annum, resulting in the death of between one and three million people annually (Snow et al. 2005). The great majority of this mortality is caused by *Plasmodium falciparum* (Pf). The heterogeneity between Pf genomes can be accounted for by structural variation (SV), including insertions and deletions (indels), inversions, translocations, and copy number variants (CNVs), and are likely to make important contributions to phenotypic diversity. SVs, in particular CNVs, are abundant in Pf genomes (Kidgell et al. 2006; Ribacke et al. 2007). Within infected hosts, malaria parasites are subjected to strong selection from exposure to antimalarial drugs and the immune system. CNVs are associated with Pf phenotypes such as erythrocyte invasion and drug susceptibility. For example, CNV in the gene encoding the multi-drug resistance protein (PfMDR1) is associated with resistance to various anti-malarial drugs: high frequencies of this CNV have been observed in South East Asia and are associated with an increase in drug resistance (Price et al. 2004) which can be reduced by transgenically lowering copy number (Sidhu et al. 2006).

1.1. Plasmodium falciparum

Malaria is one of the world's most devastating human diseases, annually infecting around 500 million people, and resulting in the deaths of between one and three million people (Snow et al. 2005). Not only is the direct cost in human suffering high, but countries with high prevalence of malaria have markedly lower GDPs compared to countries where malaria is not endemic. Countries with endemic malaria were shown to have a GDP growth rate that was 1.3% less than malaria-free countries even when other confounding factors were taken into account, such as geographical location, initial poverty levels, and life expectancy (Gallup & Sachs 2001). Climate change is also predicted to spread the disease to regions that are currently malaria-free in the future as these countries become warmer. In countries that are already endemic for malaria, an increased rate or recrudescence of malaria infections may be a result of increasing temperatures, as has already been observed in highland areas of east Africa where there has been a significant increase from the 1970s to 1980s (Alonso et al. 2011). Malaria is also thought to increase the risk of the transmission of HIV, with HIV patients displaying a transient increase in HIV viral load upon infection (Abu-Raddad et al. 2006).

1.1.1. Life cycle

The life cycle of *P. falciparum* is complex, with many developmental stages, and the requirement of a human and mosquito host to complete its life cycle. In the midgut of the Anopheles host, sexual selection results in the formation of human-infective sporozoites, which then migrate to the salivary glands. Upon biting of a human host, sporozoites are transferred into the small blood vessels, then migrate to the liver and invade hepatic cells. Within liver cells, multinucleate schizonts develop that contain between 2,000 to 40,000 uninuclear merozoites. This liver stage of replication, or exoerythrocytic schizgony, takes between 5 and 21 days to complete, after which the mature schizonts rupture the liver cells and escape into the blood stream, releasing thousands of uninucleate merozoites. The merozoites go on to invade erythrocytes where they form an enclosing parisitophorous vacuole and undergo a trophic phase. The early trophozite, because of its morphology, is sometimes called a 'ring form' stage. The trophic phase is characterised by the import of nutrients from the host cytoplasm, including hemoglobin. This occurs in the digestive vacuole with the production of haem as a by product. The haem is converted to a non-soluble non-toxic form called hemozin. Within the infected erythrocyte, as the trophic phase ends, multiple rounds of nuclear division, without cytokinesis, are initiated, which results in the formation of multinucleate schizonts.

The schizont is cleaved to release uninuclear merozoites that are released into the blood stream as the erythrocyte is ruptured. These merozoites are then able to infect other erythrocytes producing multiple rounds of invasion and increase of parasite numbers. An alternative developmental pathway to the erythrocyte -enclosed asexual cycle can also occur: The parasite differentiates into mononuclear sexual forms called microgametocytes and macrogametocytes, which can fill up most of the volume of the infected erythrocyte. When a feeding mosquito takes up an erythrocyte containing a gametocyte, gametogenesis is induced and the flagellated microgametes and the aflagellate macrogamete are released. The former fertilisers that latter forming a zygote. A motile ookinete develops from the zygote and then invades the mosquito gut epithelia where it develops into an oocyst. Multiple rounds of asexual reproduction ensues, resulting in the production of sporozoites. The sporozoites are released from the epithelial cells upon rupture into the body cavity of the host (hemocoel), after which they migrate to and invade the salivary glands from where they can go on to invade new human hosts (reviewed by Greenwood et. al. 2008; Figure 1.1)

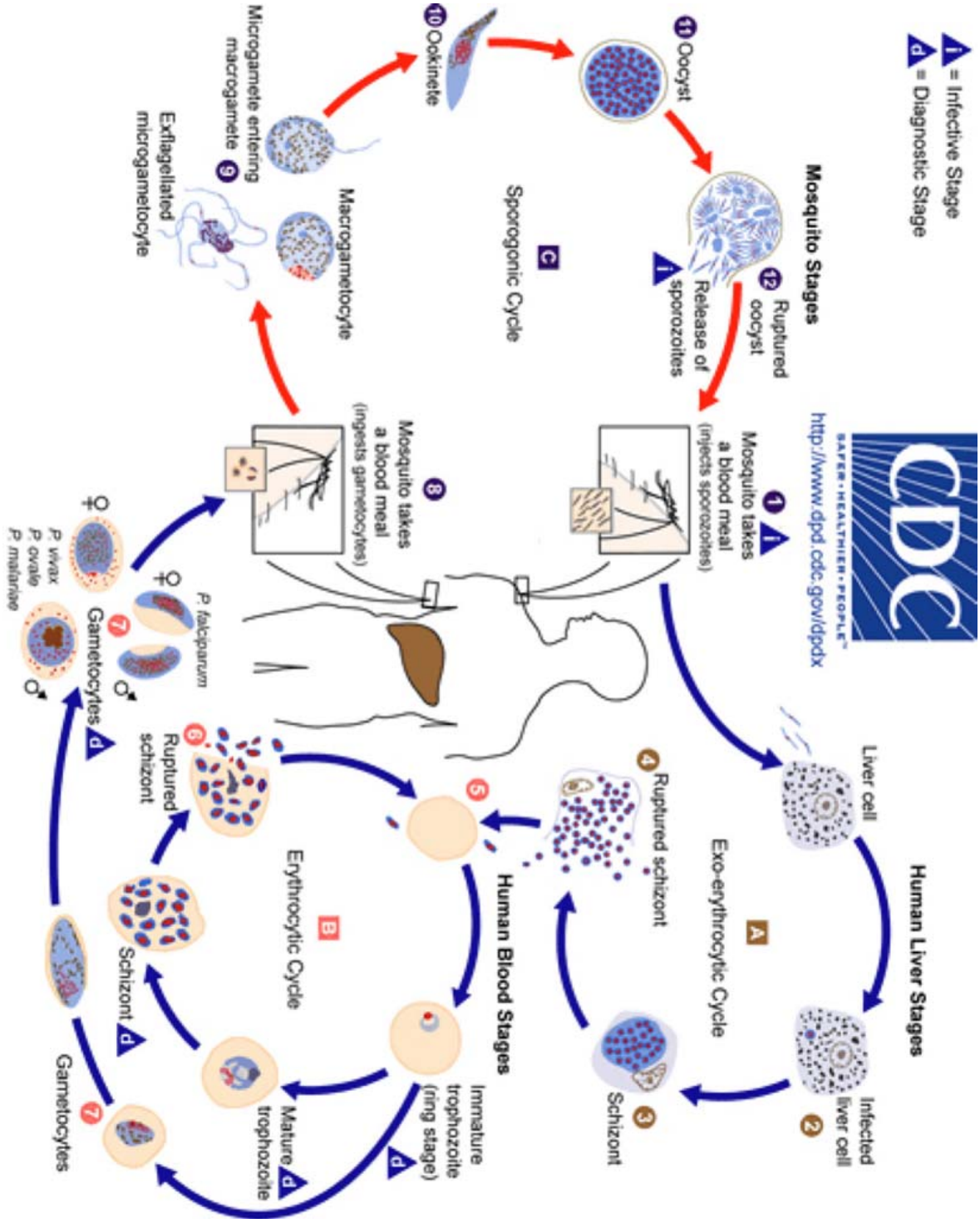


Figure 1.1: Plasmodium falciparum life cycle from http://www.cdc.gov/malaria

PfEMP1 is a collective term for the var gene products, which have been shown to be important in inducing immunity within the human host. There are around 60 var gene members are located predominantly in subteloremic regions of Pf chromosomes. PfEMP1 is localised to the surface of infected erythrocytes and is responsible for cytoadherance to other host erythrocytes, the capillary walls, and post-capillary venular endothelium by binding to various host receptors such as ICAM1 (Berendt et al. 1989), CD36 (Barnwell et al. 1989), ELAM-1, and VCAM-1 (Ockenhouse et al. 1992). This cytoaherance allows the infected erythrocytes to become sequestered within organs such as the kidney, brain, liver, and lung, preventing clearance by the spleen, and providing a microaerophillic environment that is best-suited to maturation of the parasite (L. H. Miller et al. 2002). PfEMP1 is recognised by the host immune system and *P. falciparum* employs a method of avoiding this by a process called antigenic switching, whereby limit the number of var gene variants that are expressed at any time is regulated. The mechanism of switching involves the relocation of var genes to perincuclear positions and heterochromatin modification (Duraisingh et al. 2005). About 18% of a population has been shown, *in vitro,* to switch per generation (Gatton et al. 2003). As the host develops antibodies for a specific var gene product, parasites that are expressing an alternative version are strongly selected for. STEVOR is another protein family whose encoding genes are also located in subtelomeric regions. They possess a transmembrane span, which is inserted into the infected erythrocyte plasma membrane. Similar to the var gene products, they have been shown to be clonally variant, and while their primary function is unknown, they are known to be involved in inducing a host immune response (Niang et al. 2009).

1.1.2. Chemotherapy and drug resistance

Quinine-related drugs (QRDs) including chloroquine, piperaquine, amodiaquine, primaquine, quinine and mefloquine, are one of the most widely used, cost-effective, and safe groups of chemotherapeutic agent used to treat malaria infections. They are weak bases that diffuse readily across membranes. When in an acidic environment, they become protonated thereby rendering them non-permeable to membranes. Because of this, QRDs accumulate in the acidic digestive vacuole, where they prevent the polymerisation of haem to hemozin, leading to an increase in the concentration of toxic free haem. Free haem can permabalise membranes and lead to the production of reactive oxygen species (ROS), which can damage many cellular components. As well as interfering with haem polymerization, other targets have been proposed for QRDs, including phospholipases, DNA, tyrosine kinase, and

haemoglobin- degrading proteases (Kaur et al. 2010).

Folate biosynthesis is also commonly targeted by antimalarials. The disruption of the folate biosynthesis pathway prevents the production of pyrimidines and amino acids, thus effectively blocking DNA and protein synthesis. Two main targets are present in the *P. falciparum* folate pathway: dihydroopteroate synthase, targeted by sulfonamides; and dihdrofolate reductase, targeted by pyrimethamine, cycloguanil, and methotrexate (reviewed in Gregson & Plowe 2005). The most common anti-folate preparation used presently is Fansidar, which is a mixture of sulfadoxine and pyrimethamine.

The current gold standard for the treatment of malaria is artemisinin and its derivatives. The mode of action is not fully understood, but all artemisinin-related drugs contain an endoperoxide bridge that has been shown to be required for activity. It is thought that that high free haem concentrations in the parasite digestive vacuole catalyses the cleavage of the endoperoxide bridge leading to the formation of electrophillic derivatives and free radicals that can damage membranes and have been shown to alkylate several *P. falciparum* proteins (Asawamahasakda et al. 1994). Artemisinin is effective against both ring and schizont stage parasites (Skinner et al. 1996). More recently, a L263E amino acid substitution in PfATPase6, a SERCA-type Ca^{2+}-ATPase, abolishes sensitivity to artemisinin, and has been suggested to be a target of artemisinin (Uhlemann et al. 2005).

1.1.3. Evolution

Humans are thought to have been hosts to malaria parasites since they diverged from the last common ancestor with the chimpanzee. This divergence is mirrored by the divergence of the malaria parasite *P. reichenowi* in chimpanzees (Ollomo et al. 2009). *P. falciparum* is a member of the phylum Apicomplexa that also contains other important human parasites such as *Cryptosporidium* spp. and *Toxoplasma* spp. In a recent reorganisation of eukaryotic taxa, the apicomplexans have been grouped into the diverse chromoalveolata kingdom, which contains diverse microbes such diatoms, the oomycetes that contains the causative agent of potato late-blight *Phythophthora infestans,* as well as large multicellular organisms such as brown algae. More recent work casts doubt on the monophylogeny of the chromalveolata, but it is widely accepted that the grouping of the stramenopiles and Alveolata are monophyletic (Adl et al. 2005).

1.1.4. *P. falciparum* genome organisation

The *P. falciparum* genome was sequenced by an international consortium and the draft sequence was published in 2002. The *P. falciparum* genome is 22.8 Mb and comprises 14 chromosomes with chromosome size increasing with number assigned ranging from 0.64 to 3.29 Mb. The G+C content is one of the lowest of any fully-sequenced organism at 19.4% overall and decreasing to ~10% within intergenic regions. Around 5300 protein-coding genes have been identified having an average length, excluding introns, of 2.3 kb. This gives an average gene density of one gene per 4,338 bp. Introns were identified in 54% of genes. Subtelomeric regions are thought to have been subject to promiscuous recombination between chromosomes, giving rise to high inter-chromosomal conservation at these regions (Gardner et al. 2002). Subtelomeric regions are observed to be highly diverse between *Plasmodium* species and contain 575 *P. falciparum* species-specific genes from a total of 743 identified by whole genome comparison with rodent malaria pathogens. *P. falciparum* species-specific genes are also enriched for at sites of syntenic block break-points that made up the core plasmodium genome (Kooij et al. 2005). Five subtelomeric blocks (SBs) have been identified. SB1 consists of telomeric repeats with the consensus sequence GGGTT(T/C)A. SB2 contains five different repeats that are separated by non-repetitive elements. SB2 repeats have a raised G+C content of ~30% compared to the average of 10% for other non-coding regions. SB4 contains more repetitive elements with at least one var gene that is usually in a telomere to centromere orientation. SB5 can extend up 120 kb and contains members of the stevor, rif, and var gene families (Crabb & Cowman 2002; Figure 1.2). In the original *P. falciparum* genome sequence paper (Gardner et al. 2002) no evidence of mobile genetic elements (MGE) was found. A more recent analysis has uncovered the signature of three possible MGEs (P. M. Durand et al. 2006). This paucity suggests that the evolution of the *P. falciparum* has not been greatly influenced by MGEs.

1.2. Genome variation

Variations in the genome sequence between related species can be classified into three broad categories: (1) Single nucleotide polymorphisms (SNPs), where one nucleotide has been substituted for another; (2) Structural variants (SV), which are here defined as regions of the genome displaying inversions, insertions, or deletions; (3) Copy number variants, which are genomic regions that are present in varying amounts in different individuals. CNV is a subset of SV as deletions and insertions will also decrease and increase the copy number respectively.

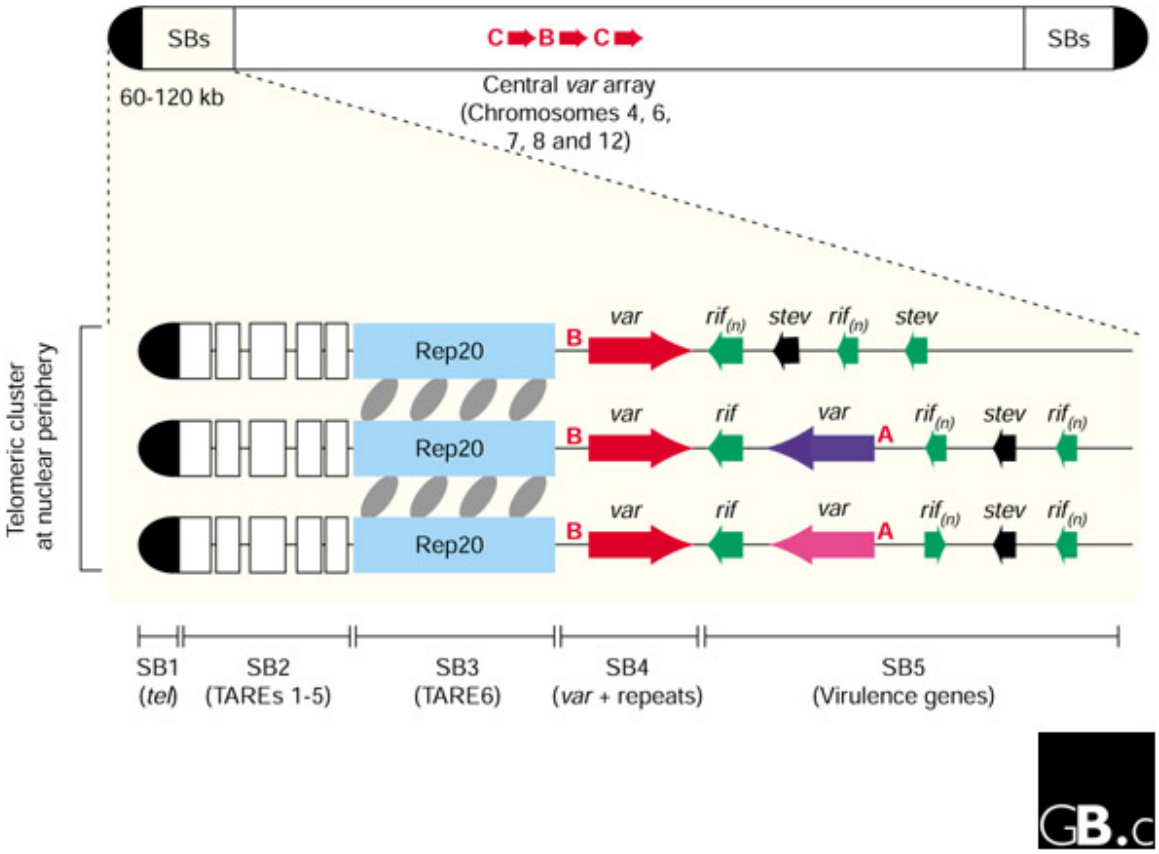


Figure 1.2: Organization of three typical P. falciparum subtelomeric region. SB, subtelomeric blocks; Rep20, 21bp repeats regions; TARE, telomere-associated repeat elements. From (Crabb & Cowman 2002).

1.2.1. Single nucleotide polymorphisms

SNPs are the most simple of polymorphisms. In recent years, they have received a great deal of attention in personalised medicine, forensics, population genetics and other fields. SNPs within coding regions, if they create a codon that encodes a different amino acid (non-synonymous substitution), will alter the encoded protein and possibly the phenotype. Identification of SNPs serves many purposes including genotyping, determining population structure, and identifying types of selection acting on

regions of a genome. Vast numbers of SNPs can exist within populations. For example, The Human Genome Project, the SNP Consortium and the International HapMap Project have together identified 10 million common (present in over 1% of the population) variants, most of which are SNPs.

1.2.2. Copy number variation

A copy number variant (CNV) or copy number polymorphism (CNP) has been defined as a region of a genome, arbitrarily, larger than 1 kb, that displays variation in the number of copies present relative to a reference genome. In humans, a seminal study of CNV in humans uncovered 1,447 CNV loci that together cover around 12% of the genome (Redon et al. 2006). In an association study of gene expression and genomic variation in lympho-blastoid cell lines from 210 unrelated HapMap individuals, CNVs were shown to be the cause of 17.7% of variability in gene expression differences between individuals (Stranger et al. 2007). Many more CNVs have been identified since then with the Database of Genomic Variation (http://projects.tcag.ca/variation/) recording CNVs (>1 kb) at almost 16,000 loci in the human genome as of April 2011. The mechanisms that confer phenotypic variation in response to CNV are varied. Gene dosage is probably the most straight-forward outcome of CNVs, where the amount of gene expression commonly correlates positively with copy number. For example, an increase in copy number, and concomitant gene-dosage of *PMP22* leads to Charcot-Marie-Tooth disease type 1A in humans (J R Lupski et al. 1992). An increase in copy number of a gene may not always lead to increased transcript abundance. For example, a modelling study of transcriptional networks has shown that changing gene copy number can create radically different steady-states, that transcriptional networks outputs can change non-linearly to changes in copy number variation, and that transcriptional networks can be driven between oscillatory and non-oscillatory states (Mileyko et al. 2008). Therefore the effect of CNV on gene expression is unpredictable. Another outcome of an increase in copy number is the loss of functional constraint on the evolution of one or the other gene and the ability for new functions to be acquired, thus providing a substrate for evolution to work on.

1.2.3. Mechanism of SV creation

Copy number variation, whatever the mechanism of formation, involves recombination and the joining together of two regions of genomic DNA that were previously separated. The regions of these joining events are called break-points and analysis of them can provide clues to the mechanism involved.

CNVs commonly accumulate due to a non-homologous end-joining (NHEJ) mechanism. NHEJ is employed by the cell to repair chromosomal breakages, sometimes leading to translocations when non-homologous regions are joined together, or deletions can occur when two-ended DSBs occur. For instance, when endonuclease-created breaks or when converging replication forks move into a region with nicks in the DNA (P. Hastings, James R Lupski, et al. 2009). Recently, another mechanism has been implicated in the generation of CNVs, termed microhomology-mediated break-induced replication or MMBIR (Hastings et al. 2009). Non-allelic homologous recombination (NAHR) can occur during meiotic recombination or during a double-stranded break repair event if the regions are non-allelic, but share homology. For example there may be a series of highly similar tandemly repeated genes, and the first gene in one chromosome recombines with the second gene of the repeat sequence in the sister chromatid leading to a duplication in one and a deletion in another (unequal crossing over). If they are directed repeats then duplication of deletion can occur, whereas inverted repeats can lead to an inversion (Speicher 2009).

Different mechanisms of CNV creation can be detected by the remnant signature that is left behind at break points. If there is over 100 bp of sequence homology at breakpoints then NAHJ is a likely cause. Blunt ends at the break points indicate NHEJ. Insertion of local sequence >20 bp indicates MMBIR. Microhomology of < 20 bp indicates NHEJ, MMEJ, or MMBIR, while dispersed duplication can indicate retrotranspostion-mediated duplication (Reviewed by Conrad et al. 2010).

1.3. Genomic variation in *P. falciparum*

1.3.1. P. falciparum CNV

The mechanism of CNV generation in *P. falciparum* is thought to be commonly due to non-homologous recombination events at repetitive regions. For example, the breakpoints surrounding *pfmdr1* occurred predominately at monomeric A or T tracts. A comparison of 16 *in vitro*-cultured *P .falciparum* genomes with the 3D7 genome using high-density Affymetrix oligonucleotide arrays uncovered a total of 186 genes that showed CNV, with each isolate having between 11 and 37 CNV genes per genome. The distribution of CNVs was non-random, with 60.8% occurring in subtelomeric regions. 18% of all genes located in the subtelomere region were CNVs compared to just 1.6% of genes internal of subtelomeres. CNVs were also enriched near sites of chromosomal segmental duplication sites. It was found that gene length was inversely proportional to the likelihood of being a CNV gene.

An over-representation of CNVs was found for genes that encode known antigenic proteins, and was proposed as a mechanism of generating antigenic diversity in order to evade immune recognition (Cheeseman et al. 2009).

1.3.2. Variation affecting antibiotic-resistance phenotypes

There are many types of genome variation seen across *P. falciparum* isolates that can confer unique phenotypes associated with pathogenicity, including single nucleotide substitutions (SNPs), small scale indels, CNV, as well as translocations and gene inversions. Deletion of a subtelomeric region of chromosome 9 containing clag9 resulted in loss of cytoadherence to melanoma cells (Trenholme et al. 2000). One study has found a very high propensity for CNV genes in *P. falciparum* to be species-specific, with around 70% of CNV genes having no ortholog in other *Plasmodium* species. This was suggested to be due to negative selection acting on the core *Plasmodium* genes that are functionally constrained, or to diversifying selection acting on the species-specific genes (Cheeseman et al. 2009). The *P. falciparum* multidrug transporter (*pfmdr1*) encodes a transporter with 12 transmembrane domains that is localised to the digestive vacuole within the parasite. Amplifications of *pfmdr1* and increase in *pfmdr1* transcript levels are associated with an increase in resistance to chloroquine (Foote et al. 1989). Ablation of one of two *pfmdr1* genes in a MFQ-resistant strain led to dramatic increases in susceptibility to MFQ and several other drugs. In an *in vitro* experiment in which *P. falciparum* was exposed to mefloquine, it was estimated that increases of copy number of *pfmdr1* from one to two copy numbers occurs at a frequency of 10^{-8} per generation, while change from two to three copies occurs at 10^{-3} per generation (Preechapornkul et al. 2009). Another multidrug resistance protein that has been shown to mediate sulfadoxine resistance is PfMRP1 (Dahlstrom et al. 2009).

Dihydropteroate synthase (*dhps*) and dihydrofolate synthase (*dhfr)* are targeted by the anti-folate drugs sulfadoxine and pyrimethamine respectively, and mutations in these genes can confer resistance to drugs that are targeted by them (C V Plowe et al. 1998). However, a fitness cost is induced, which is thought to be compensated for by an increase in copy number of GTP-cyclohydrolase *gch1,* the first enzyme in the folate biosynthetic pathway. In Thailand an extensive use of anti-folate drugs have been occurring for decades, whereas in neighbouring Laos these drugs were rarely used. Consistent with a role in positive selection favouring multiple copies of *gch1* in response to selective pressure from anti-folate drugs, 72% of parasites from Thailand had *gch1* in copy number greater than 1 (1-11 copies), while this figure was only 1.6% (max. 2 copies) for parasites isolated in Laos. Additionally, linkage

disequilibrium was observed to be increased and genetic diversity reduced in the *gch1* locus in the Thai population relative to the Laos population, indicating strong positive selection acting on *gch1 (Nair et al. 2008)*. Another example of the effect of CNV on antibiotic resistance comes from a study in which *P. falciparum* Dd2 was cultured *in vitro* in the presence of piperaquine, a chloroquine-like drug. During exposure to the drug, a 65 kb region on chromosome 5 was amplified and antibiotic resistance increased. After culturing on antibiotic-free media, this region was lost along with antibiotic resistance, suggesting that a CNV in this region mediates antibiotic resistance (Eastman et al. 2011).

1.4. Identification of genome variation

Historically SVs were largely detected between genomes using cytogenetic techniques. While relatively simple and effective for very large SVs, the lack of resolution has been limiting. Detecting SVs with much higher resolution was introduced with the advent of the revolutionary technique, whole-genome tiling arrays (array-CGH or aCGH). This has been the standard tool for detecting SV since its development in 1997 (Solinas-Toldo et al. 1997). This method involves the creation of a microarray chip that is tiled with probes that cover the whole genome. A sample genome is fragmented and labelled. After hybridisation, the intensity of signal at each probe gives an indication of the amount of the corresponding genomic sequence present in the sample and so whether there are SVs. The resolution of aCGH has been steadily increasing with SVs now able to be detected at between 50–200 bp CNV (Urban et al. 2006). Problems with this approach include cross-hybridisation that can confuse the analysis. Also limited dynamic range can be a problem. Array design and production can be costly and optimisation of conditions can be troublesome. Another drawback is that array-CGH is not very well suited to detecting SVs such as inversions and balanced translocations. Paired end sequencing of fosmids is another approach that was used in the past, but with resolutions of > 8 kb is not suitable for the majority of SVs (Korbel et al. 2007).

1.4.1. Identification of SV from high-throughput sequencing data

With the advent of the so-called 'next generation' DNA sequencing technologies or high throughput sequencing (HTS) techniques, new avenues have been opened for the identification of genomic variation. The massive amounts of sequence data produced by these technologies lend them to the accurate detection of SNPs as well as other types of SV. Advantages of HTS-based variant detection over aCGH include the fact that the sequence of the organism need not be known before the

experiment. It overcomes problems with cross-hybridization, is roughly the same cost as aCGH, and is becoming progressively cheaper. Also there is a digital output that is easy to interpret and the dynamic range is much higher (Daines et al. 2009).

Probably the most straight-forward approach to identifying genome variation would be to assemble the full genome. However, with the current size of read length from massively parallel DNA sequencing being well under 100 bp, assembly of whole genomes is problematic. 454 sequencing does provide much longer reads but the cost per base pair is much higher (Table 1.1). This has driven the development of various techniques and tools to identify SV without assembling a whole genome.

Table 1.1: DNA sequencing costs. From Gupta et al. 2010

	Roche/454	Illumina/SOLEXA	ABI SOLiD	Polonator	Heliscope	Ion Torrent
Method for sequencing	Polymerase (pyrosequencing)	Polymerase (reversible terminator)	Ligase (octomer)	Ligase (nonamer)	Polymerase (single molecule)	Polymerase (single molecule)
Template amplification method	Emulsion PCR	Bridge PCR	Emulsion PCR	Emulsion PCR	None (single molecule)	None (single molecule)
Throughput	400–600 MB	17 GB	10–15 GB	~10 GB	21–28 GB	100 MB
Run time	10 h	4 days[a]	7 days[a]	4 days	8 days	1–2 h
Read length	~400 bp	~40 bp[b]	~50 bp	~15 bp	~30 bp	100–200 bp
Accuracy	99.74%	99.99%	99.7%	98%	99%	>99.995%
Cost	$60.0/MB	$2.0/MB	$2.0/MB	$1.0/MB	$1.0/MB	$500/run

[a]Number of days required for single end reads. [b]Paired-end reads can provide DNA sequence information for up to 80 nucleotides.

Identification of SNPs and small indels from aligned data is straightforward as long as the sequence read can be confidently aligned to a reference genome, and can be carried out with tools such as SAMtools (Li et al. 2009) or GATK (McKenna *et al.* 2010)

SVs can be called from NGS data by mapping paired-end sequences to a reference genome and comparing the distance between mapped-paired ends to the average size of the insert used to generate the library (paired-end mapping or PEM). Distances between pairs larger than the insert size range give an indication of an insert, whereas smaller length between pairs suggest a deletion event. PEM was first applied to detect genome-wide SV (Korbel et al. 2007). Using 454 sequencing and a 3 kb average insert size library, they were able to gain an average resolution of the breakpoints of 644 bp, small enough to easily characterise to single base pair resolution using PCR. SV presence was called only if

there were two or more supporting mapped pair reads. The authors use this to safeguard against rare chimeric constructs that can occur during library construction. This method provides sensitive detection of relatively small deletions and can provide high resolution mapping of the breakpoint as well as handling highly-repetitive regions well. A disadvantage of this method lies in the fact that insertions larger than the average insert size of the paired-end library are largely undetectable (Korbel et al. 2007; Figure 1.3).

Another strategy similar to PEM is split-read analysis (SR), which complements PEM-based approaches by allowing for the detection of much smaller insertions and deletions. This method looks for gaps in the alignment of single reads to a reference genome. Gaps in the reference genome indicate an insertion and a gap in the aligned read indicating a deletion. SR is more effective when used with sequencing technologies that produce longer read length, such as Roche 454 as there tend to be many regions in the genome that small split-reads can map to (Pang et al. 2010).

A modification of the split reads strategy that is used to overcome some limitations of short-read length sequencing, called anchored split-mapping (ASM). Using this approach, all reads that can only align to the genome at one end are found. The mapped end is required to map to a unique region in the genome, providing an anchor. Then an attempt is made to map the other end of the read as in the SR strategy. Only medium sized insertions can be detected using this method, as increasing the search space also increases the chances of finding multiple sites that the non-anchored end can map to. ASM can also detect deletions, but as the size of the deletion increases so does the amount of total read sequence available for aligning, thus limiting it to small deletions (K. Ye et al. 2009).

Another alternative and complimentary approach to PEM and SR-based strategies is the analysis of read-depth (RD) from NGS data. RD-based methods use read coverage at locations of the chromosome to infer copy number. In theory RD should be directly proportional to the copy number of a genome sequence. An advantage of RD CNV calling is that the sequence reads do not have to be mapped to unique regions of the chromosome and so it is easier to call CNVs in regions of complex duplications, regions which PEM-based methods struggle with, although highly repetitive regions such as LINEs and SINEs in humans can be problematic. In one study where a PEM-based and a RD-bases were compared, approximately the same number of SVs were identified, but very little overlap was seem between the called SVs between the two methods.

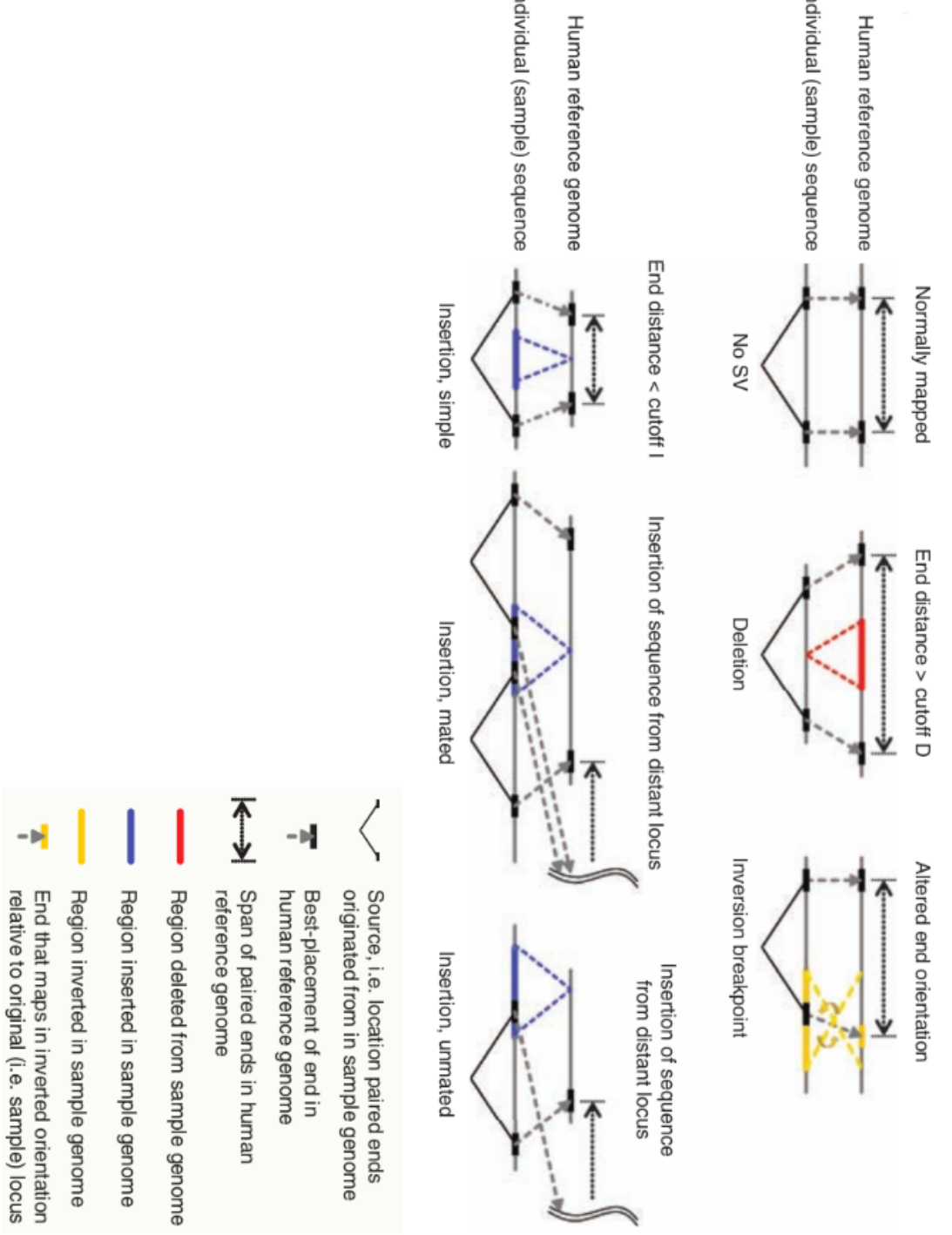


Figure 1.3: Different types of variants that can be identified using paired-end reads. From Korbel et al. (2007)

The PEM-specific calls contained only 2% segmental duplications compared to 40% for the RD method. The mean size of the SVs for the PEM method was 985 bp compared to 4.6 kb for the RD method (Yoon et al. 2009). Therefore multiple methods should be employed when a thorough investigation of SVs is required.

FREEC is a tool for automatically detecting CNV by analyzing read depth at various regions of the genome. It does not require a control genome to serve as a reference genome, although one can be used if available. This can be useful if a reference sample is not available, and also reduces costs as less sequencing needs to be performed. FREEC also can take polyploidy into account when calling CNVs. The algorithm first calculates a raw read depth based on read counts in non-overlapping windows that can be set manually or automatically based on the depth of read coverage. Two methods of normalisation can be employed: (1) The CNP is normalised using a control experiment using data from an organism that you wish to compare to (2) The CNP is normalised based on a G+C content profile. It is able to utilize information from paired-end reads, and is able to use low depth of coverage data, and because it is written in C/C++, is much faster than other programs that have been developed for similar purposes (Boeva et al. 2011).

Inversions can be difficult to detect using short-read NGS data because small reads within the inversion will still align to the reference genome (Pelak et al. 2010). Using paired-end reads can overcome this limitation: if one of two pairs maps in the reverse orientation, it is likely to be the result of an inversion. If one of a pair also maps to another chromosome, it can indicate a inter-chromosomal translocation event (Tuzun et al. 2005).

1.4.2. VCF file format

Recently the GFF file format, which has been used extensively to store many types of biological information such as genome annotations, has been modified to store variant information and is called the GVF format (Reese et al. 2010). However this format does not lend itself to storing data from multiple samples (Danecek et al. 2011). The need for a flexible format to store variant information that can handle multiple sample data has led to the development of the Variant Call Format (VCF) by the 1000 genomes project (Danecek et al. 2011). This is a text-based format that is usually compressed. See Table 1.2 for details, and appendix (7.1.1) for an example file segment.

Table 1.2: VCF file format details

Meta information lines	Start with ##
##INFO	Defines key-value fields in 8th column.
##FILTER	Indicates whether position has passed filtering.
##FORMAT	Provides information for each sample, one column per sample.
Column labels	Starts with #
CHROM	The chromosome
POS	Reference genome position at which point the first base of the variant maps
ID	List of identifiers (e.g. dbSNP) separated by semi-colons.
REF	The sequence of the reference genome at the variant position. In the case of indels the base preceding the indel must be included
ALT	List of alternate alleles found at this position, separated by commas.
QUAL	Phred-scaled quality score for the call of the alternate allele.
FILTER	Indicates whether the allele has passed a predefined cut-off filter.
INFO	Contains extra key=value fields. Can be arbitrary, but there are several optional reserved fields such as AA=ancestral allele, and AF=allele frequency.
FORMAT	contains genotype identifiers separated by colons, which correspond to the data values in each sample column. For example, for a format field containing 'GT:GQ:DP:HQ' and a sample field on the same line containing '0|0:48:1:51,51' The first line in the sample field represents a phased genotype of 0|0, the meaning of which would be defined in one of the ##FORMAT lines. 0/0 would represent an unphased genotype (i.e., if the locus is heterozygous, it is unknown which chromosome the variant originated from), the second entry (48) represents the genotype quality, DP is read depth and the second comma-separated values represent haplotype quality

Due to the potentially large size of multi-sample VCF files, they can be compressed using the bgzip program. Indexing of VCF files in order to facilitate faster lookup times using VCFtools can be performed using the tabix program. Both bgzip and tabix are part of the samtools package (Li et al. 2009).

1.4.3. BCF file format

The binary version of the VCF format is the BCF format. It stores all the same information, but is much quicker at extracting information, especially from multiple samples. The BCF file format is commonly used with the bcftools program that is part of the SAMtools (H. Li et al. 2009) package to quickly extract variant information.

1.5. Visualising and storing variant data

After the variants have been called it is useful to have a tool for visualising multiple variants at once. Several tools have been developed for looking at variation information obtained from arrayCGH, such as snoopCGH, a Java-based application that plots probe log intensity against the chromosomes (Almagro-Garcia et al. 2009). A recent application for visualising next generation alignment files in bam and sam format is MagicViewer. This tool enables the visualization of a single alignment file and its reference sequence. It is possible to filter variants using many parameters including quality and variant type. It calls variants using the incorporated Genome Analysis Toolkit (GATK, McKenna et al. 2010) and can also save the calls into VCF format. A limitation of this tool is the lack of ability to read VCF files and to be able to view multiple genomes simultaneously. The Artemis software is an established genome browser that has recently incorporated the ability to read and view VCF data. Some limitations are present in its current form though. First of all, only a single VCF file can be loaded at a time, meaning it is cumbersome to work with multiple samples. Secondly, only a colour, representing the variation type, is present and no function is available to filter variants. It has been reported that the UCSC Genome Browser will soon support the ability to view VCF file information (Fujita et al. 2010). However being an online tool, this may not be suitable for users that need fast browsing of very large VCF file.

1.6. Aims and Objectives

The initial aim of this project is to develop a software tool that can quickly visualise genomic variants. It will be required to read in the popular VCF. It should be capable of visualising multiple genomes, as this will become increasingly important as the sequencing costs per sample decrease and more samples can be compared. The tool should be able to filter variants, based on various quality control criteria. Furthermore, basic statistical analysis should be available to the user such as fixation index (FST) and expected heterozygosity (He) statistics to gain an insight into population differences and allele diversity. A core requirement will be speed of the software.

The second aim of the project will be to utilise the VCF viewer tool to analyse genomic variation in three populations of *P. falciparum* isolates and to catalogue any variation found.

The following is a list of objectives:

- The completion of a previously started desktop genome variation browser
    - Add some basic statistical tests to determine selection pressure at polymorphic loci
    - Create new tracks for displaying data such as variation density/GC content
    - Enable filtering of polymorphisms by several criteria
    - Enhance navigation capabilities
    - Optimise the code to increase browsing speed

- Use the browser to analyse HTS reads from *P. falciparum* isolates
    - Perform basic analysis of sequence reads
    - Identify polymorphisms potentially involved in antibiotic resistance

2. Creation of a variation browser tool

2.1 The preliminary variation browser

A basic variation browser has already been created by Magnus Manske from the Sanger Institute (Cambridge, UK). It takes as input VCF, FASTA reference sequence, and a GFF (v3) annotation file containing the positions of genes, coding regions and other features. The sequence file must be a multiple FASTA file listing each chromosome individually. Current functionality includes the ability to view variation data and basic annotation such as genes, CDS, coding strand, and sequence. Each chromosome is viewed individually, with a drop-down box providing the means to switch chromosome. A control panel on the left hand side of the main window contains basic control and information tools. The browser window contains three tracks: a sequence track that displays the sequence positions, and when zoomed in, the nucleotide identities; an annotation track that shows the genes and constituent CDS sequences as well as other features; and a variation track. In the latter, multiple rows are present, which correspond to individual samples. At each position where there is a variation, a rectangle is painted with the colour providing some information about the nature of the polymorphism. A zoom slider is available as well as the ability to display a region by specifying the range. When the cursor hovers over a polymorphism, the information box displays some information extracted from the VCF file. When a variation position is left-clicked, the information becomes fixed until the 'remove props' button is pressed or another variant is clicked. A scroll bar present at the bottom of the browser window allows navigation across the chromosome (Figure 2.1; Figure 2.2). As no comments or manual were available with varb, a UML diagram was created to provide an overview of the program and to help understand the main classes of the program (Figure 2.3). The program is written in C++ and uses the Qt4 (http://qt.nokia.com/) graphical framework.

Main Varb classes

The *MainWindow* controls the rest of the objects. Upon initialisation, the constructor sets up various settings in the GUI. The data is loaded by the Annotation, Variation, and SequenceData classes. These classes are subclasses of Track, and are part of TrackContainer. TrackContainer is a subclass of QWidget, the base class of all Qt UI objects.

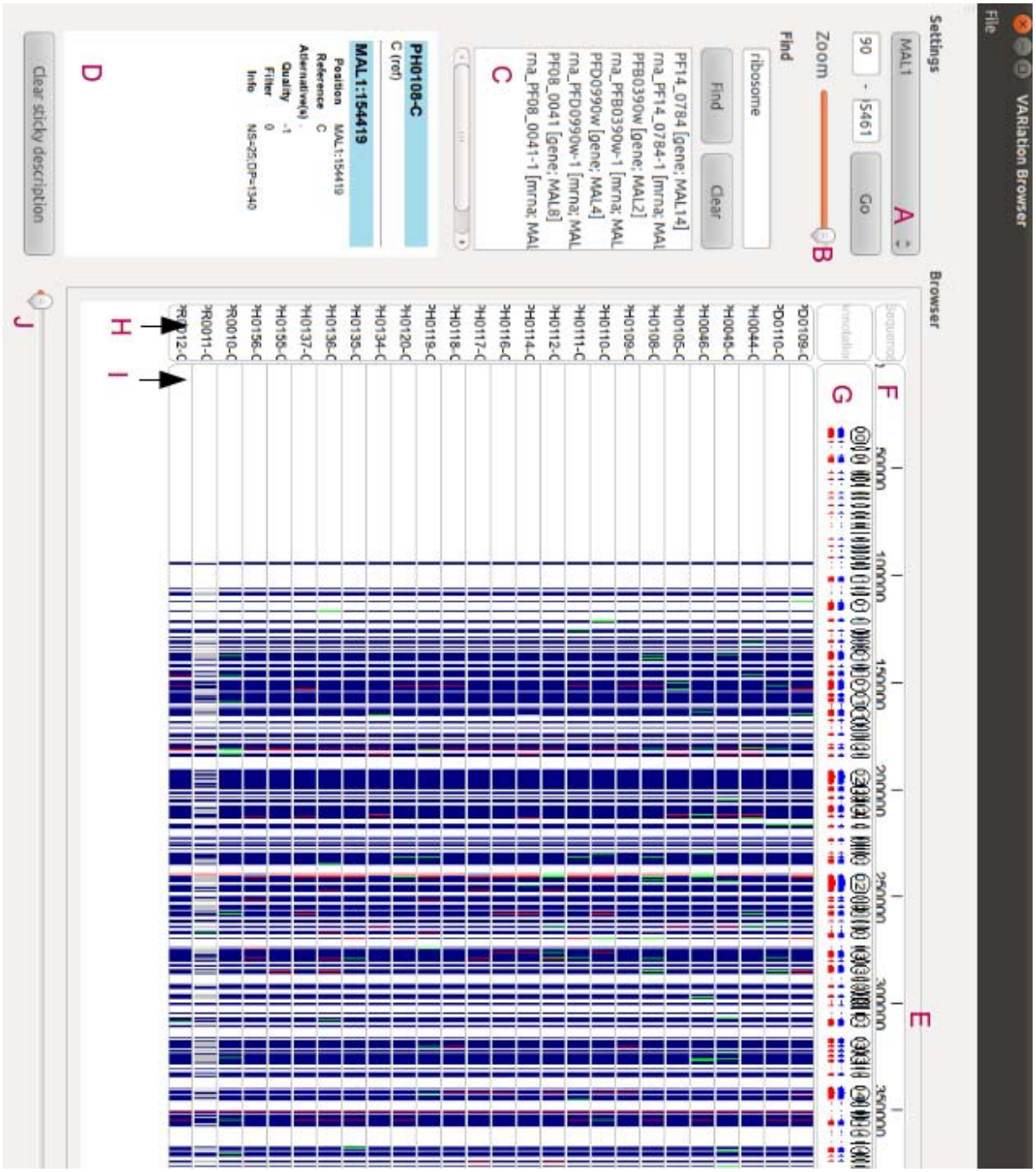


Figure 2.1: Partial display of original Varb. (A) Chromosome selection. (B) Zoom scroller (C) Search results box (D) Information box (E) browser window (F) sequence track (G) annotation track (H) Sample labels (I) variation/polymorpishm positions (J) scroll bar

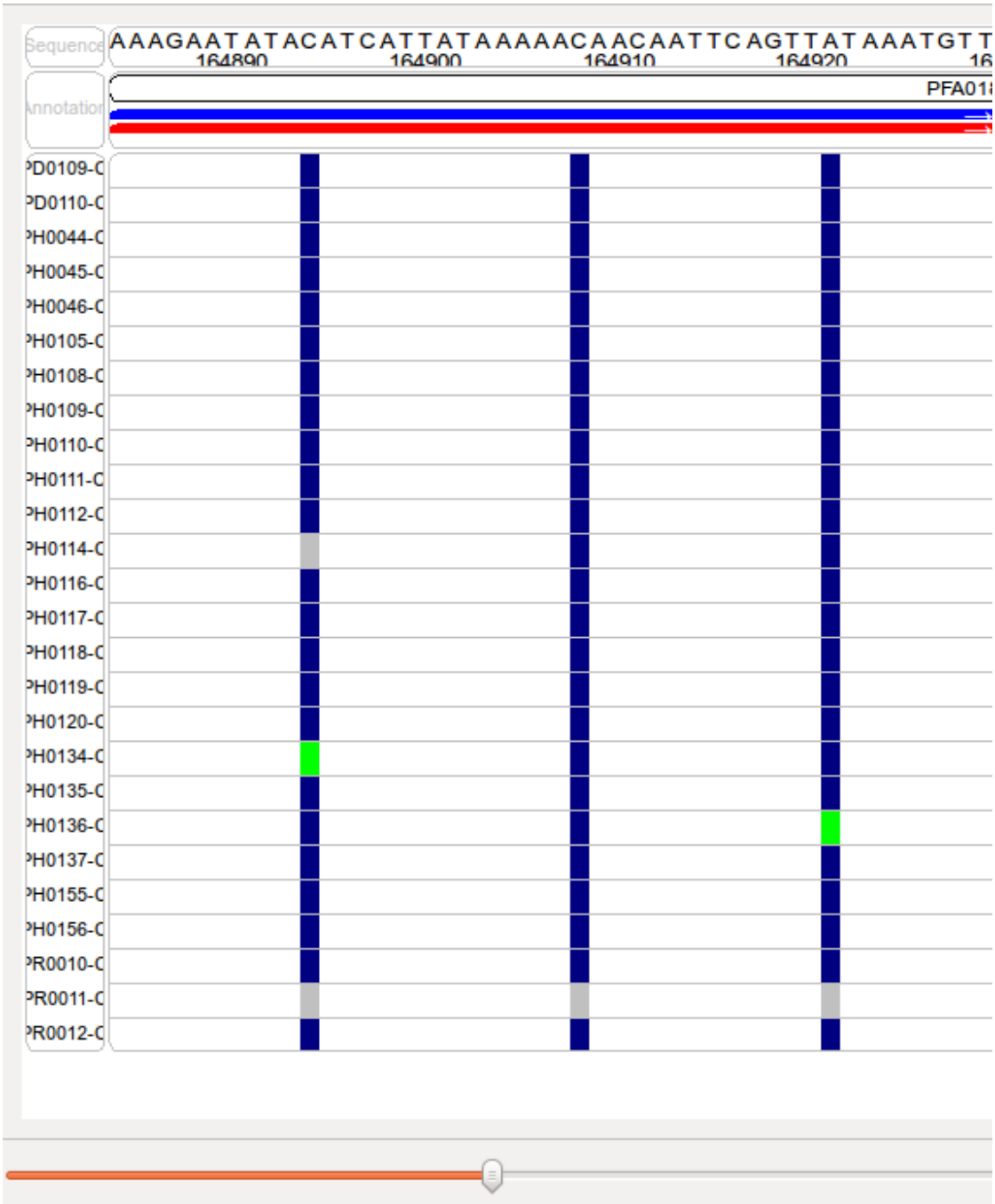


Figure 2.2: Partial display of Varb zoomed in to show individual variant positions

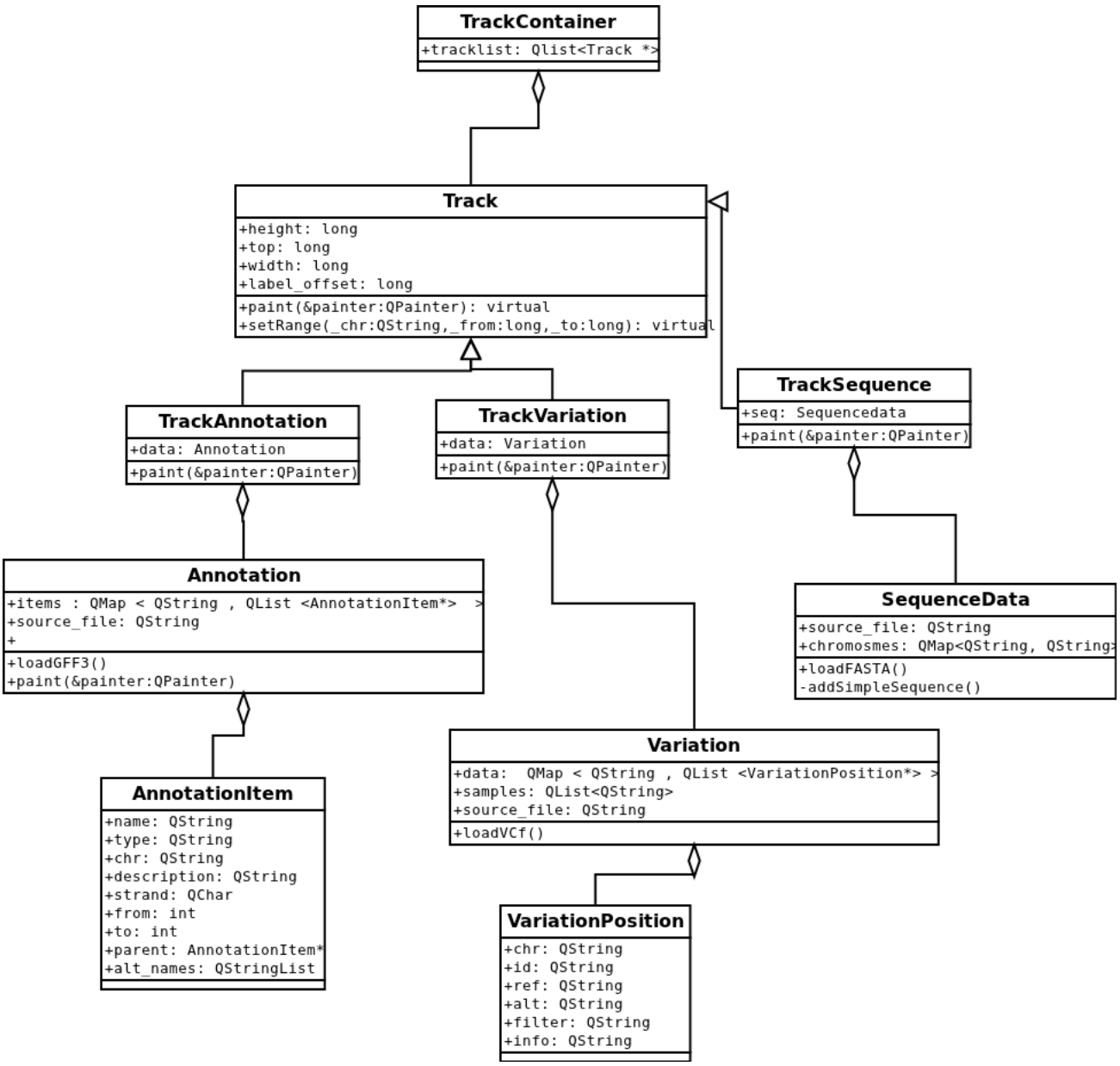


Figure 2.3: Main data structures of Varb

2.2.Modifications to the variation browser

Varb has been modified and from here it will be call VarExplorer. The next sections will describe some of the modifications that have been carried out.

2.2.1. Loading new data

Varb was able to load data only from the file locations hard-coded into the *MainWindow* constructor. *SourcesDialog* provided a means to get new file names, but was unfinished and did not load any new data. In VarExplorer, the *SourcesDialog* can be invoked from the toolbar as well as at the initialisation of the program. This then calls *Mainwindow::loadData().* A dialog appears asking the user if the current project should be saved before loading new data. Any previous data present in the *tracklist* is then deleted by calling *Track::clear()* that deletes all the data objects, freeing up space. An extra field is present in the *SourcesDialog* window that allows the loading of saved project data, the format of which is described later.

2.2.2. The sidebar

The sidebar has been changed to a almost black colour with white text for aesthetic reasons. The output of the find function now opens up a new large dialog, presenting id, description and position in separate columns for ease of viewing rather than in a small text area within the sidebar. Clicking on a result row zooms the browser so that the width visible is 110% larger than the feature. With the find results box removed from the side-bar it was possible to place other buttons and sliders there, the functions of which are described later. As the sidebar takes up a significant amount of the horizontal space of the screen, it can me hidden by unchecking the toolbar *view/sidebar* menu item freeing up more space to view the variation tracks.

2.2.3. Groups

The toolbar *tools/manage samples* menu item opens a dialog that enables the creation of groups that contain samples (Figure 2.4). For example, it can be possible to group together samples that come from specific geographical regions or have certain phenotypic characteristics. Once groups have been selected, selecting from the toolbar *filter/group*s item brings up another dialog that enables groups to be

hidden from view. When samples are hidden, the variation track is automatically repainted to remove gaps. This feature is useful if a large number of samples that have been loaded, and the user would like to concentrate on a select few samples or groups

2.2.4. Filtering

A quality cut-off function has been implemented that uses the quality score from column 6 of the VCF file. A slider on the sidebar allows the setting of the quality, under which value the variation position is not painted. SNPs and indels can be selectively viewed or hidden using checkboxes in the sidebar.

An important property of SNPs is whether they cause a change in the amino acid sequence of the encoded protein (non-synonymous of Ka) or not (synonymous or Ks). Synonymous substitutions can alter the function of a protein and so may be of particular interest to the user. As the abundance of Ks is usually higher than that of Ka, there is the ability to hide or show each category of SNP to enable rapid identification. *MainWindow::on_actionCalculate_non_synonymous_SNPs_triggered()* is activated during *MainWindow* initialisation, which creates a *CdsInfo* object for each gene in the GFF annotation file. The objects are then stored as a public member of *MainWindow* in a QList. Once all CDS have been processed, *TrackVariation*::setSynonymous() is called, which scans the CDSinfo CDS sequences for positions that are non-synonymous. The *VariationPosition* objects within *TrackVariation* then have the bool *isNonSyn* set accordingly. The initial function in this process, *MainWindow::on_actionCalculate_non_synonymous_SNPs_triggered(),* is run on a separate thread, because this process takes longer than the loading of all the other data. This enables the user to start browsing the data as soon as the other data is loaded.

When the non/synonymous SNPs have been calculated the corresponding check boxes become active. For multithreading, the Qt function *QtConcurrent::run(function)* is employed, which runs 'function' in a separate thread when one becomes available (http://doc.qt.nokia.com/latest/threads-qtconcurrent.html).

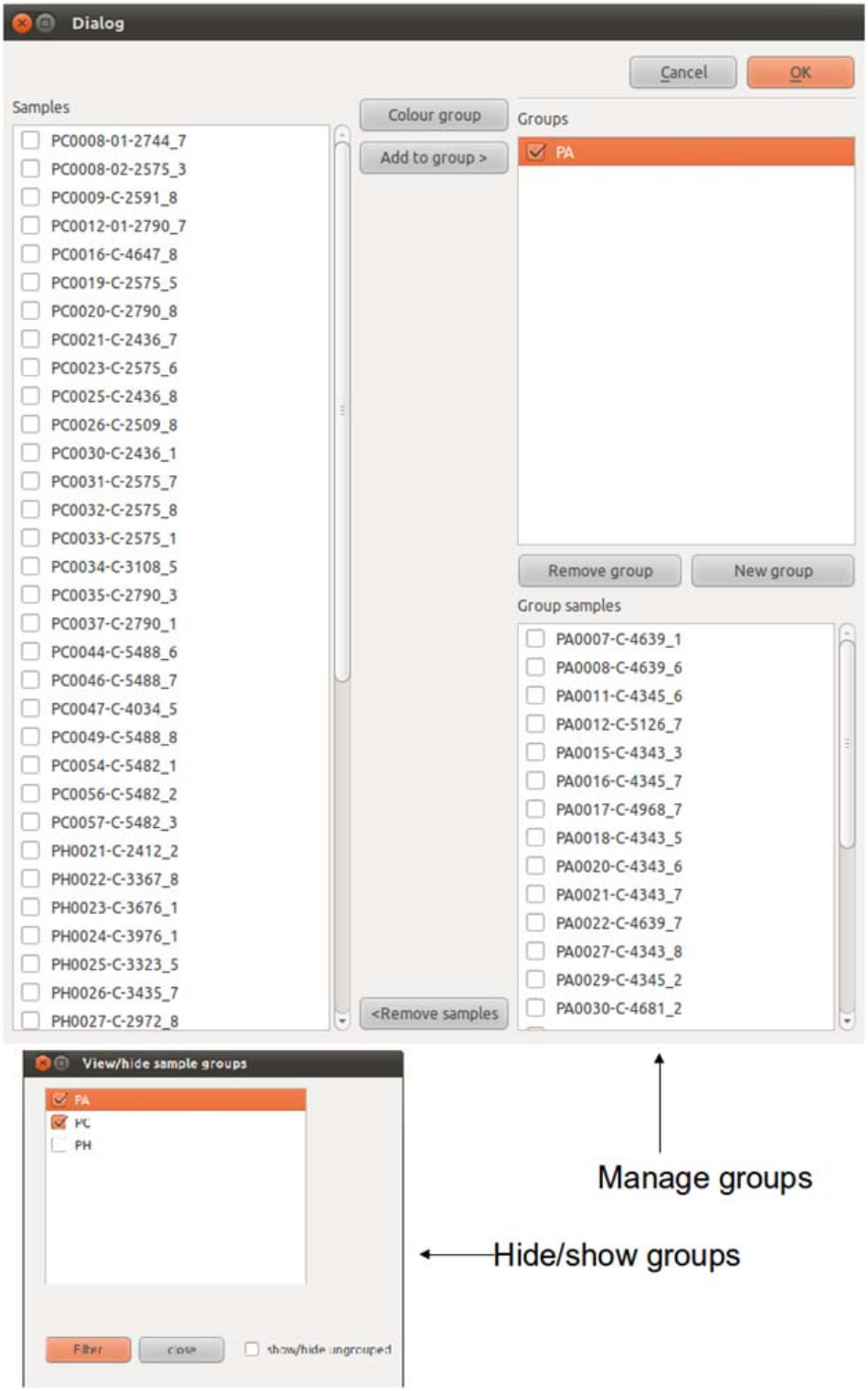


Figure 2.4: Screenshots of the manage groups dialog (above), which allows the creation and editing of sample groups. 'View/hide sample groups dialog' (below) allows fitering of samples based on group membership

2.2.5. Navigation and display

Several minor modifications were carried out to increase the ease of navigation of the data. The sliders that allows for scrolling left and right along the chromosome and zooming have been relocated to beneath the data tracks. They are now larger, which allows for more precise movements. A right click on a variation position now invokes a zooming function. If the users' mouse has a scroll wheel, this can be used to zoom in and out of the data. The original position under the cursor will then appear at the centre of the window.

Significant variants can be saved in order to easily find them at a later time: If a variant position is clicked with the left mouse button, underneath the information box that appears in the side-bar there is a button *Save var*, which opens a dialog where notes can be stored about the variation. The toolbar menu item '*data/show save variants*' displays a dialog that contains a table listing the variants. The variants can then be revisited by selecting and pressing '*go to variant*'.

When a variation position is clicked it will acquire some yellow marks to highlight it. It is then possible to browse to adjacent positions using the left and right keyboard keys.

The genomic locations have at the top of the sequence track have been changed to kb and Mb if the length of the chromosome that is being viewed is larger than a thousand or million base pairs respectively. In addition, upon zooming, when nucleotides become visible, the nucleotide letters are coloured.

The variation track is now vertically scrollable if the amount of samples takes up more space than present in the variation track window.

The size of the lanes on the variation track can be altered by using the adjacent vertical slider. This is useful if the dataset contains many samples.

Finally, the complement of the reference sequence can be painted by activating the toolbar menu 'show ref complement' item. This is useful when looking at reverse orientation CDS regions.

2.2.6. Colouring of genotypes

During the loading of the variation data, the different genotype values present in the VCF file are stored in a QList within the MainWindow. Upon activating the *tools/change variant colouring* option, a dialog appears with a table containing all the gt types present. Clicking on the second column on the table opens a colour-chooser dialog where a colour can be set for each genotype. If this is not chosen, default

colours are used that are hard-coded in *TrackVariation::pain()*.

2.2.7. Custom filtering of variation positions.

The VCF V4 specification (Danecek et al. 2011) allows for the addition of metadata in the header of the file. The #INFO lines in the header specify the type of information found within the #INFO fields with a key/value encoding. The following field for example, *##INFO=<ID=H2,Number=0,Type=Flag,Description="HapMap2 membership*">, defines entries in the file that have an ID of H2, the data is not numerical, and are of type flag meaning the presence of the entry for a variant assigns it the property of the description (membership of Hapmap2 in this case). VarExpoler can scan the VCF header for ##INFO fields, and provides a way of filtering based on the presence or absence of selected tags via an optional side window that can be shown/hidden via the toolbar *view* menu. Other keys that could be present in the VCF header ##INFO fields are integer, float, and string. Fields that contain these keys are not yet able to be used for filtering. Future versions of VarExplorer will be able to use these fields by first finding the range of values for string and float, and the different strings present before filtering can occur via a value slider for numerical fields, and drop-down QComboBoxes for strings.

2.2.8. Extra data tracks

To provide more information for the user, a new window below the main data tracks was created. The new window is a *QscrollArea*, a child of which is *TrackContainer2*, which is a slightly modified *TrackContainer* class present in the original Varb. TrackContainer2 has a public member *Qlist<Track2*>*. Again, *Track2* is a modified class of Track that contains some different virtual functions. *TrackContainer2 receives* the current chromosome, sequence start, and stop positions from *MainWindow::setRange ()*. The extra tracks become visible by clicking the check-box in bottom of the data tracks. Each track can be made separately visible/hidden by using the toolbar menu *tracks*.

2.2.8.1. GC content data track

GC content information can be useful, for example altered GC content could highlight specific chromosomal regions such as centromere and telomere regions. This information is present as a track on many browsers including the UCSC Genome Browser (Zhu et al. 2009), and the Ensembl gemome browser (Kersey et al. 2009).

To create the GC content track, a new *TrackGC* object is initialised in the constructor of *MainWindow* after which the *GcData.loadGC*( ) is called. The GC content is calculated on-the-fly outputting the mean GC content for the pixel window. If zoomed-in beyond a point where a pixel window is < 20, a window size of 20 is used (Figure 2.5).

2.2.8.2. Uniqueness data

When analysing genomic sequence data assembled from small-read data, it can be important to know how unique the region of interest is within the genome. This can give some idea of how reliable the mapping of the sequences are. With a high uniqueness, the user may wish to assign a high probability of the sequence being in the correct position, whereas in a region of low uniqueness the user may wish to be more cautious about the variation positions. The uniqueness data is handled in an almost identical manner to that of the GC data. The main difference being that the data is not loaded at initialisation of *MainWindow*. Instead the data is loaded by selecting the toolbar menu item 'data/load uniqueness data'. This action brings-up a dialog where a file containing uniqueness data can be loaded. This is required to be a three-column, space-delimited, file specifying chromosome id, sequence position, and uniqueness score at each position. The uniqueness data is generated using a script that scans the chromosomes in 50 bp windows, moving along the chromosome one bp at a time. Each window is used to interrogate the rest of the genome for a unique match. A uniqueness score for every position is then calculated by calculating how many of the windows that overlap the position have 100 % matches to other regions. A score of 0 would indicate total uniqueness, whereas a score of 50 indicates low uniqueness and so problems with assembly could have occurred at this region. The Perl script was written by Taane Clark, and will be distributed with the VarExplorer software.

2.2.8.3. Fixation index (F_{ST})

F_{ST} is a metric that measures population differentiation. When employed in a whole genome context, the F_{ST} gives a measure of how much interbreeding there is between different populations. An F_{ST} near 0 would indicate substantial gene flow, between the sub-populations, while a F_{ST} approaching 1 might indicate barriers to interbreeding between populations, such as geographical boundaries. When applied to alleles or haplotypes, F_{ST} can provide information on selection at these loci. Assuming neutrality, F_{ST} between populations is influenced by gene flow and genetic drift, and these processes should effect all loci in a similar manner. Regions of the genome that display F_{ST} values that vary significantly to the background levels van be indicative of selective pressures.

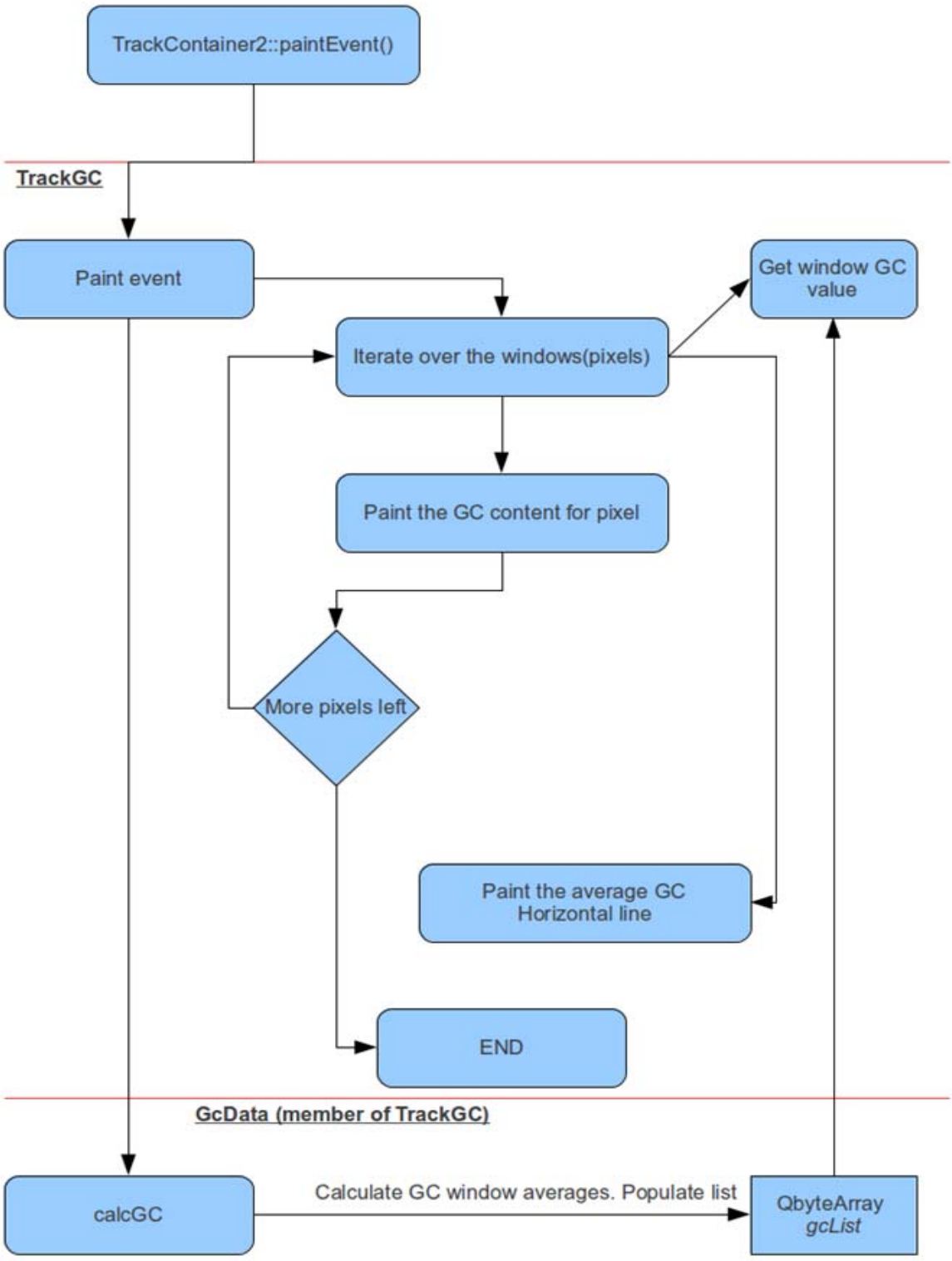


Figure 2.5: Data flow for creating GC content track

Positive selective pressures tends to increase F_{ST} at loci, while balancing or negative selection tends to result in a decrease in F_{ST}. Therefore locating regions of high F_{ST} could be useful in the context of malaria research as these regions could be under positive selection. For example, F_{ST} values at 10 *dhps* loci in *P. falciparum* were significantly higher (F_{ST} = 0.2) than at eight neutral loci (F_{ST} = 0.01)

$$F_{ST} = \frac{\sum n_i (\tilde{p}_i - \bar{p})^2 (r-1) \bar{n}}{\bar{p}(1-\bar{p})}$$

Figure 2.6: FST equation from (Weir 1996). ni sample size; n-bar, average population size; r, number of populations; p-tilde, population major allele frequency; p-bar, mean major allele frequency over all populations

indicating that positive selection had been acting upon them (Vinayak et al. 2010)

The F_{ST} value is calculated as described in Weir (1996) and the equation is shown in Figure 2.6. Before F_{ST} can be calculated in VarExplorer, at least two groups must be created, as described in section 2.2.2. The toolbar menu item 'data/calculate fst' opens a dialog that allows the selection of groups to be included in the analysis. After closing the dialog the F_{ST} is calculated and displayed as a line chart in a new track in the bottom track window. If there is more than one variation position at a pixel location, the variation with the highest F_{ST} is displayed. It was considered to display an average F_{ST} for the pixel window, but as F_{ST} values are often very low, especially for closely-related groups, this would hide significantly high values that may be of interest.

2.2.8.4. Variation density

A variation density track shows the relative abundance of polymorphisms at each pixel window, as often there will be many polymorphisms at each location. The track splits into two separate tracks upon a left mouse click, displaying the relative densities of SNPs and indels.

2.2.9. Encapsulation of data

Encapsulation of data, which is encouraged in C++ and other object-oriented programming languages, is an important way of making software more robust. If an object contains member variables or data

structures that are vital to the proper functioning of the program, any unintended modification could damage the stability of the software or, more importantly, alter data without the users knowledge, providing spurious results. In Varb, the main data structures are declared as public members. For example the *AnnotationItem* objects are stored in a *QMap* as a public member of *Annotation.* The *AnnotationItem* objects should only be modified during initialisation at the point of loading of the data by *Annotation::loadAnnotation()*. Due to its public status, any other object that has access to *Annotation* can modify the contents of *items, f*or example *Mainwindow* and *TrackVariation()*, the latter class accessing *items* from several functions. In order to encapsulate *items*, it was changed into a private member of *Annotation.* An accessor function *Annotation::getAnnItemsPtr()* was created, which returns a *const* pointer to *items*. Declaring a pointer *const* ensures that the callee function cannot modify the object or variable that it points to. While this solution does provide encapsulation of the annotation data, it does not provide a good level of abstraction, with the functions that access the data in items required to know how the data is stored. Future versions of VarExplorer will be modified to provide better data abstraction between classes.

2.2.10. Optimization

As DNA sequencing becomes progressively cheaper, the amount of genome sequences available to researchers will increase resulting in larger datasets, and thus potentially larger VCF files. According to Moore's law, computer processor speed doubles approximately every 18 months (Moore 1998), allowing for programs such as VarExlorer to handle increasingly more samples. However, as new sequencing technologies develop, and existing technologies mature, the price of DNA sequencing is currently reducing at a rate faster than Moore's law (Figure 2.7). Therefore the generation of efficient algorithms and the optimisation of software that handles large amounts of sequencing data is important to ensure the longevity of the software tool.

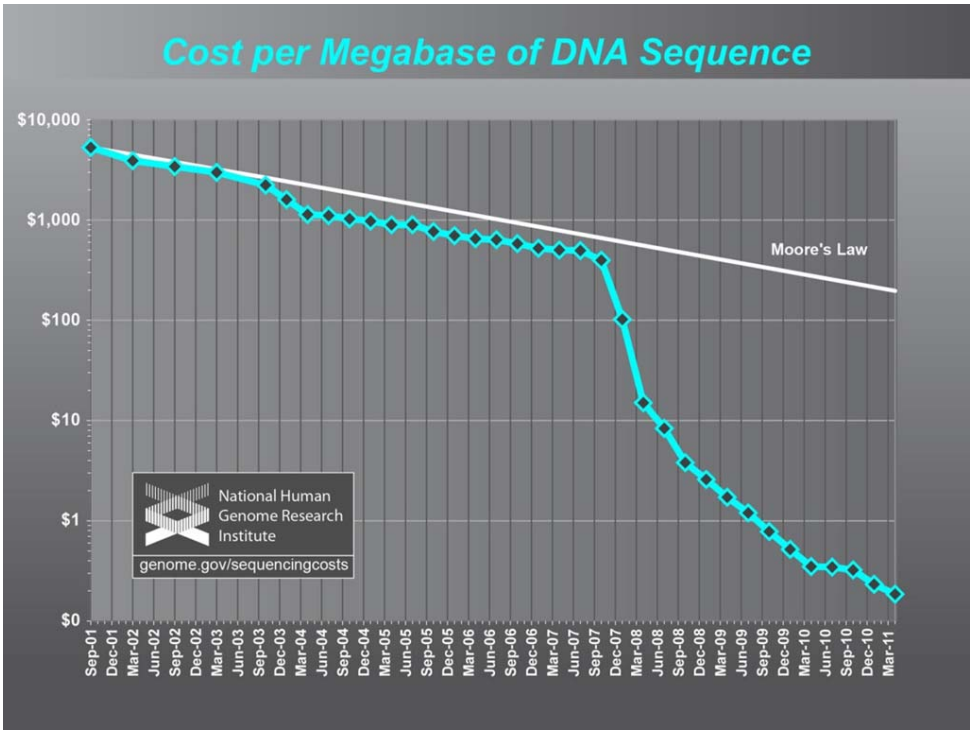


Figure 2.7: Graph showing cost of DNA sequencing per megabase over time. Note the deviation from Moore's law around 2007. From NHGRI website (http://www.genome.gov/)

2.2.11. Profiling

Profiling is the process of analysing the running of a program to measure, for instance memory usage, and the amount of CPU time a specific function consumes. Callgrind, from the Valgrind package (http://valgrind.org), was used to identify functions in VarExplorer that are consuming the most CPU time, and thus are candidates for optimization. Callgrind monitors the progress of the program and creates a call graph of all the functions called, which displays the number of instructions executed, the number of times the function was called and displays the caller/callee relationships. The goal of optimization was to increase the refresh rate of the display upon paint events to make the software more user friendly, rather than increase the initial data loading rate, which only takes a few seconds which is an acceptable length of time. Therefore, The Macro 'CALLGRIND_START_INSTRUMENTATION', which indicates to Callgrind the place to start logging data, was included at the end of *TrackVariation::setSynonymous()*, as at this point all the primary data has been loaded. This enables Callgrind to execute much quicker, as it can otherwise slow down the program dramatically, and also reduces the complexity of the output. *TrackContainer::paintEvent()*

was found to have an inclusive cost of around 75%. An inclusive cost is the CPU cost of this function and all other functions that it may call. Figure 2.8 Shows a partial call graph with the percentage values showing the cost relative to *TrackContainer::paintEvent()* performed on VarExplorer before any optimization. As can be seen, the function taking up most of the resources is *TrackVariation::paint()*, which contributes over 92% of the *TrackContainer::paintEvent()* inclusive cost. Therefore, several modifications were applied to this function in an attempt to optimize the speed. The function *TrackContainer::event()* along with *QMainWindow::event()* were found to be both calling T*rackContainer::paintEvent()*.

TrackContainer::event() was deleted as it appears to be calling unnecessary paint events. The VariationPosition data objects were originally stored as a QMap<QString, Qlist<VariationPosition> within a Variation object. In order to acquire the VariationPosition objects for displaying, TrackVariation::paint() first iterates over the samples and for each sample iterates over the Qlist<VariationPosition> and extracts the position variable from the VariationPosition object. The position is then tested to see whether it falls within the to and from variables that define the visible sequence. If it passes, then its pixel position on the PaintArea is calculated before painting. If another VariationPosition is then found which maps to the same pixel, it is discounted and only the first variation for that pixel is displayed (Figure 2.9a). In VarExplorer, the data is stored in a nested map structure: *QHash<QString, QMap<long, VariationPosition>* with the long variable holding the variant sequence position enabling faster lookup of variants. Acquiring the *VariationPosition* object is performed by iterating over the samples before iterating over each pixel in the width of the *TrackContainer* display. At each pixel, the corresponding genome sequence range is calculated and the first *VariationPosition* within this bin is extracted by key lookup. If there is no *VariationPosition* within the bin, the next pixel is moved onto. If a variant is found the data is extracted and moves straight on to the next pixel (Figure 2.9b). This is a more efficient algorithm as the *VariationPosition* objects can be accessed directly for each pixel vastly minimising the number of instructions that need to be performed

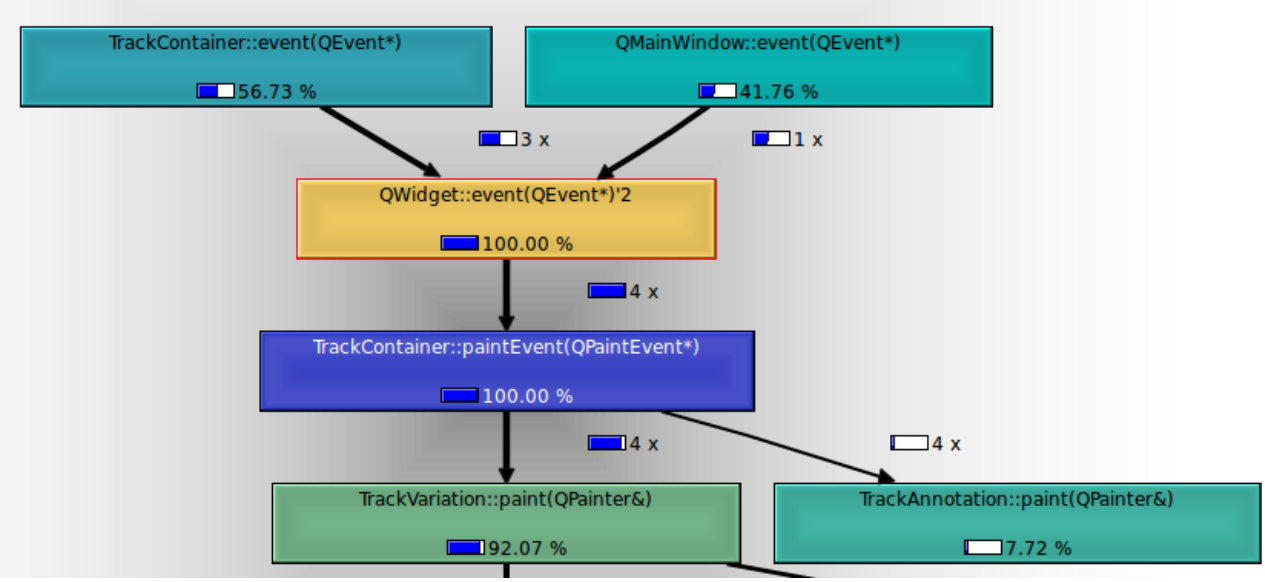


Figure 2.8: A section of a Callgrind-generated call graph showing the functions called upon a paint event using the non-optimized TrackVariation::paint() function. Percentages show inclusive cost of the function

.

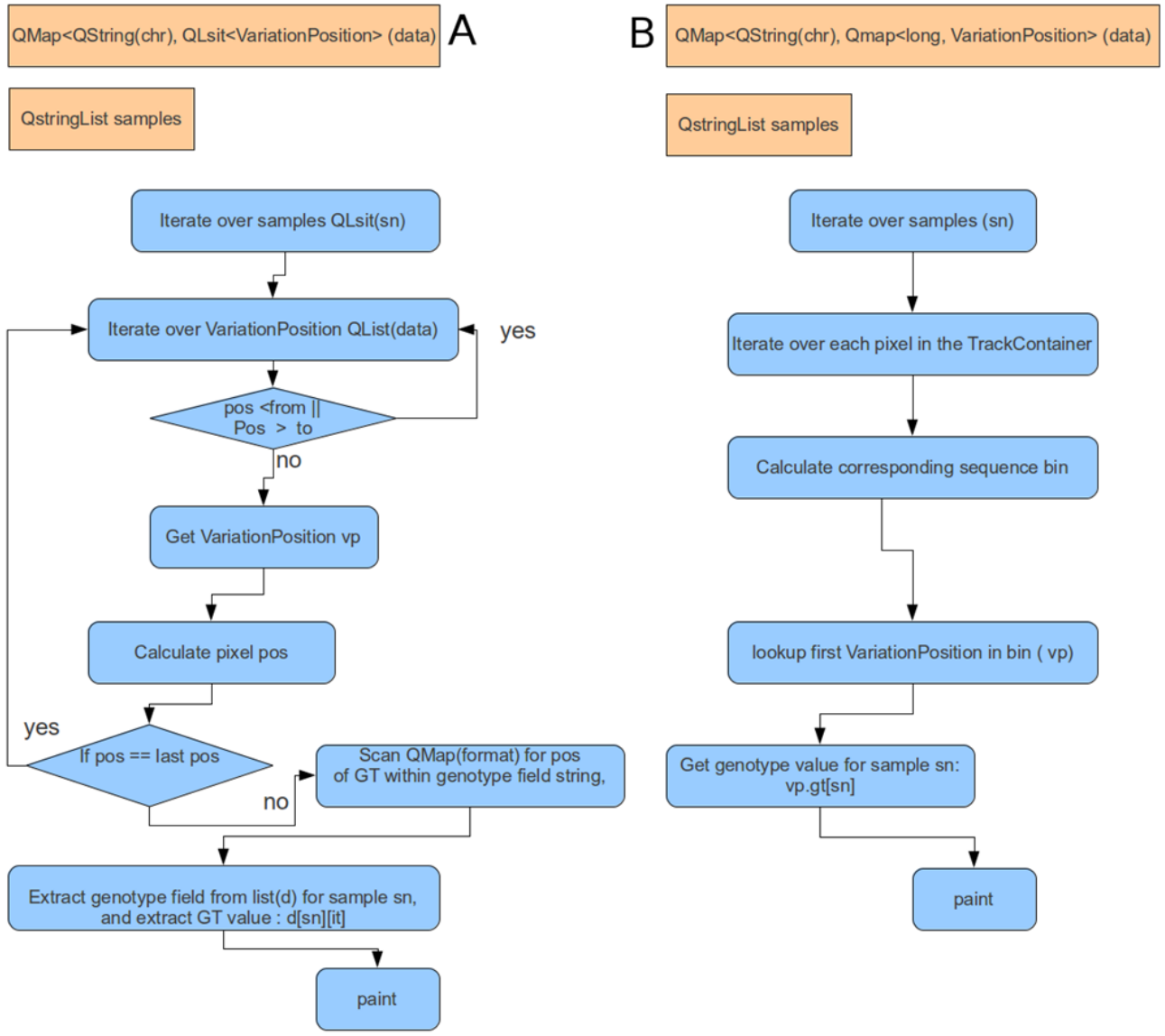


Figure 2.9: (A) Simplified overview of TrackVariation::paint() function of Varb. (B) Simplified overview of modified TrackVariation::paint() in VarExplorer.

2.2.12. Data types

Where possible, the *QMap* container class has been replaced for the *QHash* container. *QHash* provides fast lookup of values, but the data is not ordered (QMaps sort data on keys) so was not suitable for the *Variation* data that requires ordered keys.

A new List containing each genotype numerical code is now stored in a *QstringList* in each *VariationPosition* allowing quicker lookup of the genotype. This is generated at the point of data loading. Previously in Varb, gt value had to be looked up from the genotype string at each paint event.

To further test the capability of VarExplorer, the human reference genome chromosome 22 and VCF file were downloaded from dbSNP (http://www.ncbi.nlm.nih.gov/snp). The VCF files contained data from 120 individuals and 40705 polymorphic positions. This amount of data appears to slow down the workings of VarExplorer somewhat, but it is still usable, with single event zoom-ins taking < 0.5 seconds. This represents a large increase in performance over the original Varb, as using Varb on this dataset results in >15 second lag during zooming and scrolling at the most zoomed out level.

Another improvement was made by replacing all *widget::repaint()* by *widget::update()*. u*pdate()* waits for the program to return to the main loop before repainting, and can merge multiple paint events into one. This optimizes repainting of the widget and can stop flickering, which can occur when multiple paint events are occurring nearly simultaneously (http://doc.qt.nokia.com/latest/qwidget.html#update-2).

2.2.13. Memory usage

It was found that a large amount of memory was used when running large datasets in VarExplorer. For example, when loading *P. falciparum* data from 75 isolates containing 126886 polymorphic positions (VCF file 23 MB), the amount of ram used was around 2.9GB. This will be too high for many users' computers and memory usage must be addressed.

3. Analysis of polymorphism data from three populations of *Plasmodium falciparum*

3.1. Introduction

Three populations of *P. falciparum* were selected for initial analysis using VarExplorer. These were sequenced at the Sanger Institute (Cambridge, UK). The sequencing project website can be located here: http://www.sanger.ac.uk/research/projects/malariaprogramme-kwiatkowski/sequencing.html. This has provided an opportunity to test VarExplorer on a real dataset, enabling the identification of bugs and to provide ideas for new functionality. The dataset consists of a VCF file generated from Illumina sequencing (36 – 76 bp length reads). The sequenced reads were a mixture of single and paired-end reads from three populations containing 25 samples each. The samples originate from blood samples of infected patients. The genome sequences were assembled using a re-sequencing approach, with the reads aligned to the 3D7 genome (probable west African origin). The three sample populations are Gambia (PA), Kenya (PC), and Cambodia (PH). Some basic statistics about the variants were gathered, and several genes that may be involved in antibiotic resistance were identified.

3.1.1. Distribution of SNPs and indels

The *P. falciparum* chromosomes were scanned for the density of indels and SNPs in 1 kb bins to gain an idea of variation density over each chromosome. There was no visible correlation between chromosome position and indel density section (Appendix 7.1.3). In contrast, SNPs were highly concentrated at the terminal regions of all chromosomes, with a noticeable reduction in the density of SNPs at the distal terminal of chromosome 5.The location of high density SNPs at sub-telomeric regions has been previously reported and is associated with regions containing highly variable antigenic families such as PfEMP1 and rifin (Mu et al. 2007). High densities of SNPs were also observed internally of the subtelomeric regions in chromosomes 4, 6, 7, 8, and 12 (Appendix7.1.2) and these regions corresponded to internal regions of antigenic-related genes as has been noted previously (Gardner et al. 2002). The majority of indels were small, the vast majority being one and two bp indels. The largest insertion was 10 bp and the largest deletion was 9 bp (Figure 3.1)

The distribution of variants between the gene-coding regions and intergenic regions was determined as a frequency per kb of gene-coding and intergenic sequence respectively. The frequency of indels across the genome was quite consistent for intergenic indels between the chromosomes (mean 3.33 indels /kb,

sd. 0.22), and also for within-gene indels (mean 0.24 indels / kb, sd. 0.03). The SNP densities however, were more variable across the chromosomes for both intergenic SNPs (7.93 \kb, sd. 3.47) and within-gene SNPS (2.36 \ kb , sd. 1.65)(Figure 3.2). Correlation between the within-gene SNP density and the intergenic SNP density was very high ($R^2 = 0.94$) whereas there was no correlation between the within-gene indel density and the intergenic indel density ($R^2 = 0.01$) (Figure 3.3). This suggests that selective forces driving the formation of SNPs and indels is different.

3.1.2. Non-synonymous SNPs

SNPs that change the amino acid sequence of genes that they occur in are potentially important sites as they can contribute directly to different phenotypes. Table 3.1 shows the number of positions that contain at least one non-synonymous or all synonymous substitutions on each of the 14 chromosomes. Non-synonymous substitutions are more abundant on each chromosome.

3.1.3. Variants that introduce premature stop codons

Indels within gene-coding regions that introduce a stop codon or alter the reading frame, as well as SNPs that introduce premature stop codons (PSC) can have a dramatic effect on the coding protein. Therefore a whole-genome scan was performed to identify all PSCs formed by SNPs (PSCs formed by frameshift mutations were not searched for) in the hope of finding genes that were probably non-functional and so are candidates for genes that are involved in immune recognition. There were 79 genes that had new stop codons within open-reading frames relative to the 3D7 reference genome, and 20 of these shared the same annotation and were part of multi-gene families. The vast majority are annotated as being involved in host immune evasion, and virulence, such as PfEMP1, rifin, and stevor (

Table 3.2). Seven of the PSC genes are annotated as pseudogenes, and as such are likely non-functional, and have probably accrued further stop codons due to loss of functional constraint acting on them. Ten PSC genes were annotated as having unknown function; seven of these showed elevated expression in an RNA-seq expression analysis (data from PlasmDB) from intraerythrocytic parasites (Appendix 7.1.4).

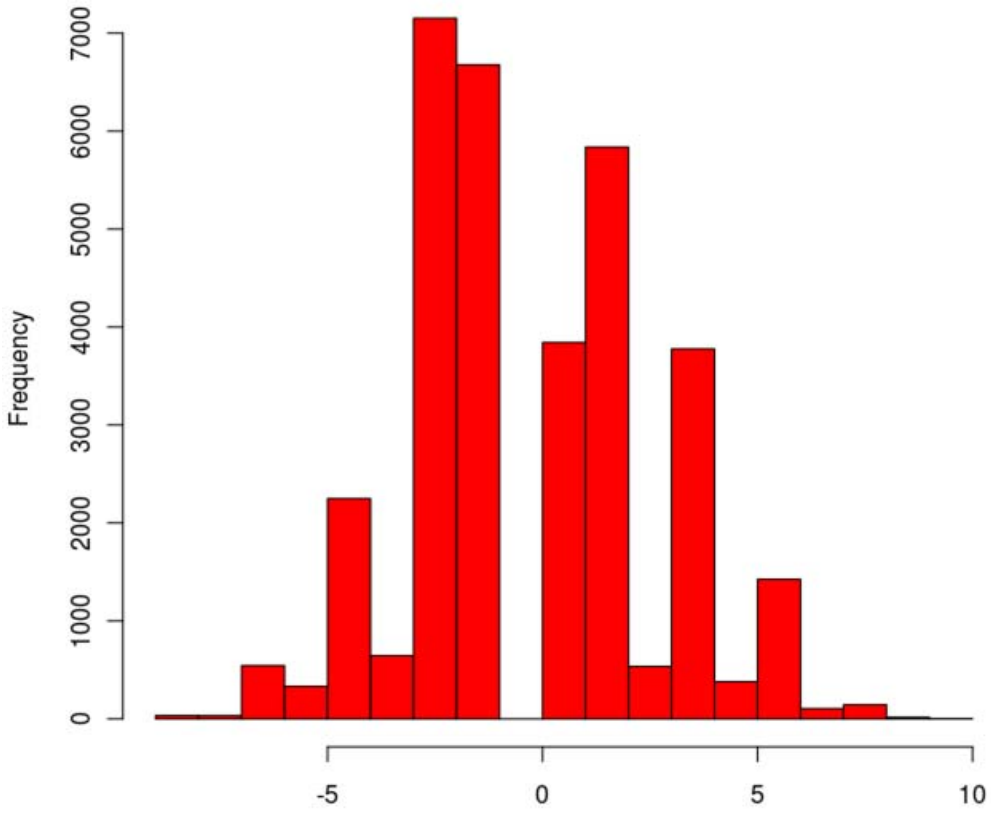


Figure 3.1:Indel size frequencies relative to the 3D7 genome from all samples

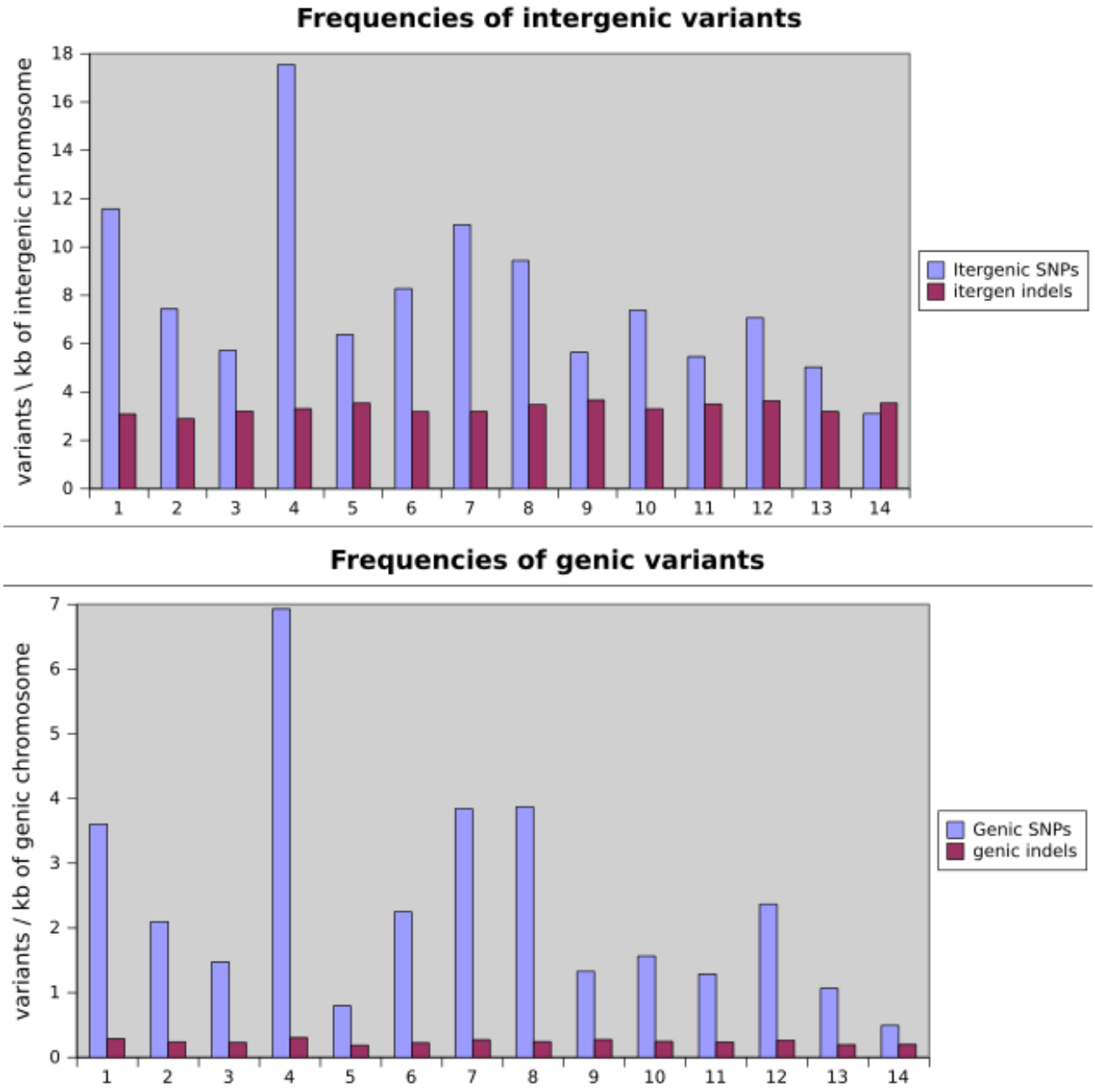


Figure 3.2: Frequency of polymorphisms within each chromosome for intergenic and within-gene regions

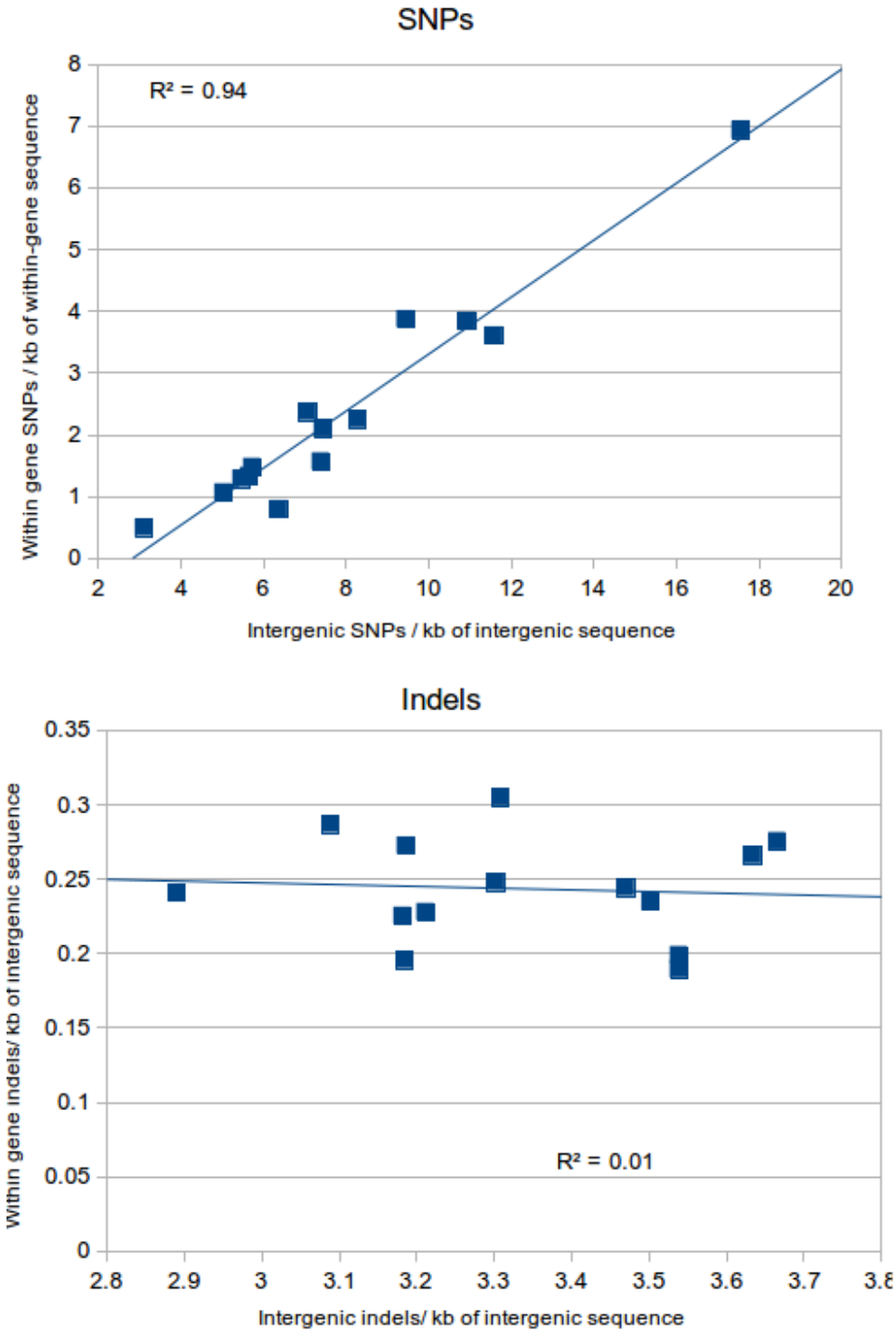


Figure 3.3: Correlation between variation density within gene-coding and intergenic regions

Table 3.1: Data for each chromosome describing the number of polymorphic sites that contain either all silent or one or more non-synonymous substitution

chr	**All synonymous substitutions**	**>0 non-synonymous substitutions**
MAL1	393	1186
MAL2	335	1037
MAL3	302	800
MAL4	3711	5017
MAL5	256	732
MAL6	967	1836
MAL7	2813	3597
MAL8	1766	4028
MAL9	716	1348
MAL10	918	1804
MAL11	879	1893
MAL12	1409	3868
MAL13	1074	2140
MAL14	498	961
Total	16037	30247

Table 3.2: Genes and genes families identified as containing nonsense SNPs

Gene annotation	**Number of genes**
conserved Plasmodium falciparum protein, unknown function, pseudogene	1
conserved Plasmodium protein, unknown function	10
cytoadherence linked asexual protein 3.2	1
erythrocyte membrane protein 1 (PfEMP1)-like protein	1
erythrocyte membrane protein 1 (PfEMP1), exon2, pseudogene	4
erythrocyte membrane protein 1 (PfEMP1), pseudogene	1
erythrocyte membrane protein 1, PfEMP1	40
hypothetical protein, pseudogene	1
Pfmc-2TM Maurer's cleft two transmembrane protein	1
Plasmodium exported protein, unknown function, pseudogene	1
RESA-like protein	1
rifin	10
rifin, pseudogene	3
stevor	1
stevor, pseudogene	1
surface-associated interspersed protein 1.3 (SURFIN 1.3)	1
var-like erythrocyte membrane protein 1	1

3.1.4. Polymorphisms with a potential role in antibiotic resistance

Using VarExplorer, the *P. falciparum* dataset was searched for polymorphisms in known antibiotic resistance genes. The following genes that are known to be involved in antibiotic resistance contained no polymorphic sites within the coding regions. *PfATPase6, pfmrp1*, *pfmdr1, pfhdr, pfdhps*, *pfgch1, pfmrp2, pfcmu, and pfnhe-1.* However, the chloroquine transporter gene (*Pfcrt)* was found to have one synonymous mutation in exon 4 that is predicted to create a I194T substitution (Figure 3.4A). This substitution was found previously in a study of Cambodian *P. falciparum isolates,* but was not found to be associated with an altered chloroquine resistance phenotype (Durrand et al. 2004). In the current dataset, this mutation was found only in the Cambodian isolate PH0024-C. A non-synonymous SNP was also present in the gene encoding the chloroquine resistance marker protein (*Pfcrmp*) creating the substitution M1957I (Figure 3.4B). This polymorphism appeared to have gone almost to fixation in the Cambodian population, with only one isolate not harbouring the SNP. It was less prevalent in the African populations, with around two thirds of the isolates containing the SNP. This mutation was not found in the literature or within the PlasmoDB genome browser.

Two in-frame deletions and an insertion were located in PFD0965w, which encodes a putative phophatidylinositol 4-kinase (PI4-K). All the variants are in a region that appears to be unique to the *P. falciparum* PI4-K that is not present in the *Plasmodium knowlesi* and *Plasmodium vivax* homologs (Figure 3.5). The two deletions both deleted an asparagine codon, while the insertion created a new asparagine codon. It has been reported that artemisinin and related compounds can target PI3-Ks in human cell lines (Xu et al. 2007). It may also be possible that they target the related PI4-K proteins. Most of these indels occurred in the African isolates, with only one isolate from Cambodia containing one of the indels.

A single base pair insertion was found directly after the start codon of PFE1355c, which encodes a putative ubiquitin carboxyl-terminal hydrolase. It was present in six of the Kenyan and two of the Gambian isolates. The insertion creates a frame-shift that introduces a stop codon two codons downstream. Therefore this mutation likely renders this gene non-functional (Figure 3.6). Mutations occurring in the homolog rodent-infecting rodent malaria homolog of PFE1355c have been suggested to be involved in resistance to artemisinin (Hunt et al. 2007).

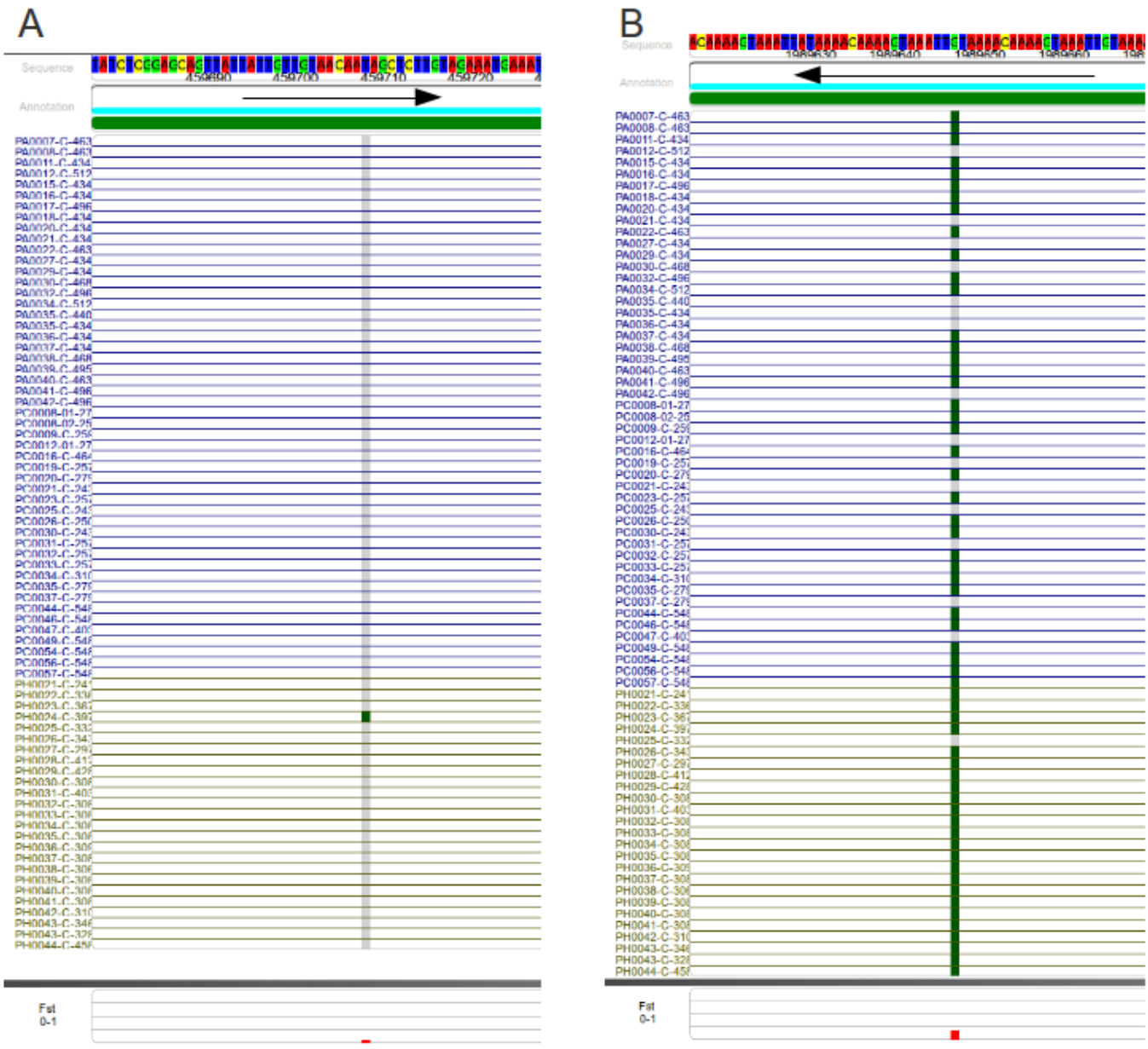


Figure 3.4: Screenshot of VarExplorer showing mutations potentially affecting chloroquine resistance. Blue samples- African, Green samples-Cambodia. Grey variant-reference, green variant-alternative (A) SNP in PfCrt creating a I194T substitution. (B) SNP in Pfcrmp1 creating a M1957I substitutions

Figure 3.5: Screenshot of VarExplorer showing mutations in PI4-K. (A) position of indels. (B) alignment showing regions of similarity between Plasmodium homologs

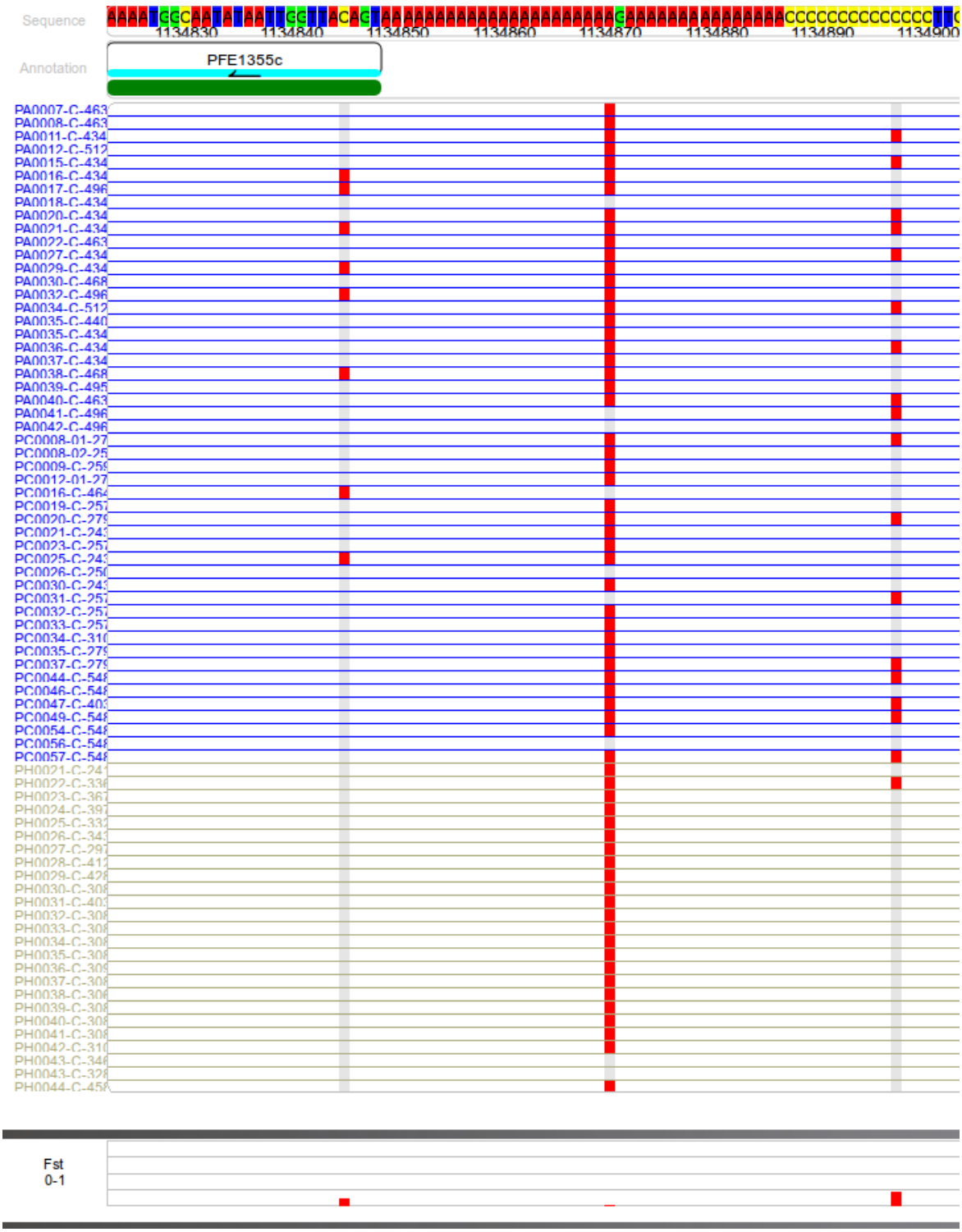


Figure 3.6: Screenshot of VarExplorer showing mutations in PFE1355c, putative ubiquitin carboxyl hydrolase coding region and 5' region

4. Discussion

VarExplorer is designed to be a tool primarily for browsing and not a tool for detailed analysis. While not as feature-rich as some programs it does provide unique functionality that will hopefully be useful amongst the microbial community and other fields. There are many other tools available that can perform advanced statistical analysis such as Bioconductor packages (http://www.bioconductor.org/), view alignments (MagicViewer, Hou *et al*. 2010; IGV, Robinson *et al*. 2007), and call variants (SAMtools, Li et al. 2009), for example, and for this reason, VarExplorer was not designed to perform these functions. VarExplorer will be useful to users who do not have a lot of experience in bioinformatics and require an easy to use program.

VarExplorer was trialled on a VCF file generated from 75 *P. falciparum* isolates and several polymorphisms were identified that may have implications for antibiotic resistance.

4.1. Proposed added functionality for VarExplorer

The ability to generate summary information for the user would be useful. For example, a report could be invoked that detailed the whole-genome polymorphism count, the number of synonymous and non-synonymous substitutions and other relevant information in a tab-delimited format, which could be easily used for further analysis. Additionally, it should be possible to bring up on the screen lists of genes that are ranked according to some of the scores of the statistical tests or other metrics such as SNP density or allele frequency from across the whole genome. This would allow the user to quickly identify regions of interest without having to manually inspect each chromosome individually. Another feature will be the ability to save alignments in, clustal or similar formats, for use in downstream applications such as phylogenetics.

One of the strengths of VarExplorer is the ability to view many variants from multiple samples over an entire chromosome simultaneously. This however poses the problem of how to convey meaningful information when taking a broad view at a large region that contains multiple polymorphisms per horizontal pixel. Each horizontal pixel in the variation tracks presently displays the colour associated with the genotype of the first polymorphism present within the pixel window, thereby potentially

masking much salient information. One approach that was considered to overcome this limitation was to compare the similarity of data within each pixel window with the corresponding window of every other sample, and colour by similarity. This would provide readily visible information of large scale differences between samples as opposed to just in comparison to the reference genome. Similarity could be inferred by calculating pairwise distances between each sample at each pixel window, providing a means to cluster the samples, and could be accomplished using the Bio++ (Dutheil et al. 2006) C++ libraries. However this is likely to be CPU-expensive and may not be feasible for a real-time genome browser. Another option would be to display two points within each horizontal pixel window whose heights corresponds to SNP and indel density within the window, thereby forming a line chart-like representation along each sample lane showing SNP and indel similarity to the reference genome. While this approach would not provide as much information regarding similarity to other samples, it would give more information than at present on how similar this region is to the reference genome. This approach would also not be largely more CPU-intensive than the currently implemented method.

F_{ST} was chosen as the first statistical test to include in VarExplorer due to its ubiquitous use within the microbial genetics and epidemiological communities. It has been implemented to allow the user to determine the level of differentiation between populations of each polymorphism. Other statistical tests will be included in future version of the program, including linkage disequilibrium (LD) and Tajimas D. Tajima's D statistic is used to identify regions of genomic DNA that are evolving in a non-neutral manner, and provides extra information on top of the F_{ST} value as this test is performed on the whole sample set without the prior requirement of creating sample groups (Tajima 1989). The ability to perform LD would provide further useful information, in particular LD is sensitive to selective sweeps that occur in a population that reduces variation in a region that has been subject to recent strong selective pressure (Robbins 1918; McVean 2006), which could include genes such as those involved antibiotic resistance or virulence.

4.1.1. Optimisation of VarExplorer

Although VarExplorer has been optimised somewhat during the project, there is still scope for significant improvement. For example, it has been shown that VarExplorer can load a VCF file derived from human chromosome 22 samples. While this chromosome data is loaded, around 1.1 GB of memory was used. In the current form of the software, in order to be able to load a full complement of human chromosomes or other large-genome organism, it would be necessary to load data from each

chromosome separately, and when the user switched chromosomes, the new data would be parsed and loaded into memory. Future versions of the program could utilise indexed VCF files that would allow for fast random access to data without having to load the whole file into memory, thereby allowing the analysis of any data regardless of genome size or number of samples in the experiment. To improve performance further, the facility to use indexed BCF files, the binary version of VCF, could be introduced. The use of BCF files could be implemented by incorporating the SAMtools binary or source code (Li et al. 2009) into the VarExplorer package and using its BCF-reading capabilities, but this could limit the usage of this functionality to Unix-type computers. However, it should be trivial to create a BCF binary parser.

4.1.2. Alterations to the software architecture

Many changes could be applied to the current code to make it more easily maintained and robust. One way is to encapsulate of as much data as possible to prevent unintended modification. Creating interfaces for classes, that hide the inner workings of the classes in order to create abstraction between the classes. This has already been started as described in 2.2.9, where the annotation data stored in the *Annotation* object was encapsulated. This also needs to be performed for the sequence data and the variation data, and several other public members in the *MainWindow*. Also the software could be redesigned somewhat to fit into an established software architectural pattern, such as model-view-controller (MVC) pattern (Curry & Grace 2008). The current software was designed in a somewhat ad hoc way without paying a lot of attention to following established design patterns. Following an established pattern allows future programmers to become quickly familiar with the software if they are already familiar with the specific design pattern. Also a design pattern will have been optimised over many years to efficiently solve many of the problems that may occur. Designing software using an MVC strategy attempts to separate the software components into three separate entities: The model is the part of the code that manages the state of the application data. It provides information about the state of the data to the view component and changes the state of the data when requested by the controller component; The view component obtains information from the model and displays it to the user; The controller component accepts input from the user interface and signals the model to alter its state accordingly. Using this system, the three components should be unaware of the internal working of the other two components. This abstraction allows for self-contained software components that can be more easily maintained, as well as being suitable for software reuse. For example, in well-designed MVC software, the user interface can easily be replaced with a new one, all that is required is to know

how that new UI must interface with the model, without having to know anything about its internal workings.

VarExplorer could have been written in Java, as many bioinformatics applications are, instead of C++. One advantage of using Java would have been the ability to use its mature biological libraries such as BioJava. This would be useful for adding further functionality. Although writing programs in more high-level languages such as Perl or Python is generally quicker, the application will generally run much more slowly, which is not suitable for a tool such as VarExplorer where real-time viewing of large amounts of data are required. In order to access the rapid development time and excellent biological libraries of Perl and Python would be to call scripts written in these languages as long as the computations were not too intensive. An alternative development model to the one used here, could have been to create the main project in Python using PyQt4, which is a set of Python bindings for the Qt graphical framework. The computationally intensive parts of the program, such as painting the variants could then have been performed by C++ code. This approach could greatly increase development time, especially for non-expert C++ programmers, without much performance cost.

As the typical user of VarExplorer may be dealing with multiple VCF files from multiple experiments, a feature of VarExplorer in the future may be to incorporate some form of database management of the VCF file. One approach would be to use the SQLite (http://www.sqlite.org/) database binary, which due to its small size (less than 300 KB) could be easily packaged with VarExplorer. SQLite is serverless and requires no configuration and so does not require installation on the users' computer. The SQLite database could be used for storing VCF files, as well as project xml files

VarExplorer was trialled on a subset of the *P. falciparum* data that is currently available. The whole dataset was not used because the memory requirements were too high. As mentioned previously, future versions of VarExplorer will be able to load chromosome data individually thereby increasing the amount of samples that are able to be loaded. Another approach to decrease the memory cost of VarExplorer could be to read in data on-the-fly from a binary BCF file. This would require an extensive reworking of the data structures that currently store the variation data, as *VariationPosition* objects that currently hold data from the VCF file also contain other data such as whether a SNP is a synonymous substitution or not. Having to calculate this information on every repaint event could be costly. Using VarExplorer on a real dataset highlighted some areas that could be improved. One such area was variant gene context: When looking for specific variations in a gene, it would make the task much

easier if the translated CDS was visible underneath the DNA sequence track, along with numbers indicating the amino acid position in the protein.

4.2. Analysis of *P. falciparum* polymorphisms

To trial VarExplorer on some real data, a VCF file created form sequencing of 75 Pf isolates and was loaded into VarExplorer along with the 3D7 reference genomic FASTA file and a GFF gene annotation file.

The initial analysis of the *P. falciparum* genomic data found that there is a lot of variation between isolates. The vast majority of the variation was at subtelomeric regions of the chromosomes as has been described previously (Gardner et al. 2002). The polymorphisms in these regions were not looked at in any detail as genes in these are known to be hyper-variable between isolates. It was decided instead to look for novel variations in genes that may have a role in antibiotic resistance. The identification of novel mutations that confer antibiotic resistance could inform public policy regarding antibiotic regimes in various regions.

The majority of indels were 1-2 bp in size with the largest insertions and deletions being 10 bp and 9 bp respectively (Figure 3.1). A previous study has shown that larger indels within different populations of *P. falciparum* are common (Hawkins et al. 2008). Therefore it is possible that many more larger indels were not included in the assembly of the genomes, possibly due to limitations of the short read length resequencing strategy used. However some larger indels, if present, would have been expected to have been discovered using the paired-end data that was used to generate some of the assemblies.

There was little variation of indel frequencies across chromosomes within gene-coding regions. Similarly there was little variation between chromosomes in intergenic regions. In contrast, there was much variation in SNP frequency between chromosomes and a high correlation of SNP frequency between gene-coding regions and intergenic regions. Therefore it is possible that the SNP-induced mutations are under different selective pressures than the indel mutations, and further analysis will be needed to characterise this.

As well as mutation creating I194T substitution in the chloroquine transporter protein (PfCrt) that had already been reported and was found to have no effect in antibiotic resistance, another polymorphic site was found in the coding sequence of the chloroquine resistance marker protein (Pfcrmp). The presence

of indels in the coding gene has been linked to a reduction in chloroquine sensitivity. The function of the protein is currently unknown, but it shares homology with DNA binding proteins and contains nuclear localisation signals (Li 2008).

In six African isolates, insertions were found in a putative ubiquitin carboxyl-terminal hydrolase (CTH) gene (PFE1355c), which introduced a premature stop codon into the coding region. CTH has been implicated in artemisinin resistance in the rodent pathogen, *Plasmodium chabaudi.* Two artensuate-resistant strains were independently created by exposing parasites to multiple rounds of passaging in the presence of sub-lethal concentrations of artemisinin. In both strains, resistance was mapped to a region on chromosome 2 that contained non-synonymous substitutions in a ubiquitin carboxyl-terminal hydrolase (Hunt et al. 2007). The authors suggest, but have not verified, that these mutations were contributing to antibiotic resistance, and that CTH-mediated alterations of the ubiquitination status of the Pfmdr1 multi-drug resistance protein may be mediating the phenotype. Targeted mutations of this gene could uncover any role in antibiotic resistance.

The idea that the mutations in PI4-K are involved in antibiotic resistance is speculative. Artemisinin and related drugs target PI3-Ks in human cell lines (Xu et al. 2007). It may also be possible that they target the related PI4-K proteins. Most of these indels occurred in the African isolates, with only one isolate from Cambodia containing one of the indels. However artemisinin resistance has currently not been observed in Africa, but has recently been found in the Cambodia-Thai border region (Noedl et al. 2008).

5. Conclusion

The initial goal of the current project was to look at genetic variation between populations of *P. falciparum* now that large amounts of sequence data has become publicly available. This led to the idea of developing a software tool that was capable of handling a large number of samples quickly and would facilitate the identification of variants. The product of this, VarExplorer, has already fulfilled many of the requirements; it is a fast polymorphism browser that enables the simultaneous viewing of multiple samples, has the ability to create sample groups and perform basic statistical analysis, and finally save the details of variants of interest. Filtering, based on several criteria, has also been implemented to facilitate rapid identification of variants of interest. User customisation is a feature that has started to be implemented with the ability to colour variants and groups to suit a user's needs. Future versions of VarExplorer will include the following improvements, amongst others, in order to further enhance the usefulness of the software:

- Produce reports of summary data
- Save alignments to file
- Alter the display of variations when zoomed-out
- Include more statistical test (e.g. Tajima's D and LD)
- Ability to load indexed BCF files
- Save sessions to SQLite database

The usefulness of VarExplorer was ascertained by using it for a preliminary analysis on HTS genome sequencing data from 75 isolates of *P. falciparum*. Several polymorphisms were identified that may possibly have a role in antibiotic resistance, as identified from literature searches. A more thorough analysis of the data will possibly identify more significant polymorphisms. The current project only focused on 75 isolates due to limited time available to produce the VCF files. Future work could include the whole dataset, which currently numbers around 370 (http://www.sanger.ac.uk/research/projects/malariaprogramme-kwiatkowski/sequencing.html). The acquisition and use of antibiotic resistance in the isolates data may also be useful in identifying polymorphisms involved in antibiotic resistance.

6. References

Abu-Raddad LJ, Patnaik P, and Kublin James G. 2006. Dual Infection with HIV and Malaria Fuels the *Spread of Both Diseases in Sub-Saharan Africa. Science **314**: 1603 -1606.*

Adl SM, Simpson AGB, Farmer MA, Andersen RA, Anderson OR, Barta JR, Bowser SS, Brugerolle G, Fensome RA, Fredericq S, et al. 2005. The new higher level classification of eukaryotes with emphasis on the taxonomy of protists. *J. Eukaryot. Microbiol* **52**: 399-451.

Almagro-Garcia J, Manske M, Carret C, Campino S, Auburn S, MacInnis BL, Maslen G, Pain A., Newbold CI, Kwiatkowski DP, et al. 2009. SnoopCGH: software for visualizing comparative genomic hybridization data. *Bioinformatics* **25**: 2732-2733.

Alonso D, Bouma MJ, and Pascual M. 2011b. Epidemic malaria and warmer temperatures in recent decades in an East African highland. *Proc. Biol. Sci* **278**: 1661-1669.

Altshuler. 2010. Integrating common and rare genetic variation in diverse human populations. *Nature* **467**: 52-58.

Asawamahasakda W, Ittarat I, Pu YM, Ziffer H, and Meshnick SR. 1994. Reaction of antimalarial endoperoxides with specific parasite proteins. *Antimicrob Agents Chemother* **38**: 1854-1858.

Barnwell JW, Asch AS, Nachman RL, Yamaya M, Aikawa M, and Ingravallo P. 1989. A human 88-kD membrane glycoprotein (CD36) functions in vitro as a receptor for a cytoadherence ligand on Plasmodium falciparum-infected erythrocytes. *J. Clin. Invest.* **84**: 765-772.

Berendt AR, Simmons DL, Tansey J, Newbold CI, and Marsh K. 1989. Intercellular adhesion molecule-1 is an endothelial cell adhesion receptor for Plasmodium falciparum. *Nature* **341**: 57-59.

Boeva V, Zinovyev A, Bleakley K, Vert J-P, Janoueix-Lerosey I, Delattre O, and Barillot E. 2011. Control-free calling of copy number alterations in deep-sequencing data using GC-content normalization. *Bioinformatics* **27**: 268-269.

Cheeseman IH, Gomez-Escobar N, Carret CK, Ivens A, Stewart LB, Tetteh KK, and Conway DJ. 2009. Gene copy number variation throughout the Plasmodium falciparum genome. *BMC Genomics* **10**: 353.

Cheng C, Ho WE, Goh FY, Guan SP, Kong LR, Lai W-Q, Leung BP, and Wong WSF. 2011. Anti-malarial drug artesunate attenuates experimental allergic asthma via inhibition of the phosphoinositide 3-kinase/Akt pathway. *PLoS ONE* **6**: e20932.

Conrad DF, Bird C, Blackburne B, Lindsay S, Mamanova L, Lee C, Turner DJ, and Hurles ME. 2010. Mutation spectrum revealed by breakpoint sequencing of human germline CNVs. *Nat Genet* **42**: 385-391.

Crabb BS, and Cowman AF. 2002. Plasmodium falciparum virulence determinants unveiled. *Genome Biol* **3**: reviews1031.1-reviews1031.4.

Curry E, and Grace P. 2008. Flexible Self-Management Using the Model-View-Controller Pattern. *IEEE Software* **25**: 84-90.

Dahlstrom S, Veiga MI, Martensson A, Bjorkman A, and Gil JP. 2009. Polymorphism in PfMRP1 (Plasmodium falciparum Multidrug Resistance Protein 1) Amino Acid 1466 Associated with Resistance to Sulfadoxine-Pyrimethamine Treatment. *Antimicrob. Agents Chemother.* **53**: 2553-2556.

Daines B, Wang H, Li Y, Han Y, Gibbs R, and Chen R. 2009. High-Throughput Multiplex Sequencing to Discover Copy Number Variants in Drosophila. *Genetics* **182**: 935-941.

Danecek P, Auton A, Abecasis G, Albers CA, Banks E, Depristo MA, Handsaker RE, Lunter G, Marth GT, Sherry ST, et al. 2011. The variant call format and VCFtools. *Bioinformatics* **27**: 2156-2158.

Duraisingh MT, Voss TS, Marty AJ, Duffy MF, Good RT, Thompson JK, Freitas-Junior LH, Scherf A, Crabb BS, and Cowman AF. 2005. Heterochromatin Silencing and Locus Repositioning Linked to Regulation of Virulence Genes in Plasmodium falciparum. *Cell* **121**: 13-24.

Durand PM, Oelofse AJ, and Coetzer TL. 2006. An analysis of mobile genetic elements in three Plasmodium species and their potential impact on the nucleotide composition of the P. falciparum genome. *BMC Genomics* **7**: 282.

Durrand V, Berry A, Sem R, Glaziou P, Beaudou J, and Fandeur T. 2004. Variations in the sequence and expression of the Plasmodium falciparum chloroquine resistance transporter (Pfcrt) and their relationship to chloroquine resistance in vitro. *Mol. Biochem. Parasitol* **136**: 273-285.

Dutheil J, Gaillard S, Bazin E, Glémin S, Ranwez V, Galtier N, and Belkhir K. 2006. Bio++: a set of C++ libraries for sequence analysis, phylogenetics, molecular evolution and population genetics. *BMC Bioinformatics* **7**: 188.

Eastman RT, Dharia NV, Winzeler EA, and Fidock DA. 2011. Piperaquine Resistance is Associated with a Copy Number Variation on Chromosome 5 in Drug-Pressured Plasmodium falciparum Parasites. *Antimicrob. Agents Chemother.* AAC.01793-10.

Foote SJ, Thompson JK, Cowman AF, and Kemp DJ. 1989. Amplification of the multidrug resistance gene in some chloroquine-resistant isolates of P. falciparum. *Cell* **57**: 921-930.

Fu W, Zhang F, Wang Y, Gu X, and Jin L. 2010. Identification of copy number variation hotspots in human populations. *Am. J. Hum. Genet* **87**: 494-504.

Fujita PA, Rhead B, Zweig AS, Hinrichs AS, Karolchik D, Cline MS, Goldman M, Barber GP, Clawson H, Coelho A, et al. 2010. The UCSC Genome Browser database: update 2011. *Nucleic Acids Research*. http://nar.oxfordjournals.org/content/early/2010/10/18/nar.gkq963.abstract (Accessed September 7, 2011).

Gallup JL, and Sachs JD. 2001. The economic burden of malaria. *Am. J. Trop. Med. Hyg* **64**: 85-96.

Gardner MJ, Hall N, Fung E, White O, Berriman M, Hyman RW, Carlton JM, Pain Arnab, Nelson KE, Bowman S, et al. 2002. Genome sequence of the human malaria parasite Plasmodium falciparum. *Nature* **419**: 498-511.

Gatton ML, Peters JM, Fowler EV, and Cheng Q. 2003. Switching rates of Plasmodium falciparum var genes: faster than we thought? *Trends Parasitol* **19**: 202-208.

Greenwood BM, Fidock DA, Kyle DE, Kappe SHI, Alonso PL, Collins FH, and Duffy PE. 2008. Malaria: progress, perils, and prospects for eradication. *Journal of Clinical Investigation* **118**: 1266-1276.

Gregson A, and Plowe Christopher V. 2005. Mechanisms of Resistance of Malaria Parasites to Antifolates. *Pharmacological Reviews* **57**: 117 -145.

Gupta R, Nagarajan A, and Wajapeyee N. 2010. Advances in genome-wide DNA methylation analysis. *BioTechniques* **49**: iii-xi.

Hastings P, Lupski James R, Rosenberg SM, and Ira G. 2009. Mechanisms of change in gene copy number. *Nat Rev Genet* **10**: 551-564.

Hastings PJ, Ira G, and Lupski James R. 2009. A Microhomology-Mediated Break-Induced Replication Model for the Origin of Human Copy Number Variation. *PLoS Genet* **5**: e1000327.

Hawkins VN, Auliff A, Prajapati SK, Rungsihirunrat K, Hapuarachchi HC, Maestre A, O'Neil MT, Cheng Q, Joshi H, Na-Bangchang K, et al. 2008. Multiple origins of resistance-conferring mutations in Plasmodium vivax dihydrofolate reductase. *Malar. J* **7**: 72.

Hou H, Zhao F, Zhou L, Zhu E, Teng H, Li X, Bao Q, Wu J, and Sun Z. 2010. MagicViewer: integrated solution for next-generation sequencing data visualization and genetic variation detection and annotation. *Nucleic Acids Research* **38**: W732-W736.

Hunt P, Afonso A, Creasey A, Culleton R, Sidhu ABS, Logan J, Valderramos SG, McNae I, Cheesman S, do Rosario V, et al. 2007. Gene encoding a deubiquitinating enzyme is mutated in artesunate- and chloroquine-resistant rodent malaria parasites. *Mol. Microbiol* **65**: 27-40.

Kaur K, Jain M, Reddy RP, and Jain R. 2010. Quinolines and structurally related heterocycles as antimalarials. *European Journal of Medicinal Chemistry* **45**: 3245-3264.

Kersey PJ, Lawson D, Birney E, Derwent PS, Haimel M, Herrero J, Keenan S, Kerhornou A, Koscielny G, Kahari A, et al. 2009. Ensembl Genomes: Extending Ensembl across the taxonomic space. *Nucleic Acids Research* **38**: D563-D569.

Kidgell C, Volkman SK, Daily J, Borevitz JO, Plouffe D, Zhou Y, Johnson JR, Le Roch K, Sarr O, Ndir O, et al. 2006. A systematic map of genetic variation in Plasmodium falciparum. *PLoS Pathog* **2**: e57.

Kooij TWA, Carlton JM, Bidwell SL, Hall N, Ramesar J, Janse CJ, and Waters AP. 2005. A Plasmodium Whole-Genome Synteny Map: Indels and Synteny Breakpoints as Foci for Species-Specific Genes. *PLoS Pathog* **1**: e44.

Korbel JO, Urban AE, Affourtit JP, Godwin B, Grubert F, Simons JF, Kim PM, Palejev D, Carriero NJ, Du L, et al. 2007. Paired-End Mapping Reveals Extensive Structural Variation in the Human Genome. *Science* **318**: 420 -426.

Li G-D. 2007. Plasmodium falciparum chloroquine resistance marker protein (Pfcrmp) may be a chloroquine target protein in nucleus. *Med. Hypotheses* **68**: 332-334.

Li H, Handsaker B, Wysoker A, Fennell T, Ruan J, Homer N, Marth G, Abecasis G, and Durbin R. 2009. The Sequence Alignment/Map format and SAMtools. *Bioinformatics* **25**: 2078-2079.

Lupski J R, Wise CA, Kuwano A, Pentao L, Parke JT, Glaze DG, Ledbetter DH, Greenberg F, and Patel PI. 1992. Gene dosage is a mechanism for Charcot-Marie-Tooth disease type 1A. *Nat. Genet* **1**: 29-33.

McKenna A, Hanna M, Banks E, Sivachenko A, Cibulskis K, Kernytsky A, Garimella K, Altshuler D, Gabriel S, Daly M, et al. 2010. The Genome Analysis Toolkit: a MapReduce framework for analyzing next-generation DNA sequencing data. *Genome Res* **20**: 1297-1303.

McVean G. 2007. The Structure of Linkage Disequilibrium Around a Selective Sweep. *Genetics* **175**: 1395 -1406.

Mileyko Y, Joh RI, and Weitz JS. 2008. Small-scale copy number variation and large-scale changes in gene expression. *Proceedings of the National Academy of Sciences* **105**: 16659 -16664.

Miller LH, Baruch DI, Marsh Kevin, and Doumbo Ogobara K. 2002. The pathogenic basis of malaria. *Nature* **415**: 673-679.

Moore GE. 1998. Cramming More Components Onto Integrated Circuits. *Proceedings of the IEEE* **86**: 82-85.

Mu J, Awadalla P, Duan J, McGee KM, Keebler J, Seydel K, McVean GAT, and Su X. 2007. Genome-wide variation and identification of vaccine targets in the Plasmodium falciparum genome. *Nat Genet* **39**: 126-130.

Nair S, Miller B, Barends M, Jaidee A, Patel J, Mayxay M, Newton P, Nosten F, Ferdig MT, and Anderson TJC. 2008. Adaptive Copy Number Evolution in Malaria Parasites. *PLoS Genet* **4**: e1000243.

Niang M, Yan Yam X, and Preiser PR. 2009. The Plasmodium falciparum STEVOR Multigene Family Mediates Antigenic Variation of the Infected Erythrocyte. *PLoS Pathog* **5**: e1000307.

Noedl H, Se Y, Schaecher K, Smith BL, Socheat D, and Fukuda MM. 2008. Evidence of artemisinin-resistant malaria in western Cambodia. *N. Engl. J. Med.* **359**: 2619-2620.

Ockenhouse CF, Tegoshi T, Maeno Y, Benjamin C, Ho M, Kan KE, Thway Y, Win K, Aikawa M, and Lobb RR. 1992. Human vascular endothelial cell adhesion receptors for Plasmodium falciparum-infected erythrocytes: roles for endothelial leukocyte adhesion molecule 1 and vascular cell adhesion molecule 1. *The Journal of Experimental Medicine* **176**: 1183 -1189.

Ollomo B, Durand P, Prugnolle F, Douzery E, Arnathau C, Nkoghe D, Leroy E, and Renaud F. 2009. A New Malaria Agent in African Hominids. *PLoS Pathog* **5**: e1000446.

Pang AW, MacDonald JR, Pinto D, Wei J, Rafiq MA, Conrad DF, Park H, Hurles ME, Lee C, Venter JC, et al. 2010. Towards a comprehensive structural variation map of an individual human genome. *Genome Biol* **11**: R52.

Pelak K, Shianna KV, Ge D, Maia JM, Zhu M, Smith JP, Cirulli ET, Fellay J, Dickson SP, Gumbs CE, et al. 2010. The Characterization of Twenty Sequenced Human Genomes. *PLoS Genet* **6**: e1001111.

Plowe C V, Kublin J G, and Doumbo O K. 1998. P. falciparum dihydrofolate reductase and dihydropteroate synthase mutations: epidemiology and role in clinical resistance to antifolates. *Drug Resist. Updat* **1**: 389-396.

Preechapornkul P, Imwong M, Chotivanich K, Pongtavornpinyo W, Dondorp AM, Day NPJ, White NJ, and Pukrittayakamee S. 2009. Plasmodium falciparum pfmdr1 Amplification, Mefloquine Resistance, and Parasite Fitness. *Antimicrob. Agents Chemother.* **53**: 1509-1515.

Price RN, Uhlemann A-C, Brockman A, McGready R, Ashley E, Phaipun L, Patel R, Laing K, Looareesuwan S, White NJ, et al. 2004. Mefloquine resistance in Plasmodium falciparum and increased pfmdr1 gene copy number. *Lancet* **364**: 438-447.

Redon R, Ishikawa S, Fitch KR, Feuk L, Perry GH, Andrews TD, Fiegler H, Shapero MH, Carson AR, Chen W, et al. 2006. Global variation in copy number in the human genome. *Nature* **444**: 444-454.

Reese MG, Moore B, Batchelor C, Salas F, Cunningham F, Marth GT, Stein L, Flicek P, Yandell M, and Eilbeck K. 2010. A standard variation file format for human genome sequences. *Genome Biol* **11**: R88.

Ribacke U, Mok BW, Wirta V, Normark J, Lundeberg J, Kironde F, Egwang TG, Nilsson P, and Wahlgren M. 2007. Genome wide gene amplifications and deletions in Plasmodium falciparum. *Mol. Biochem. Parasitol* **155**: 33-44.

Robbins RB. 1918. Some Applications of Mathematics to Breeding Problems Iii. *Genetics* **3**: 375 -389.

Robinson JT, Thorvaldsdottir H, Winckler W, Guttman M, Lander ES, Getz G, and Mesirov JP. 2011. Integrative genomics viewer. *Nat Biotech* **29**: 24-26.

Scott MP. 2007. Developmental genomics of the most dangerous animal. *Proceedings of the National Academy of Sciences* **104**: 11865 -11866.

Sidhu ABS, Uhlemann A-C, Valderramos SG, Valderramos J-C, Krishna S, and Fidock DA. 2006. Decreasing pfmdr1 copy number in plasmodium falciparum malaria heightens susceptibility to mefloquine, lumefantrine, halofantrine, quinine, and artemisinin. *J. Infect. Dis* **194**: 528-535.

Skinner TS, Manning LS, Johnston WA, and Davis TME. 1996. In vitro stage-specific sensitivity of Plasmodium falciparum to quinine and artemisinin drugs. *International Journal for Parasitology* **26**: 519-525.

Snow RW, Guerra CA, Noor AM, Myint HY, and Hay SI. 2005. The global distribution of clinical episodes of Plasmodium falciparum malaria. *Nature* **434**: 214-217.

Solinas-Toldo S, Lampel S, Stilgenbauer S, Nickolenko J, Benner A, Döhner H, Cremer T, and Lichter P. 1997. Matrix-based comparative genomic hybridization: biochips to screen for genomic imbalances. *Genes Chromosomes Cancer* **20**: 399-407.

Speicher M. 2009. *Vogel and Motulsky's Human Genetics Problems and Approaches*. Springer, New York :

Stranger BE, Forrest MS, Dunning M, Ingle CE, Beazley C, Thorne N, Redon R, Bird CP, de Grassi A, Lee C, et al. 2007. Relative Impact of Nucleotide and Copy Number Variation on Gene Expression Phenotypes. *Science* **315**: 848 -853.

Tajima F. 1989. Statistical Method for Testing the Neutral Mutation Hypothesis by DNA Polymorphism. *Genetics* **123**: 585 -595.

Trenholme KR, Gardiner DL, Holt DC, Thomas EA, Cowman AF, and Kemp DJ. 2000. clag9: A cytoadherence gene in Plasmodium falciparum essential for binding of parasitized erythrocytes to CD36. *Proceedings of the National Academy of Sciences* **97**: 4029 -4033.

Tuzun E, Sharp AJ, Bailey JA, Kaul R, Morrison VA, Pertz LM, Haugen E, Hayden H, Albertson D, Pinkel D, et al. 2005. Fine-scale structural variation of the human genome. *Nat Genet* **37**: 727-732.

Uhlemann A-C, Cameron A, Eckstein-Ludwig U, Fischbarg J, Iserovich P, Zuniga FA, East M, Lee A, Brady L, Haynes RK, et al. 2005. A single amino acid residue can determine the sensitivity of SERCAs to artemisinins. *Nat. Struct. Mol. Biol* **12**: 628-629.

Urban AE, Korbel JO, Selzer R, Richmond T, Hacker A, Popescu GV, Cubells JF, Green R, Emanuel BS, Gerstein MB, et al. 2006. High-resolution mapping of DNA copy alterations in human chromosome 22 using high-density tiling oligonucleotide arrays. *Proceedings of the National Academy of Sciences of the United States of America* **103**: 4534 -4539.

Vinayak S, Alam MT, Mixson-Hayden T, McCollum AM, Sem R, Shah NK, Lim P, Muth S, Rogers WO, Fandeur T, et al. 2010. Origin and Evolution of Sulfadoxine Resistant Plasmodium falciparum. *PLoS Pathog* **6**: e1000830.

Weir BS. 1996. *Genetic Data Analysis 2: Methods for Discrete Population Genetic Data*. 2 Sub. Sinauer Associates Inc.

Xu H, He Y, Yang X, Liang L, Zhan Z, Ye Y, Yang X, Lian F, and Sun L. 2007. Anti-malarial agent artesunate inhibits TNF-alpha-induced production of proinflammatory cytokines via inhibition of NF-kappaB and PI3 kinase/Akt signal pathway in human rheumatoid arthritis fibroblast-like synoviocytes. *Rheumatology (Oxford)* **46**: 920-926.

Ye K., Schulz MH, Long Q, Apweiler R, and Ning Z. 2009. Pindel: a pattern growth approach to detect break points of large deletions and medium sized insertions from paired-end short reads. *Bioinformatics* **25**: 2865-2871.

Yoon S, Xuan Z, Makarov V, Ye Kenny, and Sebat J. 2009. Sensitive and accurate detection of copy number variants using read depth of coverage. *Genome Research*. http://genome.cshlp.org/content/early/2009/08/05/gr.092981.109.abstract (Accessed April 28, 2011).

Zhu J, Sanborn JZ, Benz S, Szeto C, Hsu F, Kuhn RM, Karolchik D, Archie J, Lenburg ME, Esserman LJ, et al. 2009. The UCSC Cancer Genomics Browser. *Nat Meth* **6**: 239-240.

7. Appendix

7.1. Instructions to run VarExplorer

Installation of Qt Creator allows the installation of all the required Qt libraries. Loading the varb.pro file into Qt Creator then running the program by pressing the green arrow will compile and run VarExplorer. An executable binary is also present in the sources folder that has been tested on Ubuntu 11.04. VarExplorer has also been successfully compiled on Mac OS, but has not yet been tested on a Microsoft Windows system. However as Qt is a cross-platform framework, the program should successfully compile on Windows.

7.1.1. Example VCF file

(a) VCF example

Header

```
##fileformat=VCFv4.1
##fileDate=20110413
##source=VCFtools
##reference=file:///refs/human_NCBI36.fasta
##contig=<ID=1,length=249250621,md5=1b22b98cdeb4a9304cb5d48026a85128,species="Homo Sapiens"
##contig=<ID=X,length=155270560,md5=7e0e2e580297b7764e31dbc80c2540dd,species="Homo Sapiens"
##INFO=<ID=AA,Number=1,Type=String,Description="Ancestral Allele">
##INFO=<ID=H2,Number=0,Type=Flag,Description="HapMap2 membership">
##FORMAT=<ID=GT,Number=1,Type=String,Description="Genotype">
##FORMAT=<ID=GQ,Number=1,Type=Integer,Description="Genotype Quality">
##FORMAT=<ID=DP,Number=1,Type=Integer,Description="Read Depth">
##ALT=<ID=DEL,Description="Deletion">
##INFO=<ID=SVTYPE,Number=1,Type=String,Description="Type of structural variant">
##INFO=<ID=END,Number=1,Type=Integer,Description="End position of the variant">
```

Body

#CHROM	POS	ID	REF	ALT	QUAL	FILTER	INFO	FORMAT	SAMPLE1	SAMPLE2
1	1	.	ACG	A,AT	40	PASS	.	GT:DP	1/1:13	2/2:29
1	2	.	C	T,CT	.	PASS	H2;AA=T	GT	0|1	2/2
1	5	rs12	A	G	67	PASS	.	GT:DP	1|0:16	2/2:20
X	100	.	T	<DEL>	.	PASS	SVTYPE=DEL;END=299	GT:GQ:DP	1:12:.	0/0:20

(b) SNP

Alignment

```
1234
ACGT
ATGT
 ^
```

VCF representation

POS	REF	ALT
2	C	T

(c) Insertion

```
12345
AC-GT
ACTGT
  ^
```

POS	REF	ALT
2	C	CT

(d) Deletion

```
1234
ACGT
A--T
 ^^
```

POS	REF	ALT
1	ACG	A

(e) Replacement

```
1234
ACGT
A-TT
 ^^
```

POS	REF
1	ACG

(f) Large structural variant

Alignment

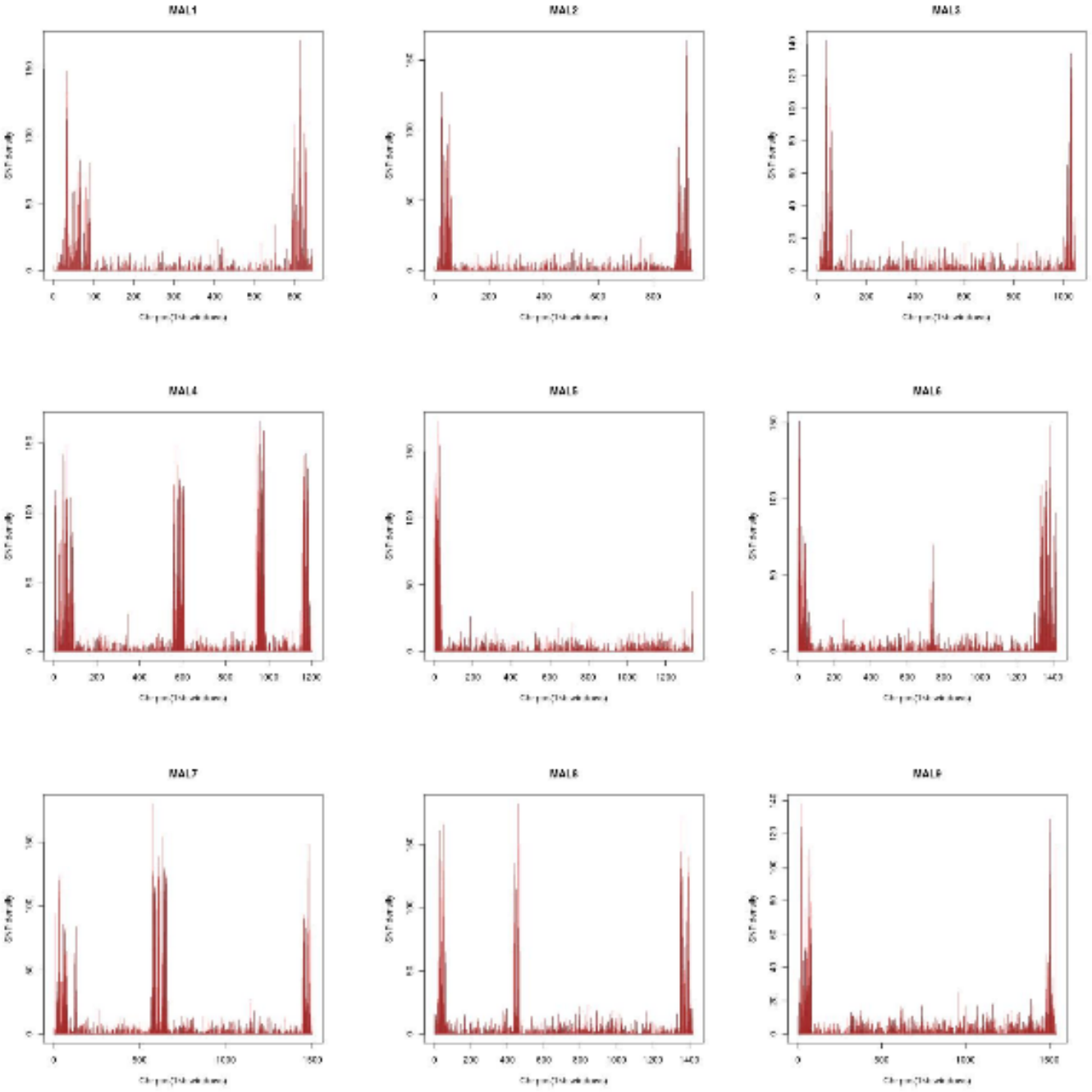
```
   100       110       120       290       300
    .         .         .         .         .
ACGTACGTACGTACGTACGTACGT[...]ACGTACGTACGTAC
ACGT------------------------[...]----------GTAC
```

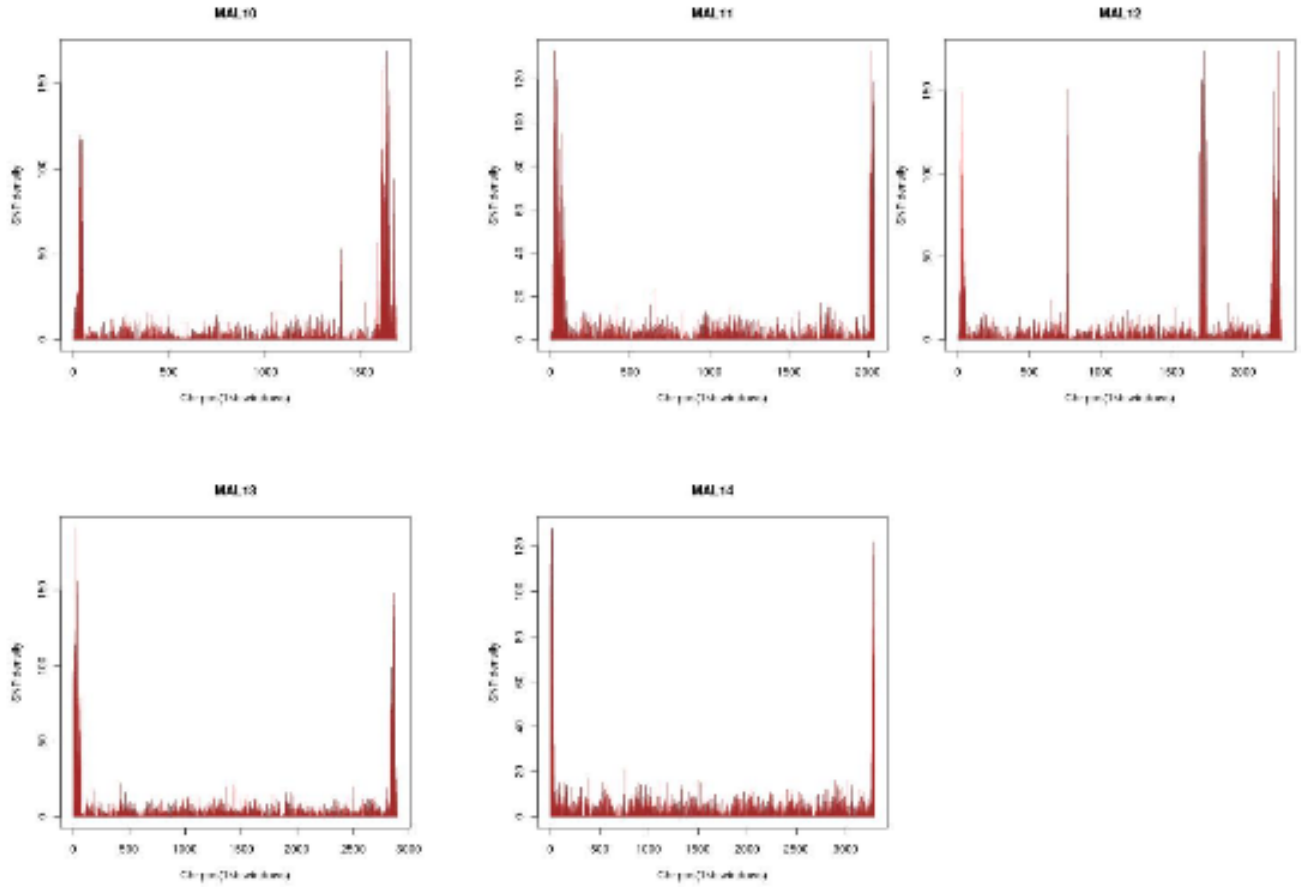
VCF representation

POS	REF	ALT	INFO
100	T	<DEL>	SVTYPE=DEL;END=299

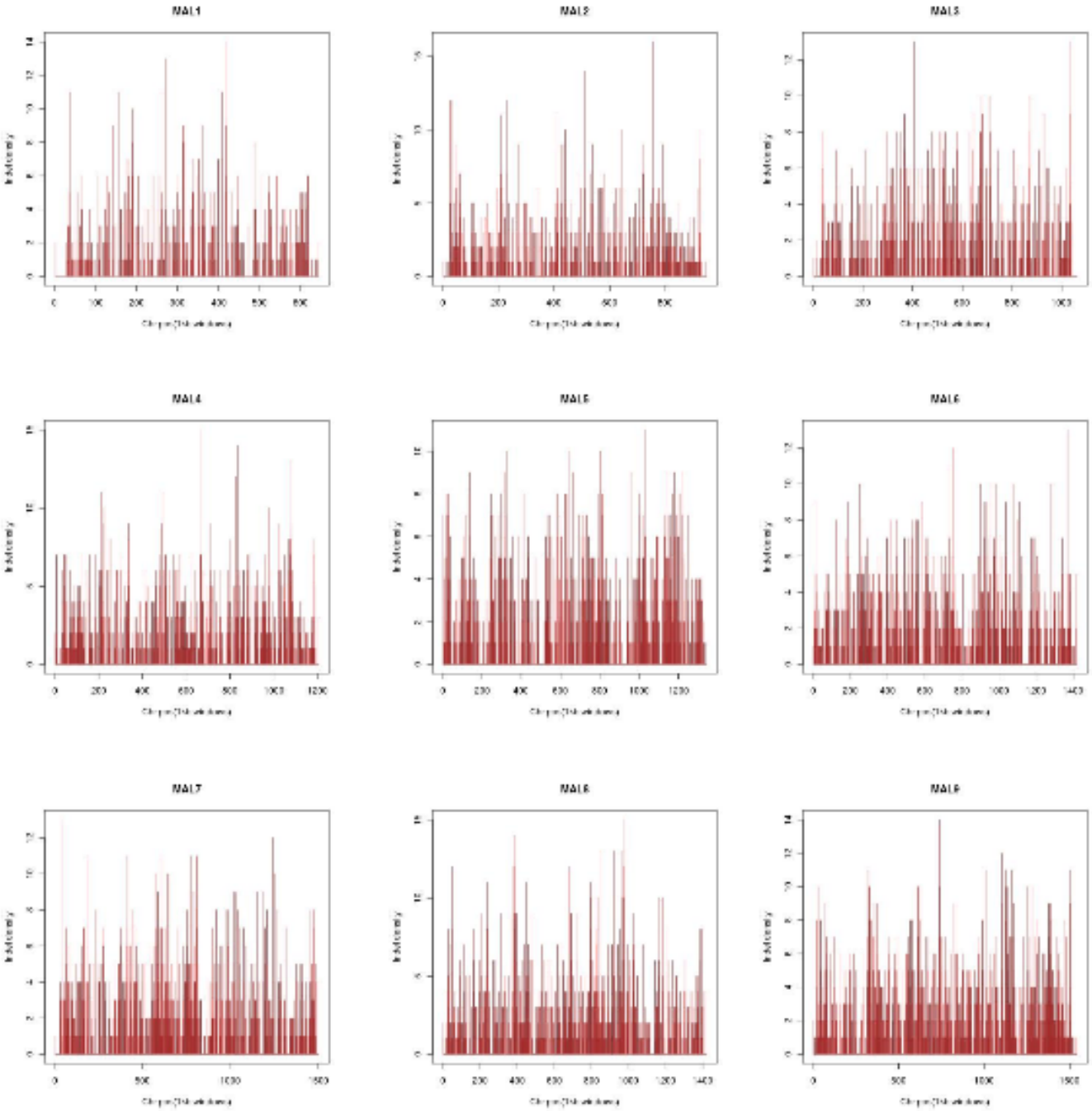
7.1: Partial VCF file example from Danecek et al. (2011).

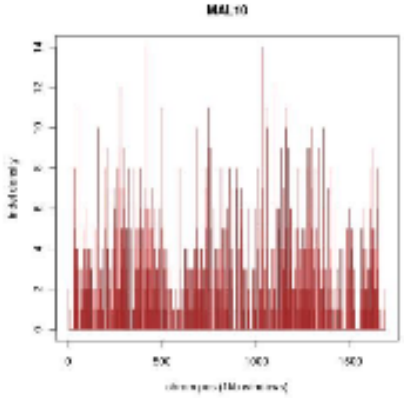
7.1.2. SNP density of each chromosome

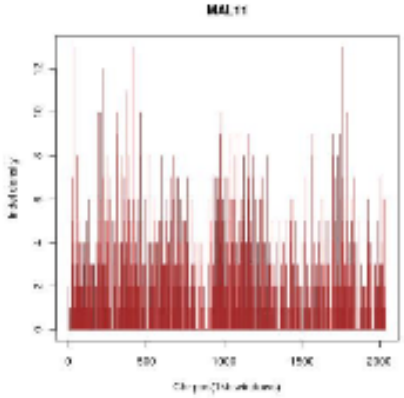


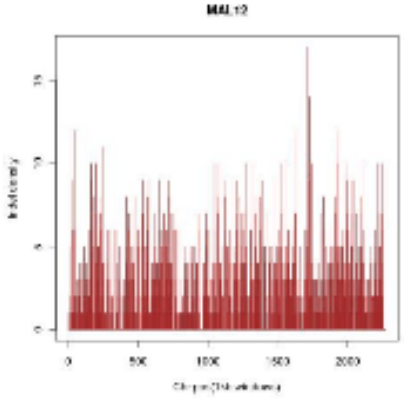

MAL10
MAL11
MAL12
MAL13
MAL14

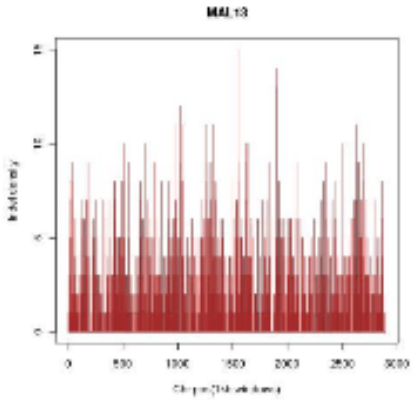
7.1.3. Indel density of each chromosome

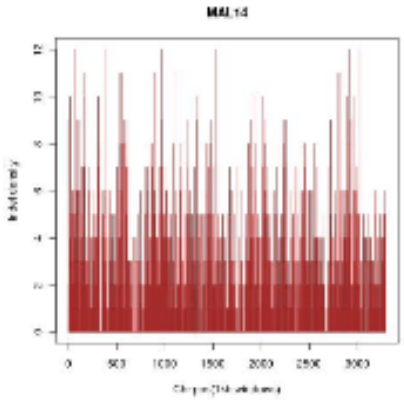



MAL10


MAL11


MAL12

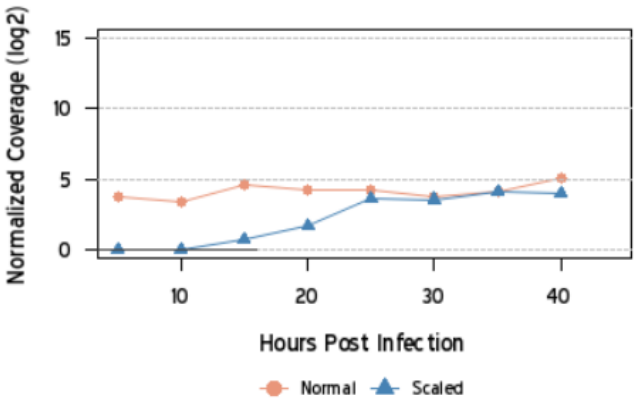

MAL13


MAL14

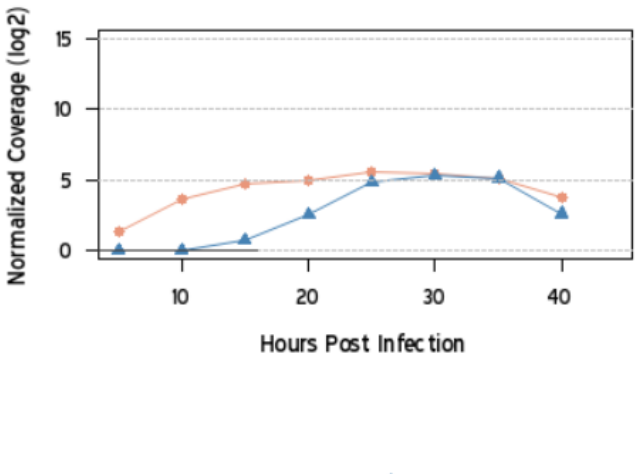
7.1.4. Expression profiles of unknown genes containing premature stop codons

A- MAL13P1.17, B-MAL13P1.155, C-PFL1375w, D-PFI1205c, E- PFL1375w, F- PFL1445w, G-PFC0325c, H-PFC0545c

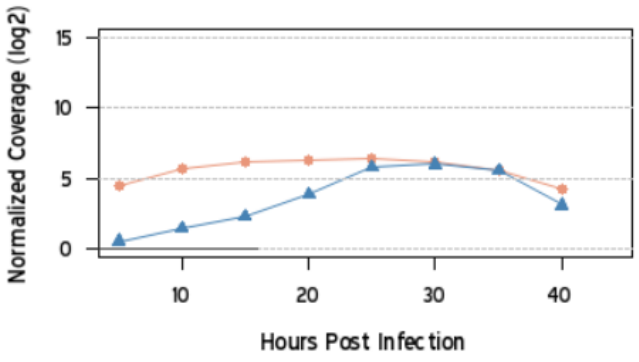
A

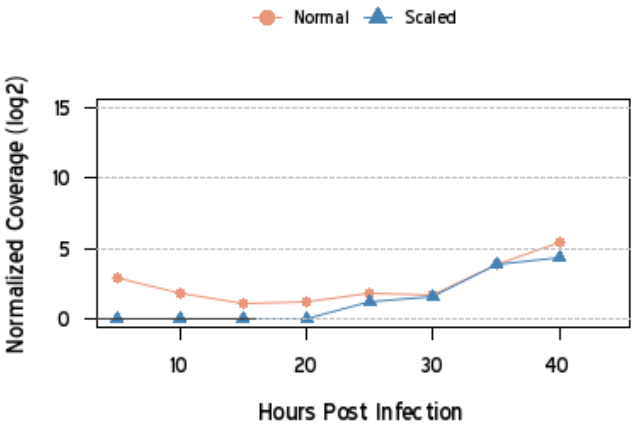


B

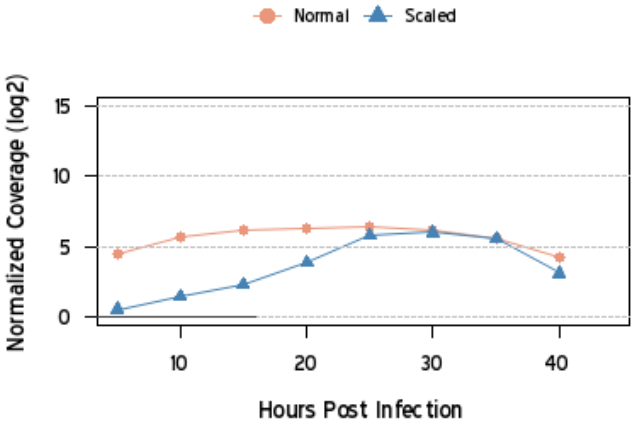


C

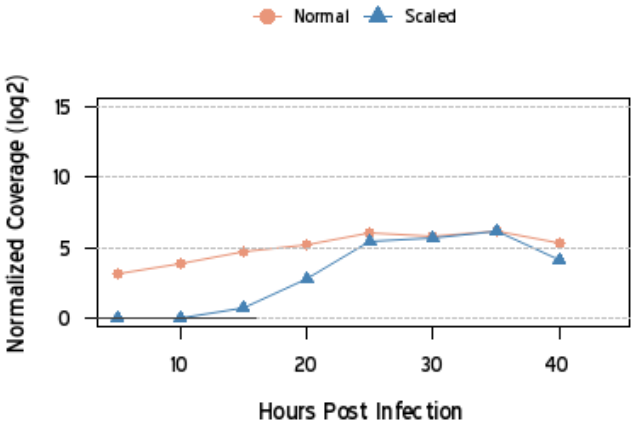



Normal
Scaled
Normalized Coverage (log2)
15
10
5
0
10
20
30
40
Hours Post Infection

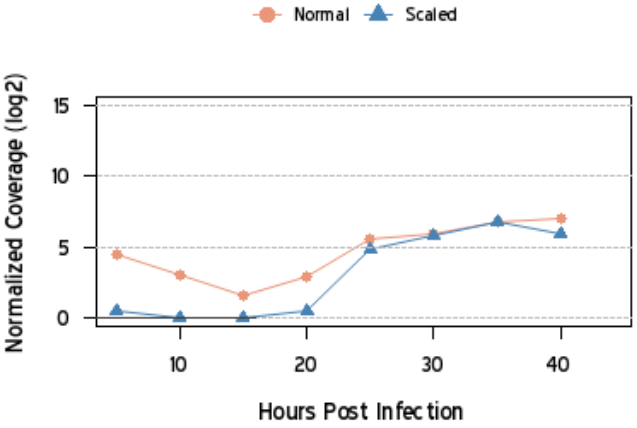
D


Normal
Scaled
Normalized Coverage (log2)
15
10
5
0
10
20
30
40
Hours Post Infection

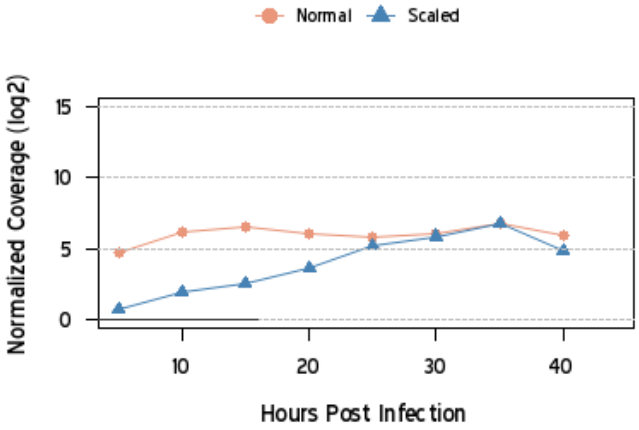
E


Normal
Scaled
Normalized Coverage (log2)
15
10
5
0
10
20
30
40
Hours Post Infection

F


Normal
Scaled
Normalized Coverage (log2)
15
10
5
0
10
20
30
40
Hours Post Infection

G


Normal
Scaled
Normalized Coverage (log2)
15
10
5
0
10
20
30
40
Hours Post Infection

H