1-1-2007

An efficient parallel algorithm for haplotype inference based on rule based approach and consensus methods.

Qamar Saeed
University of Windsor

An Efficient Parallel Algorithm for Haplotype Inference based on Rule Based Approach and Consensus Methods

by

Qamar Saeed

A Thesis
Submitted to the Faculty of Graduate Studies and Research
Through Computer Science
In Partial Fulfillment of the Requirements
For the Degree of Master of Science at the
University of Windsor

Windsor, Ontario, Canada
2007

Your file Votre référence
ISBN: 978-0-494-42329-5
Our file Notre référence
ISBN: 978-0-494-42329-5

Abstract

With recent advancements in bioinformatics technology, completion of 'Human Genome Project', and the exponential growth in genetic data, there is an ever-growing need in genomics to determine the relevance of genetic variations with observed phenotypes and complex diseases. The most common form of genetic variation is a single base mutation called SNP (Single Nucleotide Polymorphism), resulting in different alleles present at a given locus. Grouped together, a set of closely linked SNPs on a particular copy of chromosome is defined as a 'Haplotype'. Research has proven that analysis of haplotypes is potentially more promising and insightful, and hence, at the forefront of bioinformatics investigations, especially due to its significance in the complex disease association studies. Current routine genotyping methods typically do not provide haplotype information, essential for many analyses of fine-scale molecular-genetics data. Biological methods for Haplotype Inference are cost prohibitive and labor intensive. Hence continues the search for more accurate computational methods for haplotype determination from abundantly and inexpensively available ambiguous genotype data.

Continuing the search for a more efficient algorithm, we present in this work two parallel algorithmic approaches, based on 'Inference Rule' introduced by Clark in 1990 and the consensus method reported by Dr. Steven Hecht Orzack in 2003. One approach parallelizes the consensus method. As although, consensus method produces results comparable to other leading haplotype inference algorithms, its time efficiency can be significantly improved to further investigate this promising method for haplotype inference with larger data sets and greater number of iterations of Clark's algorithm. The parallel algorithm is also used to study the affect of different number of iterations used for consensus method. Second parallel approach introduces an Enhanced Consensus algorithm that improves upon the average accuracy achieved by consensus method in much smaller time interval.

Dedication

To my parents and my mother-in-law for their compassionate support and endless love and my caring and loving husband Saeed

Acknowledgments

I am deeply indebted to and would like to express my deepest gratitude to my advisor Dr. Alioune Ngom for his patience, supervision, dedication, suggestions and support throughout my graduate studies at Windsor. His overly enthusiasm, deep insight, and mastery of the subject matter have made working on this thesis a wonderful experience. I have had the opportunity to broaden and improve my analytical skills, as well as gain further understanding of how to formulate and solve intricate problems. I owe him lots of gratitude for having me shown this way of research.

I also owe special thanks to Dr. S. H. Orzack, whose earlier work has been the inspiration for the work presented in this thesis and who always gave so generously of his guidance and ideas. His insight, suggestions and support have been invaluable throughout my research, for which I am deeply grateful to him.

Thanks are due to members of my committee, and especially Dr. R. Kent for his constructive criticism, valuable suggestions and referring.

I would also like to thank a wealth of other people, who have been helpful in large and small ways over the recent years. These include staffs of School of Computer Science: Marry, Debbie, Mandy, Sanjay, and Anis. Many thanks are due to Graduate Studies and Research (University of Windsor) for their financial support during my studies at University of Windsor, and to the School of Computer Science and University of Windsor for providing all the research facilities.

On a personal note I am truly thankful for my parents, whose love and continual faith in me have always been my source of strength and support. Also a special thanks, to our family friends, Badar Gillani and his wife, who have been a constant source of encouragement and great support in my research work. Above all, I am especially indebted to my husband Saeed Ul Haq, whose unwavering support, unconditional solidarity, forbearance, patience and understanding deserve my deepest gratitude and respect. This would have been a daunting task without him indeed.

Table of Contents

List of Tables

List of Figures

Chapter 1

Haplotype Inference

1.1 Overview

Humans have always been interested in gaining a better understanding of themselves and their environment, to satisfy their curiosity, to survive, evolve and most of all to alleviate suffering associated with human disease and improve living conditions.

Scholars have always been aware of differences in human race. Although, members of the same species, millions of human before us and billions alive today, are phenotypically and genetically quite diverse. A quick glance around at people belonging to different races and nationality gives us an idea of the differences amongst us.

However, biologist before Charles Darwin argued that the essential nature of a species does not vary too much from one generation to the next. Charles Darwin contended that the members of any species can vary, and that the variations within a species can be either advantageous or disadvantageous leading to differential selection and eventually contributes to the ongoing process of evolution [Jones 02].

Although Charles Darwin a British naturalist and Gregor Mendal called father of modern genetics were contemporaries. The origin of biological variations and how they

are passed on to the members of the next generations within the same species was not known, until the rediscovery of Mendel's work by Hugo de Vries and Carl Correns in the early 1900s. Mendel described the fundamental principles at the core of genetic inheritance and demonstrates that the inheritance of traits follows particular laws. According to Mendelian laws, physical characteristics are the result of the interaction of genes contributed by each parent, which make up the characteristics of the offspring. Mendel's work provided a genotypic understanding of heredity, missing in earlier studies focused on phenotypic approaches. Then, later in the 1930s and 1940s, the combined understanding of Mendelism and Darwinism culminated in the "modern synthesis of evolution" and formed the basis for modern genetics [Mayr 91].

1.1.1 Genetic Material

Although, the similarities between the theoretical behaviour of Mendel's particles, and the visible behaviour of the newly discovered chromosomes, convinced scientists that chromosomes carry genetic information. However, it was uncertain which component of chromosomes either DNA or protein carried the hereditary information. It was not until the late 1940s and early 1950s that DNA was identified and accepted as the chromosomal component that carried hereditary information. Later, in 1953, Watson & Crick discovered the structure of DNA [Shermer 06].

1.1.2 Structure & Organization of Genetic Material

Human bodies consist of close to 50 to 100 trillion cells, organized into tissues such as skin, muscle, and bone. As shown in Figure 1.1, each cell contains all of the organism's genetic instructions, which are stored as DNA. DNA molecule is tightly wound and packaged as a chromosome. Humans have two sets of 23 chromosomes in each cell. One set inherited from each parent, a human cell; therefore, contains 46 of these chromosomal DNA molecules [Shabarova 94].

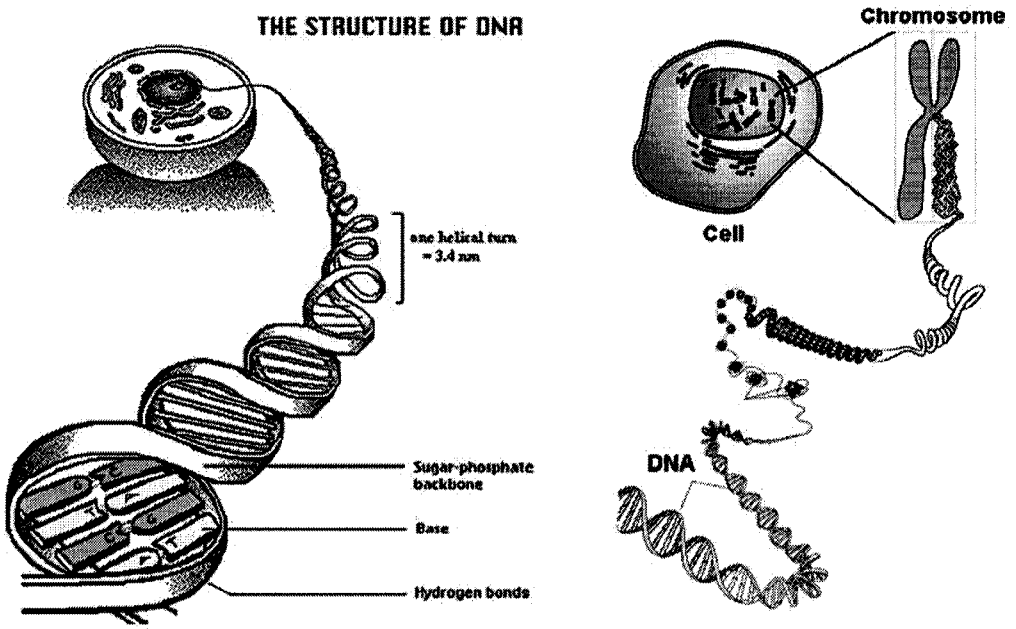

Figure 1.1: Structure and organization of genetic materials in human cell [Jelinek 82].

Each DNA molecule that forms a chromosome can be viewed as a set of short DNA sequences, know as genes. A set of human chromosomes contains one copy of each of the roughly estimated 30,000 genes in the human genome. A haplotype consist of all alleles, one from each locus that is on the same chromosomes. To explain this concept, a structure of a pair of chromosomes from mathematical point of view is further depicted in Figure 1.2.

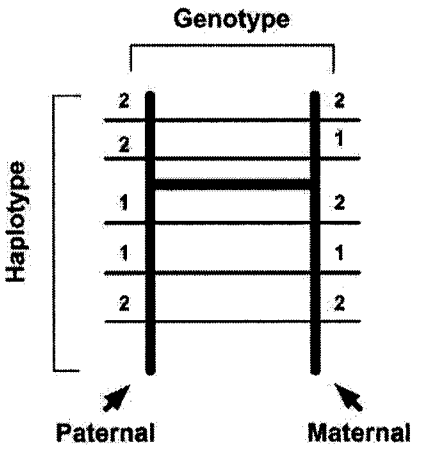

Figure 1.2: Mathematical model of chromosomes structure [Li 03]

Components of a DNA molecule

1. DNA is a polymer made up of monomer units called nucleotides.

2. Each nucleotide consists of 5-carbon sugar (deoxyribose), a nitrogen-containing base attached to the sugar, and a phosphate group.

3. There are four different types of nucleotides found in a DNA molecule. Each nucleotides is representable by the following:

A: adenine (a purine)
G: guanine (a purine)
C: cytosine (a pyrimidine)
T: thymine (a pyrimidine)

The number of purine bases equals the number of pyrimidine bases. The structure of the DNA molecule is helical, which contains equal numbers of purines and pyrimidine bases that are piled on top of each other, is depicted in Figure 1.3.

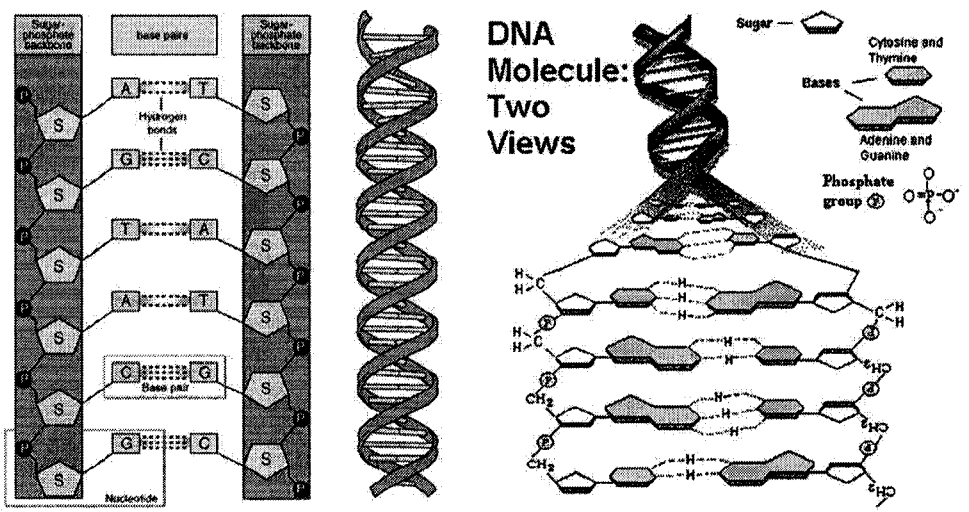

Figure 1.3: Structure of DNA molecule [Watson 80].

1.1.3 Analysis of Genetic Material

Watson and Crick's discovery of the structure of DNA has facilitated the development of genetic material analysis at a molecular level. Currently, DNA analysis is fast and accurate. This is due to powerful new techniques and technological advances.

Now that the Human Genome Project (HGP) is almost complete, the next task is to organize, analyze and interpret the massive amount of data from sequencing projects throughout the world; and to perceive genetic differences as the basis of the wide range of human traits.

1.1.4 Genetic Variation

The ultimate cause of genetic variation is the differences in the DNA sequences. Unlike other species that predate humans by 150,000 years, humans have not had much time to accumulate genetic variations. Approximately 99% of genetic information of humans is identical, as confirmed by the completion of the HGP. The HGP information was completed in 2003 with the sequencing of 99% of the genome with 99.99% accuracy [Francis 03].

Despite so many shared characteristics, the 1% difference is the cause of a wide range of the phenotypic differences in human traits. The human genome comprises of about 3 billion base pairs of DNA. The level of human genetic variation is such that no two humans, except for identical twins, have ever been, or will be, genetically identical.

Most of genetic variations do not affect how individuals function. However, some genetic variations are related to disease, or the ability to survive changes in the environment. Genetic variation; therefore, is the foundation of evolution by natural selection.

Genomes are considered to be a collection of long strings, or sequences, from the alphabet {A, C, G, and T}. Each element of the alphabet encodes one of four possible nucleotides, presented in Section 1.1.2 [Halldorsson 04]. Single Nucleotide Polymorphisms (SNPs), caused by the alteration of a single nucleotide (A, T, C, or G) in

the genome sequence (see Figure 1.4), are the most common and smallest evolutionarily stable variations of the human genome [Beaumont 04], [Chakravarti 98].

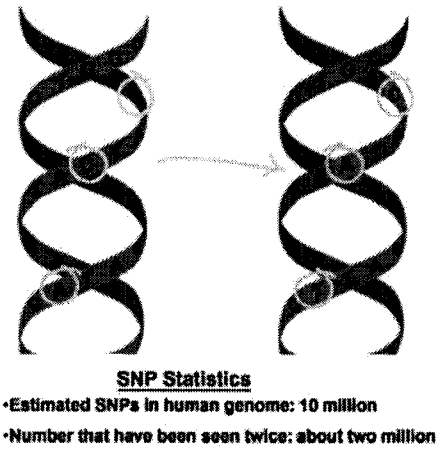

Figure 1.4: SNPs observed between chromosomes in DNA.

1.1.5 Single Nucleotide Polymorphisms (SNPs)

A Single Nucleotide Polymorphism (SNP) is a single base pair in genomic DNA at which different nucleotide variants subsist in some populations; each variant is called an allele [Lin 97], [Stephens 00]. For instance, an SNP can change the DNA sequence AAGGCTAA to ATGGCTAA. In Figure 1.5, a more simplistic example of chromosome with three SNP sites is given. The individual is heterozygous at SNPs 1 and 3 and homozygous at SNP 2. The haplotype are CCA and GCT.

Chrom. C, parental: ataggtccCtatttccaggcgcCgtatacttcgacgggaActata

Chrom. C, maternal: ataggtccGtatttccaggcgcCgtatacttcgacgggaTctata

Haplotype 1 →	**C**	**C**	**A**
Haplotype 2 →	**G**	**C**	**T**

Figure 1.5: A chromosome and the two haplotypes [Rizzi 02].

Even though more than two alleles are not uncommon. In humans, SNPs are usually biallelic; that is, there are two variants at the SNP site. The most common variant is referred to as a major allele, and the less common variant is called a minor allele. SNPs are differentiated from other mutations by a frequency of more than 1% in the population [Orzack 03], [Peer 03].

SNPs are composed of about 90% of human genetic variations, and occur at 100 to 300 bases along the 3-billion-base human genome. Other types of polymorphisms, including, differences in copy number, insertions, deletions, duplications, and rearrangements, are less frequent.

Two of every three SNPs involve the substitution of C for T, and can occur in the coding (gene) and non-coding regions of the genome. Although, many SNPs have no effect on cell function, they can predispose people to a certain disease or influence their response to a drug. SNPs constitute the genetic aspect of human response to environmental stimuli such as bacteria, viruses, toxins, chemicals, and drugs. SNPs are extremely valuable for biomedical, pharmaceutical research and medical diagnostics [Rizzi 02]. An important characteristic of SNPs is that they are stable from one generation to the next, which come in handy for tracking them in population studies.

SNP maps are being created by the joint efforts of HGP and SNP consortium, consisting of a large number of pharmaceutical companies. So that additional markers can be found along the human genome, which will make it much easier to navigate the very large genome map created by scientist in HGP [Jorde 01].

1.1.6 Haplotypes

Although SNPs help to identify multiple genes, related with complex diseases such as cancer, and diabetes [Lippert 02], [Niu 04a]. SNPs have provided researcher with even more powerful tool, called "Haplotypes" [Clark 90].

Haplotypes are the sets of SNPs along the human genome or chromosomes that are most likely to be inherited generation after generation without recombination [Lam 00]. They can be defined as a selection of alleles at different markers along the same

chromosome that are inherited as a unit [Lin 02]. As a result, an individual possesses two haplotypes, representing the maternal and paternal chromosomes for a given segment of the genome [Bonizzoni 03], [Halldorsson 03]. Therefore, it can be stated that each copy of the chromosome is called a haplotype. The conflation of the two inherited haplotypes, which may be identical, is called a genotype. The difference in both haplotypes and genotypes are elaborated below in Figure 1.6.

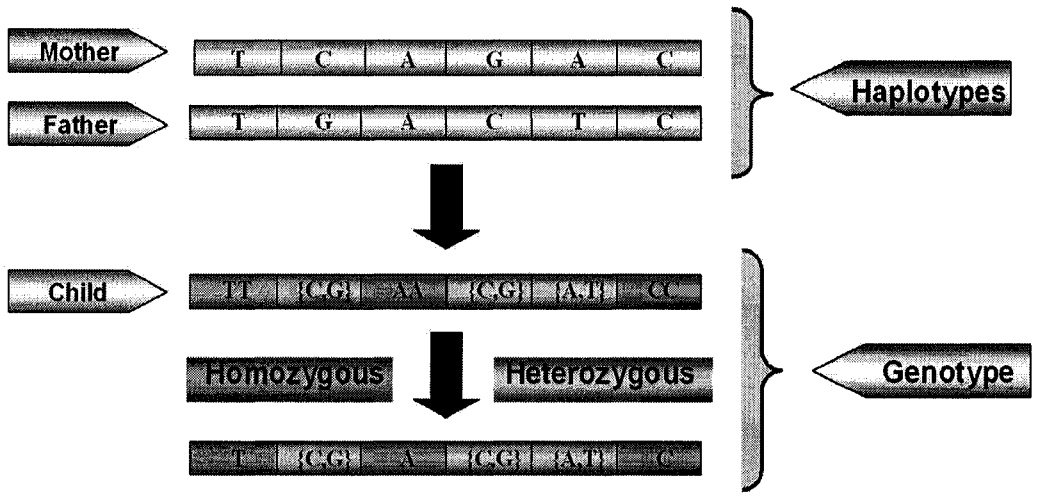

Figure 1.6: Difference between haplotypes and genotypes.

Only a small number of haplotypes are found in human. For example, as shown in Figure 1.7, in any given population, the distribution of different haplotypes can be 55%, 30% and 8%. The other haplotypes are a variety of less common haplotypes [Gusfield 04].

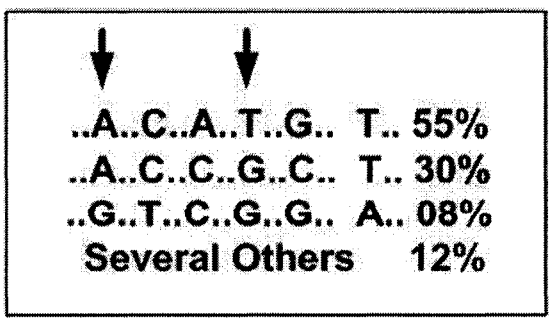

Figure 1.7: Distribution of haplotypes in population.

1.1.7 Significance of Haplotypes

Complex diseases such as diabetes, cancer, heart disease, stroke, depression, or asthma are affected by more than one gene. As a result haplotype data have been proven to be much more informative than genotype data. Haplotype analysis can help researchers in the following ways.

- To pinpoint the disease-causing locus, by observing the recombination events in both family-based and population-based studies [Johnson 01].
- To identify the genetic variants that contribute to longevity and resist disease, leading to new therapies with widespread benefits.
- To discover the genetic variants, involved in the disease and individual responses, to therapeutic agents.
- To uncover the origin of illnesses and discover new ways to prevent, diagnose, and treat such illnesses.
- To customize medical treatments, based on a patient's genetic make-up, to maximize effectiveness and minimize side effects.

Also haplotypes are considered to be "molecular fossils", because the frequency of a certain haplotype in the sample can be used to infer its topological position in a cladogram, providing snapshots of human evolutionary history [Niu 04b].

1.2 Haplotype Inference

The haplotyping problem for an individual is to determine a pair of haplotypes for each copy of a given chromosome [Rizzi 02], [Wang 03].

Currently practical laboratory techniques provide un-phased genotype information for diploids, that is, an unordered pair of alleles for each marker. The reconstruction of haplotypes from genotype data is now a crucial step in the analysis process.

Unless an organism is a homozygote, its genotype might not uniquely define its haplotype. A direct sequencing of genomic DNA from diploid individuals leads to ambiguities on sequencing gels. This results in more than one mismatching site in the sequences of the two orthologous copies of a gene (see Figure 1.8). These ambiguities cannot be resolved from a single sample without resorting to other experimental methods.

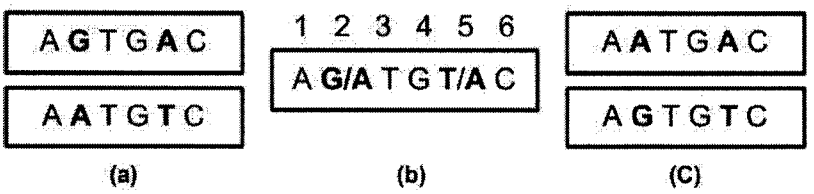

Figure 1.8: An example of 6 SNPs with two homologues of an individual (a) Individual's haplotypes, (b) individual genotype, (c) another potential haplotype pair.

Although the acquisition of sequences of multiple alleles from natural populations provides the ultimate description of genetic variation in a population, the time and labor involved in obtaining the sequence data has limited the number of such studies. Any means of acquiring the sequence data from population samples that decreases this effort is pivotal to further achievements.

1.2.2 Biological Methods for Haplotype Inference

The haplotype determination of several markers for a diploid cell is difficult. Unless the individual is homozygous or haploid DNA is being sequenced existing genotyping techniques cannot determine the phases of several different markers. For example as shown in Figure 1.9, a genomic region with three heterozygous markers can yield eight possible haplotypes. This uncertainty can, in some cases, be solved if pedigree genotypes are available. However, even for a haplotype of only three markers: the genotypes of father, mother, and offspring trios can fail to predict offspring haplotypes as much as 24% of the time.

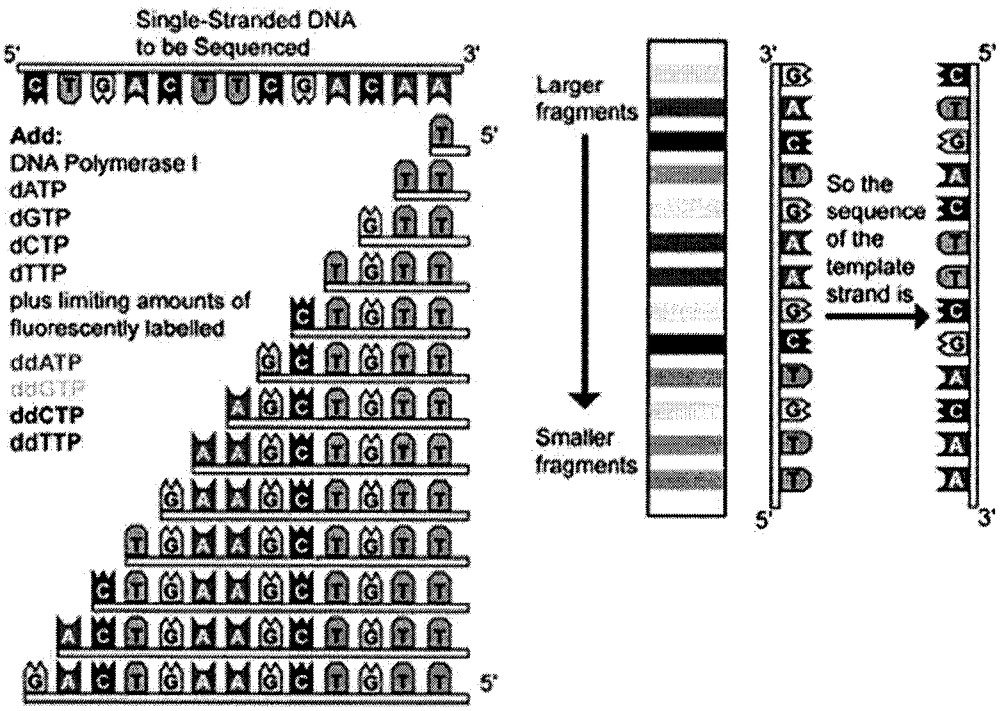

Figure 1.9: Sequencing example of a single stranded DNA.

An example is shown below in Table 1.1. Consider two loci on the same chromosome. Each locus has two possible alleles; the first locus is either, A, or a, and the second locus being B or b. If the organism's genotype is AaBb, there are two possible sets of haplotypes, corresponding to which pairs occur on the same chromosome:

Table 1.1: Example of two possible sets of haplotypes.

Set	Haplotype at chromosome 1	Haplotype at chromosome 2
Haplotype set 1	AB	ab
Haplotype set 2	Ab	aB

In this case, more information is required to determine which particular set of haplotypes occurs in the organism (i.e., which alleles appear on the same chromosome).

Alternatively, several direct molecular haplotyping methods have been developed for haplotype inference. The methods are based on the physical separation of two homologous genomic DNAs before genotyping can be used. Examples include the following:

1. M1-PCR Method
2. DNA cloning somatic cell hybrid construction
3. Single molecule dilution
4. Allele-specific, single-molecule, and long-range PCR

1.2.3 Significance and Shortcomings of Molecular Haplotyping Methods

Significance

1. Typically, these approaches are largely independent of pedigree information [Ding 03], [Niu 04b], [Ruano 90], [Tost 02].
2. Molecular haplotyping techniques, such as the M1-PCR method, have resulted in a genotyping success rate, for single copy DNA molecules, of approximately 100%.

Shortcomings

1. Typically, molecular haplotyping methods are time consuming and labor intensive.
2. Some of these methods, including for instance the allele-specific and single-molecule PCR are limited to short genomic regions (<3 kb).

1.2.4 Computational Methods

The advent of the polymerase chain reaction (PCR) [Saiki 85], [Scharf 86] has substantially accelerated the process of genomic DNA to sequence data by eliminating the cloning step. The direct sequencing of PCR products works well for mtDNA or for DNA from isogenic, or otherwise homozygous or hemizygous regions, but heterozygosity in diploids results in an amplification of both alleles.

By using asymmetric amplification, with unequal concentrations of the two primers, single-stranded DNA products can be directly sequenced. In a heterozygote, an asymmetric PCR results in amplification products of both homologues. The resulting superimposition of the two sequencing ladders for the two alleles produces a vast number of possible haplotypes for any heterozygous individual. If there are n such "ambiguous" sites in an individual, then there are 2^n" possible haplotypes. The challenge is to devise a scheme, whereby haplotypes can be inferred from a series of these ambiguous sequences, constructed from samples of diploid natural populations.

Input to the computational method consists primarily of n genotype vectors, each with length m, where each value in the vector is 0, 1, or 2. Each position in the vector represents an SNP on the chromosome. The position in the genotype vector has a value of 0 or 1 for homozygous site, and a value of 2, otherwise, for a heterozygous site [Lippert 02], [Niu 04a].

Given an input set of n genotype vectors, the computational methods are used to calculate an optimal solution to the Haplotype Inference (HI) problem that is to determine the best possible haplotype pairs, which provide complete SNP fingerprint information for that chromosome, and for each individual in a sample data set [Lancia 01], [Li 04].

1.2.5 Significance and Shortcomings of Computational Methods

Although haplotypes can be directly determined by biological methods, an experimental study of haplotypes is technically difficult, because the direct molecular haplotyping methods are labour intensive and expensive, and exhibit a low throughput [Terhalle 03].

However, computational methods, although not perfectly accurate, are cost efficient and less laborious than that of biological methods, providing a more practical and accurate alternative to experimental haplotyping methods [Adkins 04].

1.3 Different Algorithmic Approaches

Haplotypes are much more informative and helpful than SNPs, since they capture linkage disequilibrium information far better than SNPs alone [Casey 03], [Clark 04], [Service 99]. Consequently, haplotype inference is important for many genetic studies, including related studies for complex diseases that are related a particular phenotype in a specific genetic region.

There are different approaches to solve this problem, including the nuclear family design, in which the haplotypes are inferred from the parent's genotypes of the individual. However this involves a substantial increase in genotyping costs. Also, it is difficult to recruit the parents of the diseased individuals, since most of the complex diseases are late onset. As a result, it is much easier to genotype individuals for complex diseases by a case-control design rather than a family-based design.

The computational methods for the haplotype reconstruction problem can be categorized as follows:

Haplotype Inference Using Population Data

1. **Combinatorial Methods**

   a. Rule-Based Methods, e.g., Clark's Inference Rule.

   b. Coalescent Models

   Markov-chain Monte Carlo (MCMC) Method

   (e.g., Pseudo-Gibbs Sampler (PGS) Algorithm)

   c. Perfect Phylogeny (e.g., HAP)

2. Statistical Methods

a. Maximum Likelihood Approach, e.g., Expectation maximization algorithm.

b. Bayesian Inference Methods, e.g., Partition Ligation (Haplotyper, Phase.)

1.3.1 Rule based Methods

Clark's Rule

Clark introduced the earliest algorithm for haplotype reconstruction, and the inference of haplotype frequencies from genotype data, based on the principle of maximum parsimony [Clark 90].

The algorithm, known as Clark's Rule is used to assign the smallest number of haplotypes to the observed genotype data. First, the algorithm forms an initial set of all the unambiguous haplotypes that include all the homozygotes and single-site heterozygotes. This is the set of resolved haplotypes. The algorithm is then applied to determine the remaining haplotypes, based on the set of unambiguous haplotypes that is, Clark's Rule is used to determine if any of the ambiguous haplotypes can be explained according to the recently resolved haplotypes. Each time that a new haplotype is identified, it is added to the set of resolved haplotypes, until all the genotypes are resolved or cannot be explained by the resolved haplotypes.

For this algorithm, it is assumed that the input genotype data contains unambiguous homozygous haplotypes, a necessary condition for starting the algorithm. According to Clark, when all the haplotypes are determined, based on the principle of maximum parsimony, the solution is unique and accurate [Clark 90]. Compared with the results obtained by experimental methods, the results generated by application of Clark's algorithm to certain empirical data is found to be reliable.

The algorithm does not assume HW equilibrium, unlike the other procedures for haplotype reconstruction; however, the departure from the HW equilibrium, by depending on Clarks' algorithm, has proven to skewed results. To find a robust and fast

modification of Clark's algorithm Gusfield redefined it as the "Maximum Resolution" (MR) problem [Gusfield 00], [Gusfield 01]. The MR problem can be stated as follows:

"Given a set of both ambiguous and unambiguous vectors, which is the maximum number of ambiguous vectors that can be resolved by successive iterations of Clark's inference rule".

The MR problem has proved to be NP-hard and Max-SNP-complete by Gusfield. He has also proposed a solution by first reformulating the problem as a directed graph problem and then solving the graph problem with integer linear programming [Gusfield 01], [Gusfield 03]. Despite the Clark's algorithm's simple nature, it is quite popular due to its relatively straightforward procedure, and capability to handle a large number of loci, when the haplotype diversity is rather limited in the population.

Clark's algorithm does exhibit some limitations. For instance, the algorithm might not even run, when there are no unambiguous haplotypes (i.e., either homozygotes or single-site heterozygotes) in the population genotype data. Also, since the results depend on the order of entry of ambiguous haplotypes the algorithm does not give unique solutions; thus, Clark's algorithm might not be able to resolve many haplotypes, where there are a greater number of distinct haplotypes, due to recombination of the hotspots, in a relatively small sample size. Although Clark's algorithm does not explicitly assume the Hardy-Weinberg equilibrium (HWE), the performance is still relatively sensitive to the deviation from the HWE [Niu 02].

1.3.2 Coalescent Models

The coalescent-based model is assumed to represent the evolutionary history by a rooted tree, where each leaf of the tree represents each given sequence [Gusfield 91]. The model also yields, at most, one mutation per given site in the tree, infinite site model, restricting the repeated mutations, and thus supporting no recombination assumption. As a result, the mutations are directed [Hajiaghayi 05].

Pseudo-Gibbs Sampler (PGS) Algorithm

The coalescence based Markov-chain Monte Carlo (MCMC) approach has been proposed for haplotype reconstruction from genotypic data [Liu 01], [Stephens 01]. The PGS has been developed by using Gibbs sampling algorithm to calculate the approximate sample from posterior distribution [Stephens 01].

Also, the PGS algorithm has been modified to use a modified version of the partition ligation approach allows the recombination and decay of linkage-disequilibrium with distance [Niu 02]. The advantages of PGS include the incorporation of the coalescence theory into its prior and a reliable performance. Conceptually, the Gibbs sampler alternates between two coupled stages.

However, the PGS algorithm lacks the overall "goodness" of the constructed haplotypes, since the algorithm is not a full Bayesian model. Also, PGS's piece-by-piece strategy to update the haplotypes, based on the existing haplotypes is slow and requires a tremendous number of iterations in the range of millions for an accurate result generation. [Stephens 03].

Perfect and Imperfect Phylogeny (PPH)

The perfect phylogeny haplotype problem can be defined as follows:

Given a set of genotypes, find a set of explaining haplotypes, which defines an un-rooted perfect phylogeny. The perfect phylogeny problem assumes "no recombination", as well as the infinite site model [Bafna 02], [Bafna 03]. The problem has been proven to be NP-hard [Kimmel 04]. A phylogeny-based algorithm is adopted to deterministically infer haplotypes, according to phylogenetic reconstruction [Chung 03b], [Eskin 04a], [Gusfield 02].

The program, ''Perfect Phylogeny Haplotyper'' has been developed for determining if the input un-phased genotype data can be explained by haplotype pairs evolved on perfect phylogeny [Chung 03a]. The PPH program solves a problem by first reducing it to a graph realization problem, solving it, and then translating the results back to a PPH

problem [Barzuza 04], [Daly 01], [Gusfield 02]. The program is fast, verifies its results, and also provides a corresponding tree format for the solution [Halperin 04b].

Recently, a new method has been developed realized by using imperfect phylogeny that extends the framework of PPH by allowing for recurrent mutations and recombinations [Halperin 04a]. The method partitions the multiple linked SNPs into blocks, and for each block, predicts an individual's haplotype. The haplotype blocks are then assembled to form a long-range haplotype by utilizing the PL idea [Niu 02]. The imperfect phylogeny method appears to be more reliable than PPH, and allows for the handling of missing data and for resolving a large number of SNPs.

1.3.3 Expectation-Maximization (EM) Algorithm

The expectation-maximization (EM) algorithm was first introduced, by Dempster *et al.* in 1977 [Dempster 77]. They have formalized the use of the EM algorithm for a haplotype frequency estimation that maximizes the sample probability.

Excoffier and Slatkin [Excoffier 95] were the first to discuss the use of the EM algorithm for estimating the population haplotype probabilities, based on the maximum likelihood, and finding the haplotype probability values that optimize the probability of the observed data, based on the assumption of HWE. They authors have demonstrated how the EM algorithm can be extended to apply to a genetic sequence and highly variable loci data [Excofier 95].

Under the assumption of the HWE, the EM algorithm is an iterative method for computing successive sets of haplotype frequencies. The iterative step can be represented as follows:

$$\theta_a^{(k+1)} = E_\Theta^{(k)}(n_a|G) / 2n, \qquad (1.1)$$

where $\Theta^{(k)}$ is the present estimate of the haplotype frequencies, and n_a is the count of haplotype 'a' in G.

The EM algorithm is based on the solid statistical theory, and performs well for the simulated data for both the genetic sequence and highly variable loci [Clark 01],

[Excofier 95], [Fallin 00]. Although the algorithm makes an explicit assumption of HWE, its performance in simulation studies is not much affected by the departures from HWE, especially increased homozygosity [Niu 02].

The limitations of the EM algorithm include its sensitivity to the initial values of haplotype frequencies. Also it is well known that the EM algorithm can be trapped in a local mode, if there exist a local maxima, leading to locally optimal maximum likelihood estimates (MLEs). The local mode problem increases with the increasing number of distinct haplotypes. It is noteworthy that the initial values of the haplotype frequencies are reasonably close to the true population frequencies and help [Excofier 95], [Kelly 04]. In addition, the standard EM algorithm is limited in its capability of handling a large number of loci [Eskin 04b].

1.3.4 Partition Ligation (PL) Algorithm

The Partition-ligation (PL) algorithm, a divide-conquer-combine approach has been introduced to handle a large number of loci. [Niu 02]. The PL algorithm depends on the assumption that a long-range haplotype is a combinatorial set of atomistic units, supported by the empirical observations of the underlying haplotype block patterns of the human genome [Daly 01].

First, he long-range haplotype is partitioned into a series of smaller, atomistic units. As a result, the haplotype phasing for each atomistic unit becomes tractable, and the long-range haplotype is considered the ligated product of these units. In the ligation step, one of two strategies can be adopted: hierarchical ligation or progressive ligation [Niu 02]. HAPLOTYPER and PLEM PL developed are realized by hierarchical ligation; whereas, the variants of the progressive ligation were implemented by a modified version of PHASE version 1.0 and SNPHAP [Niu 02], [Qin 02].

The algorithm, implemented in HAPLOTYPER, is based on a new Monte Carlo approach, a robust Bayesian procedure with the same statistical model as the EM. The new algorithm has been designed to overcome the shortcomings of the existing, and otherwise, efficient and feasible methods such as Clark's, EM, or coalescence-based

iterative-sampling algorithms. The method uses two novel techniques PL and prior annealing to divide the haplotypes into smaller segments, and Gibbs sampler for partial haplotype construction and segment assembly. The method can handle HWE violations, missing data, and recombination hotspot occurrences [Niu 02].

1.3.5 Haplotype Inference using Pedigree Data

Pedigree data differs from population data in the relations that exist among the genotyped individuals, which otherwise, are considered to be independent in population-based studies. For pedigree data, the relations of the parenthood are related individuals [Li 03a].

A number of linkage analysis programs are used to reconstruct the haplotypes, by using pedigree data. These include Genehunter, Simwalk2, and Merlin [Sobel 96]. Here, it is assumed that the linked markers are in linkage equilibrium, implying that all possible haplotype configurations have same likelihood in the case of phase ambiguity [Schaid 02]. For Genehunter, the results depend on the order in which the alleles are entered. Consequently, for the few haplotype ambiguities, Genehunter, Merlin, and Simwalk2, should be applied. The use of trios is becoming more and more popular due to their relative efficiency and ease of sample collection.

The Rule-based method, along with the genetic algorithm, has also been proposed for haplotype inference from pedigree genotype data. Implemented in HAPLOPED, the algorithm has shown potential, but cannot be compared to the existing programs such as GENHUNTER and SIMWALK due to the differences in assumptions and data requirements [Tapadar 00].

EM-based haplotype inference methods have been developed without the assumption of linkage equilibrium among the linked markers. The nuclear family data have been shown to be more efficient than that of the Lander-Green algorithm of Genehunter [Rhode 01].

A rule-based ''minimum- recombinant haplotyping'' (MRH) algorithm exhaustively searches for all possible minimum-recombinant haplotype configurations in large pedigrees with many markers [Li 03b], [Li 03c]. MRH allows missing genotype data to

be imputed from identity-by-descent alleles [Doi 03], [Orzack 03]. Although trios are more suitable to sample than large pedigrees, for certain complex traits with the late onset of a disease, the parental DNA samples often cannot be typed.

1.3.6 Haplotype Inference using Pooled DNA Samples

To reduce the high cost of genotyping, pooled DNA samples have been introduced. An allele-specific, long-range PCR method for the direct measurement of haplotype frequencies in DNA pools has been devised [Istrail 97]. This method, however, is not feasible for neighboring SNPs that are separated by intervals longer than 420 kb.

Pooling complicates the haplotype configurations and adds to the ambiguities in the haplotype frequency estimation. Although DNA pooling is very cost-efficient, it can increase errors in allele and haplotype frequency estimations [Yang 03]. Instead, the use of small DNA pools, for instance, a pool size of 2, has been proven to be superior to the use of large pools [Hoh 03], [Wang 03].

Two EM-based algorithms have been developed for haplotype reconstruction from the pooled genotype data. An EM algorithm for obtaining the maximum likelihood estimations (MLEs) and the haplotype frequency estimations has been developed, to cope with the missing data [Yang 03].

1.4 Organization of Thesis

This thesis is organized in 5 chapters. Chapter-2 gives an overview of the previous research in the related area and discusses different approaches. In chapter-3 we presented the proposed methodology for an efficient parallel algorithm for haplotype inference based on rule based approach and consensus methods. In chapter-4 we discuss the results based on the proposed approach. The analysis in this chapter measures the performance of parallel algorithm. Chapter-5 includes conclusion and suggests directions for future work.

Chapter 2

Review on Clark's Algorithm

2.1 Introduction to Clark's Algorithm

In 1990s, A. G. Clark first introduces the Clark's Algorithm. It continues to be a popular method for solving the haplotype inference problem. Clark's algorithm consist of the following four steps:

1. The identification of initial resolved haplotypes: these are retrieved from the genotype vectors with either no ambiguous sites or a single ambiguous site, since these genotypes can be resolved only in one way.
2. Resolution of the ambiguous genotypes: here, there is more than one ambiguous site. It requires the help of the initial resolved genotypes and Clark's Inference Rule. Clark's Inference Rule is stated as follows:

Suppose A is an ambiguous vector with h ambiguous sites, and R is a resolved vector that is a haplotype in one of the $2h-1$ potential resolutions of vector A. Then, infer that A is the conflation of one copy of resolved vector R and another (uniquely determined)

resolved vector NR. All the resolved positions of A are set the same in NR, and all of the ambiguous positions in A are set in NR opposite to the entry in R. Once inferred, vector NR is added to the set of the known resolved vectors, and vector A is added to the set of ambiguous vectors.

3. To resolve vector R can be used to resolve an ambiguous vector A if and only if vector A and vector R consist of identical unambiguous sites.
4. Clark's inference rule is repeatedly applied until either all the genotypes have been resolved or no further genotypes can be resolved.

Clark's algorithm appears to be straightforward and should work in theory. However, following situations are problematic:

1. The sample data do not contain any homozygotes or heterozygotes; therefore, the algorithm cannot be run.
2. All the genotypes are not resolved by the end of the inference method.
3. The haplotypes can be erroneously inferred, if the crossover product of two real haplotypes is identical to that of another actual haplotype.
4. It is possible that these problems depend on the average heterozygosity, the length of the genomic sequence, sample size, and the recombination rates of the sites.

The basis of the method is that a phase-ambiguous sampled genotype data is likely to contain known common haplotypes.

2.2 Genetic Model of Clark's Algorithm

Clark's algorithm assumes that the sample is small, compared to the actual population size. The genotypes in the population are the result of the random mating of the parents

of the current population and that the sampled individuals are randomly drawn from the population. Thus, the initial resolved haplotypes represent common haplotypes, occurring with a high frequency in population.

Also, Clark's method is used to resolve the identical genotypes in the same way, assuming that the history leading to the two identical genotypes is identical. This is justified by the "infinite sites" model of population genetics, in which only one mutation at a given site has occurred in the history of the sampled sequences [70].

The previous assumptions are consistent with the way Clark's algorithm gives preference to resolutions, involving two initially resolved haplotypes. The application of the Inference Rule is justified as long as the result is the accurate resolution of a previously ambiguous genotype. Similarly, the rule gives preference to resolutions, involving one originally resolved haplotype, compared with involving no initially resolved haplotypes.

The "distance" of an inferred haplotype NR from the initial resolved vectors can be defined as the number of inferences used on the shortest path of inferences from some initial resolved vector, to vector NR. Such rationalization of Clark's algorithm becomes weaker, since it is used to infer the vectors with the increasing distance from the initial resolved vectors. However, Clark's Inference Rule is justified in [14] by the empirical observation of the aforementioned consistency.

2.3 Effect of Different Implementation Specifications on Clark's Algorithm

The implementation of the Inference Rule results in many options regarding vector R: the initial resolved vector, for any ambiguous vector, and any one choice, can result in a different set of future choices. Consequently, one succession of the choices might resolve in all the ambiguous vectors in one way; whereas, another implementation, involving different choices might resolve the vectors in a different way, or result in orphans because the ambiguous vectors cannot be resolved.

For example, consider two resolved vectors 0000 and 1000, and two ambiguous vectors 2200 and 1122. Vector 0000 is used to resolve vector 2200, creating the new resolved vector 1100, which is then used to resolve 1122. In this way, both of the ambiguous vectors are resolved; that is, the result is the resolved vector set 0000, 1000, 1100, and 1111. But, 2200 can also be resolved by applying 1000, resulting in resolved vector 0100. However in this case, none of the three resolved vectors, 0000, 1000, or 0100 can be further used to resolve the orphan vector 1122.

The results generated by Clark's method depend on the order of the data set. As a result, Clark suggests a reordering of the data multiple times, and running the algorithm on each reordered data set. The "best" solution among these executions is then selected.

In regards to the best solution, Clark reports that the solution with the highest number of resolved genotypes; that is, a lower number of orphans, should be preferred, since the results indicate that the inference method tended to produce incorrect vectors only when the execution also leaves orphans. This implies that if the early ambiguous genotypes are inferred, incorrectly, by the inference rule, it will render the method, unable to move further and resolve the remaining genotypes [Clark 90].

2.4 Clark's Algorithm and Maximum Resolution Problem

According to Clark, a complete haplotype resolution, based on maximum parsimony, results in a solution that is more likely to be correct and unique. This is depicted by the results obtained by the implementation of Clark's algorithm on certain empirical data. The method is proved to be reliable by comparing the results with those of the haplotypes, obtained from direct molecular methods.

This has led to a reformulation of Clark's algorithm by Gusfield as a "maximum resolution" (MR) problem. The MR problem is stated as follows:

Given a set of both homozygous and heterozygous genotypes, what is the maximum number of ambiguous genotypes that can be resolved by successive applications of Clark's inference rule on the available genotype data?

The MR problem requires that Clark's algorithm be implemented in such a way that the consequences of the earlier choices of the inference rule on the later applications of the inference rule can be weighed. Gusfield has demonstrated that the MR problem is NP-hard, and Max-SNP complete. He reformulated the MR problem as a problem on directed graphs. An exponential time reduction to a graph theoretic problem can be solved by using integer linear programming.

An acyclic graph is used to represent all the possibilities, resulting from the implementation of Clark's algorithm, where each node represents a haplotype resolution in some implementation of Clark's inference rule. Nodes u and v are connected by an edge, only if this haplotype resolution can be used to resolve an ambiguous genotype in the sample data, resulting in the inference of the haplotype at node v.

Gusfield has also showed that the MR problem can thus be formulated as a search problem on this graph, and can be solved by using integer linear programming. Gusfield's results indicate that this approach is very efficient in practice, and that linear programming alone is sufficient to solve the maximum resolution problem. However, the results also indicate that the solution of the MR problem is not an entirely suitable way to find the most accurate solutions. One major problem is that there are often many solutions to the MR problem, that is, many ways to resolve all of the genotypes. Moreover, since Clark's method should be run many times, generating many solutions, it is not clear how to use the generated results. In fact, no published evaluation of Clark's method exists, except for the evaluation by [Liu 04]. He proposed an approach to this issue. Almost all have run Clark's method only once on any given data set. This ignores the stochastic behaviour of the algorithm, and these evaluations are uninformative. The critical issue in Clark's method is how to understand and exploit its stochastic behaviour.

2.5 Variations of the Implementation of Clark's Algorithm

In Gusfield's work, different variations of the implementation of Clark's method have been studied. These variations of the rule-based approach differed in how the list of reference haplotypes can be created and maintained during the process of inferral and

how the list of ambiguous genotypes is analyzed. These variations can differ, in the genetic models with which they are consistent [Orzack 03].

They report the results of several variations of the rule-based method, including Clark's original method by using a set of 80 genotypes at the human APOE locus; of these genotypes, 47 are ambiguous with 9 SNP sites in each genotype. Table 2.1 shows the different variations they studied in [Gusfield 03]. They were designed the variations in response to following basic questions:

- Does the reference list also include inferred haplotypes in addition to the real haplotypes or not?
- Whether duplicate haplotypes are removed or retained in the reference list?
- Whether the duplicate ambiguous genotypes are consolidated or not?
- Is the reference list randomized?

Table 2.1: The characteristics of the eight rule-based algorithms for haplotype inferral.

Variation	Duplicate haplotypes		Haplotype list randomized?	Ambiguous genotypes consolidated?	Preference for real haplotypes?	Frequency preference	Identical genotypes
	Real	Inferred					
1	Retained	Retained	Yes	No	No	Population	May be resolved differently
2	Retained	Retained	Yes	Yes	No	Population	Resolved identically
2a	Removed	Retained	Yes	Yes	No	Weak	Resolved identically
2b	Retained	Retained	No	No	Yes	Population	Resolved identically
2c	Removed	Retained	No	Yes	Yes	No	Resolved identically
3	Retained	Retained	No	No	No	Strong	May be resolved differently
4a	Removed	Removed	Yes	Yes	No	No	Resolved identically
4b	Removed	Removed	No	Yes	Yes	No	Resolved identically

The real haplotype pairs are experimentally inferred in order to evaluate the inferral accuracy of each variation. Almost all variations produce a large number of different

solutions. Each is capable of resolving all of the 47 ambiguous genotypes. The solutions vary considerably with respect to accuracy. As a result, a solution, chosen at random from the solutions, is probably not very accurate. Consequently, an important issue in using rule-based methods, such as Clark's method, is how to exploit the many different solutions that it can produce.

2.6 Consensus Approach

The multiplicity of solutions demands an effort to understand how they can be used to provide a single accurate solution. Gusfield has introduced the following strategy in [Gusfield 03] to greatly improve the accuracy of any of the variations of the rule-based method.

1. Run the algorithm repeatedly, each time randomizing the order of the input data. Depending on the variation, the decisions made by the method are also randomized. The result is a set of solutions that might be quite different than another.
2. Tally the different inferrals for any given genotype across the results of all the repetitions for the given sample data. In these runs, record the haplotype pair that was most commonly used to explain each genotype g. The set of such explaining haplotype pairs is called the "full consensus" solution.
3. Or Select the runs that produce a solution using the fewest, or close to the fewest number of distinct haplotypes. In those runs, record the haplotype pair that was most commonly used to explain each genotype g. The set of such explaining haplotype pairs is called the "consensus" solution.

The consensus solution is reported to have a significantly higher accuracy than the average accuracy of the 10,000 solutions. Also, Gusfield, reports a strong negative

relationship between the numbers of haplotypes used in a solution and the correct number of inferrals, as depicted in Figure 2.1 [Orzack 03].

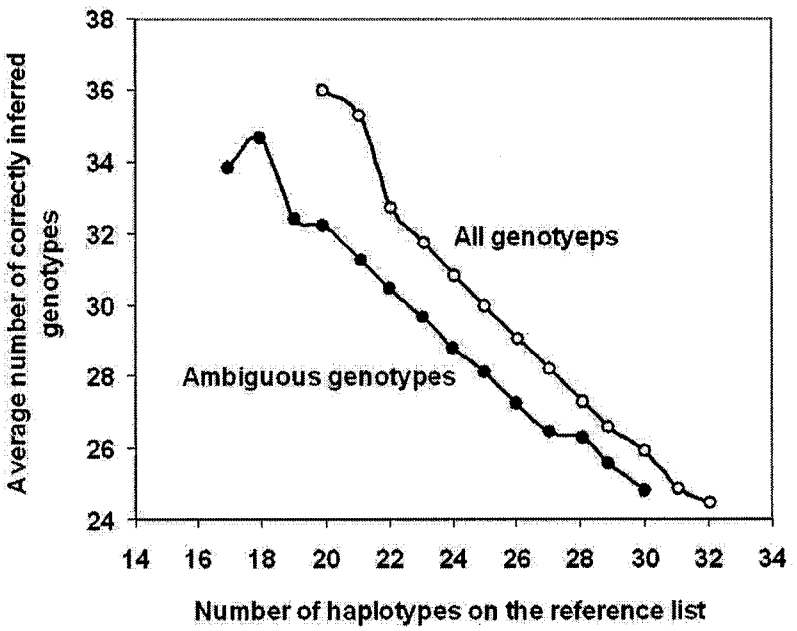

Figure 2.1: The relationship between the average number of correctly inferred haplotype pairs and the number of reference haplotypes.

Also, it is reported that the solutions that uses the smallest and next-to-smallest number of distinct haplotypes, the haplotype pairs used with higher frequency were almost always correct. For instance, genotype resolutions reported above 85% of the time in results were correct in most cases. This provides an opportunity to home in on the inferred haplotypes in results that can be used with confidence. Also, it can be used to guide the experimental efforts for the detection of true haplotypes so as to minimize the experimental effort.

Chapter 3

Proposed Methodology

This thesis presents two new and efficient parallel algorithms for Haplotype Inference based on Consensus Method proposed by Orzack and Gusfield in [Orzack 03]. One approach parallelizes the consensus method. Whereas the second parallel algorithm introduces an improved variation of consensus method. The parallel algorithms are then further used firstly to systematically investigate the affect of, number of iterations of Clark's Inference rule, on the number of accurately inferred genotypes, and secondly to explore better ways to fish out the results, with higher accuracy, from among the many results generated by numerous iterations of Clark Inference Rule.

The parallel algorithms were implemented using Message Passing Interface (MPI). The code was developed in C++. The algorithms divide the computational tasks almost equally among the number of nodes. The data set remains the same for each processor, which assists in keeping the communication overhead to a minimum. Thus ensuring maximum possible speed up, for optimal combination of number of iterations and number of processors.

The sequential algorithm used as basis for the parallel algorithms is based on Clark's Inference Rule and is briefly explained in next section.

3.1 The Sequential Algorithm

The sequential algorithm for consensus method closely follows Clark's Inference rule, and resembles variations 4a and 4b of Clark's algorithm reported by Orzack. The sequential algorithm however departs from variations 4a and b in that the duplicate genotypes are not consolidated, and thus could be either resolved identically or differently from each other.

The algorithm first reads in the input data, consisting of phase unknown ambiguous genotypes. The input data is then processed to create a Difference Matrix, an array that stores the number of site differences for each haplotype pair. The difference matrix is then used to select the haplotype pairs with either none or one site difference i.e. homozygotes or single site heterozygotes. These haplotype pairs are then used to generate the Initial Reference List.

The Clark Inference Rule is then applied repeatedly for the specified number of times to resolve the ambiguous genotypes and the results are integrated to generate the Consensus Results.

The pseudo code for the sequential algorithm is given below. Also, the sequential algorithm is depicted in Figure 3.1.

Algorithm:

Read input data
Generate difference matrix
Generate reference list
Generate list of ambiguous genotypes

While less than iterations
  Randomize haplotype and ambiguous genotypes lists (only for variation 4a)
  Perform Clark's Inference Rule
Generate final consensus results

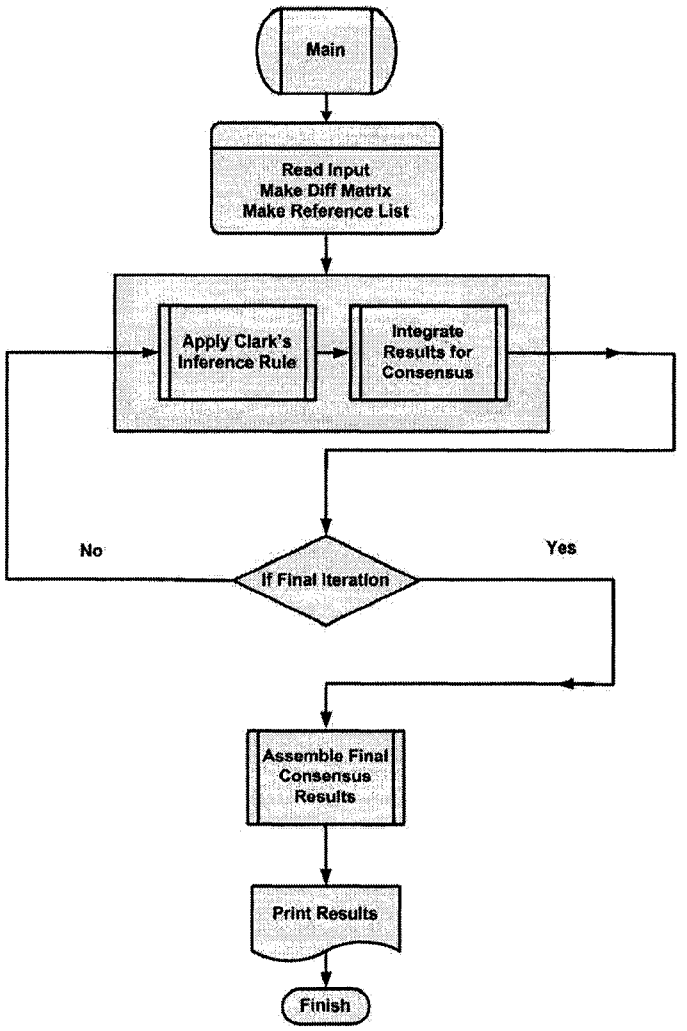

Figure 3.1: Flowchart showing sequential algorithm process.

3.2 Parallel Algorithm For Consensus Method

3.2.1 Computational Task Distribution

As Consensus method is based on the stochastic behavior of the Clark's algorithm, it requires a large number of iterations of Clark's algorithm. The new algorithm therefore exploits the parallelism in Consensus method by dividing the required number of iterations of Clark's algorithm among the available nodes. Each node thus first performs the assigned number of repetitions of Clark's algorithm on the given data, compiles the

results and then performs consensus method calculations on the available partial data. Each node then passes on its results to the master node, where the results are compiled and consensus method computations are finalized to give complete consensus results.

3.2.2 Data Distribution

As the accuracy of the results can be affected by the earlier inference made by the algorithm, thus dividing the data among the nodes would result in much higher need for information interchange between them. Thus in order to minimize the communication costs and even distribution of computation among the nodes, each node is provided with complete data set.

3.2.3 Structure of Parallel Algorithm

The structure of the proposed parallel algorithm is based on master slave architecture. The master node is responsible for generating the computational tasks and collection of results from the slaves, while the slaves execute the tasks issued to them and take care of the major portion of the computational load, as shown in Figure 3.2.

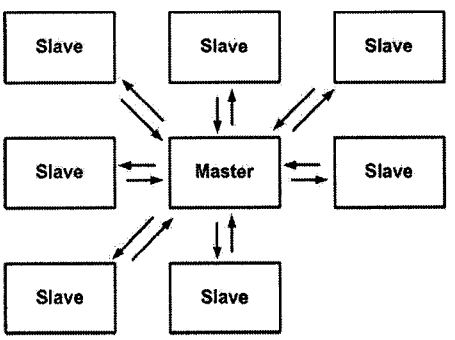

Figure 3.2: Master/Slave architecture.

First each node reads in the phase unknown genotype data from the input file. The phase unknown genotype data is then processed to generate the difference matrix and the Initial reference list of unambiguous haplotypes, and a list of ambiguous genotypes.

Master Node

First the master node reads in the phase unknown genotype data from the input file, as well as other specifications such as number of iterations of Clark's algorithm (before the consensus solution is generated), name of the output file if needed to be specified etc.

After generating the Difference Matrix, the Reference list and ambiguous genotypes list, the master node then splits the specified number of iterations of Clark's algorithm among the slave nodes and also sends them the their assigned task. After each slave node completes its computational calculations, master node receives the results generated by the slaves and integrates and compiles them together to manufacture the final consensus results.

Slave Nodes

Each slave node receives its assigned task from the master node i.e. the number of iterations of Clark's algorithm the node has to perform. Each, iteration of Clark's algorithm, generates a result, containing the resolved genotypes for each individual. After, each iteration the generated results are integrated into the previous ones. Once, the assigned number of iterations, of Clark's algorithm are achieved, the compiled results are used to generate partial consensus results which are then sent back to master for final assembly.

The pseudo code used for parallel algorithm for consensus method is systematically provided below:

Algorithm:

If master
Read input data
Get number of processors (nproc)
Generate difference matrix
Generate reference list
Generate list of ambiguous genotypes
Divide task by dividing number of iterations by number of processors –1

While less than nproc
Send iterations (number of iterations of Clark's algorithm node has to perform)
Update loop counter

While less than nproc
Receive arrays of partial consensus results from node
Update loop counter
Generate final consensus results

If slave
Receive iterations

While less than iterations
Randomize the reference list & ambiguous genotypes list (for variation 4a only)
Perform Clark's Inference Rule
Integrate results to generate partial consensus results
Update loop counter

Update loop counter

Figure 3.3 depicts a flowchart, which gives an overview of the steps followed by the parallel algorithm; whereas, steps followed by master and slave nodes are separately shown in Figures 3.4 and 3.5.

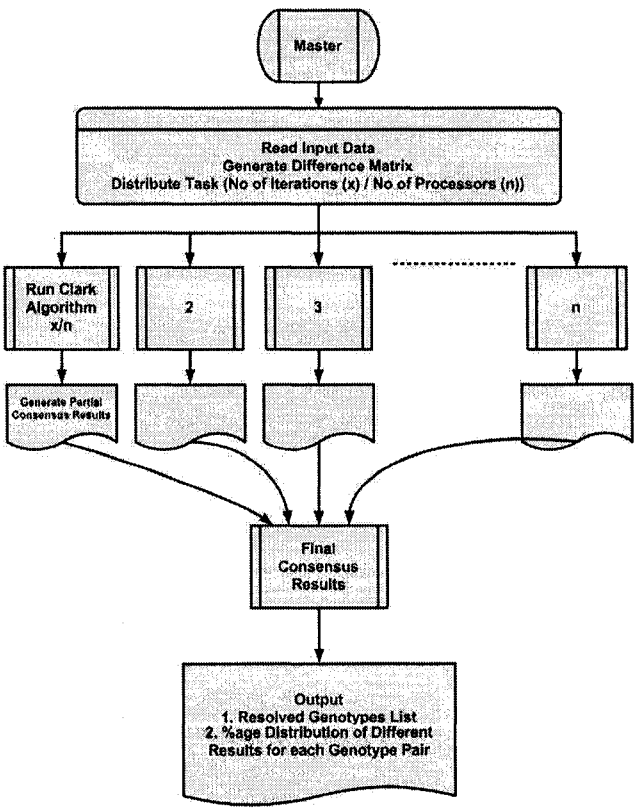

Figure 3.3: Flowchart of parallel algorithm.

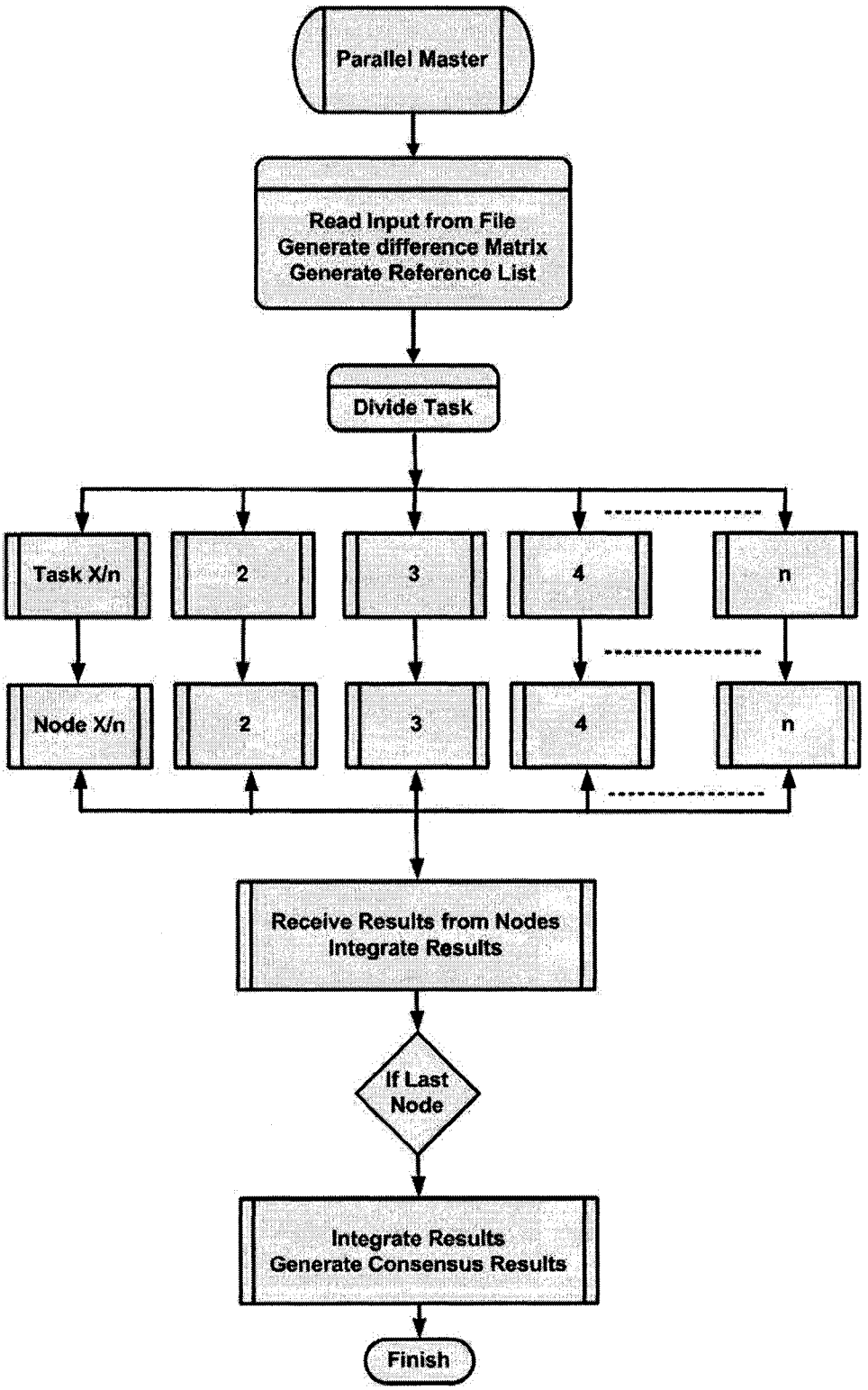

Figure 3.4: Flowchart showing steps followed by master.

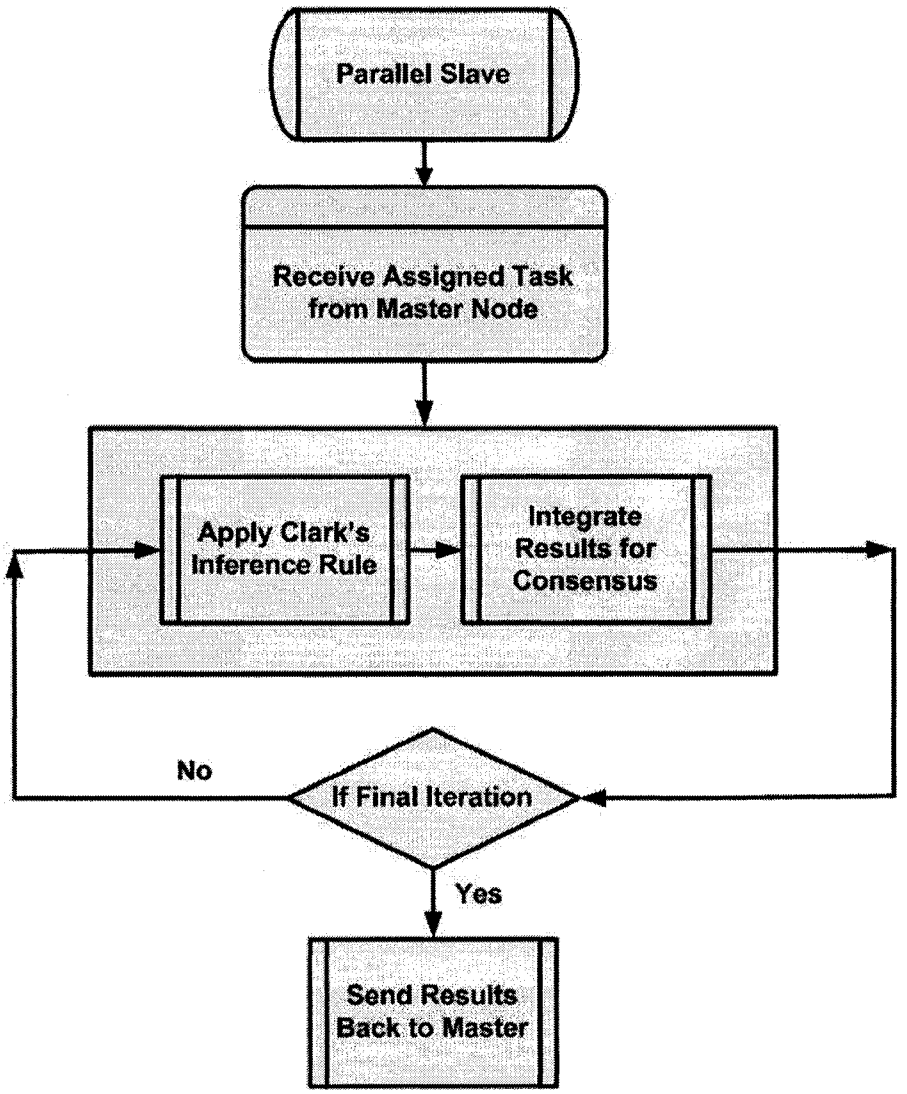

Figure 3.5: Flowchart showing steps followed by Slave.

3.3 Parallel Algorithm For Enhanced Consensus Method.

The parallel algorithm for the enhanced consensus method exploits the stochastic nature of consensus results and the computational resources made available by parallelization of the method to select the result with higher accuracy among the available results.

The parallel algorithm generates many consensus results and then applies consensus method on those results. This is explained by the pseudocode and flow chart given below.

Algorithm:

If master

Read input data
Get number of processors (nproc)
Get number of Consensus Iterations (no. of times Consensus method is to be applied)
Get number of Iterations (no. of times Clark Inference Rule is to be applied for each consensus iterations)
Generate difference matrix
Generate reference list
Generate list of ambiguous genotypes
Divide task by dividing number of iterations by number of processors –1

While less than nproc-1
Send Consensus Iterations
Send Iterations
Update loop counter
While less than nproc-1
Receive arrays of partial Enhanced Consensus results from node
Update loop counter
Generate final Enhanced Consensus results

If slave
Receive Consensus Iterations
Receive Iterations
While less than Consensus Iterations
While less than iterations
Randomize the reference list & ambiguous genotypes list (for variation 4a only)
Perform Clark's Inference Rule
Integrate results to generate partial consensus results
Update loop counter
Integrate results to generate partial Enhanced Consensus results
Update loop counter

Send the partial Enhanced Consensus results to master node.

3.4 Problem Statement

Promising results were reported by Orzack and Gusfield for different variations of Clark algorithm coupled with Consensus method in [Orzack 03]. However consensus methods demands further investigation. The proposed parallel algorithms not only increase the time efficiency of the consensus method they provide an opportunity to further investigate consensus method for instance the affect of different number of iterations on the consensus results. Also it provides the opportunity to study different variations of Consensus method such as Enhanced Consensus method introduced in this thesis to help distinguish the results with higher accuracy from the available results.

Chapter 4

Experimental Results and Discussion

This chapter summarizes the performance results for parallel consensus algorithm, and parallel enhanced consensus algorithm. First objective behind parallelization of consensus method is to decrease the required computational time thus making it feasible to analyze larger data sets with optimal number of iterations of Clark's inference rule. The second objective is to study, if the number of iterations of Clark's inference rule run before applying the consensus method affected the inference accuracy of the consensus method. Finally, the last objective is to investigate, if parallel algorithm can be used, to pick out the results with better accuracy among many available, in shorter period of time.

In literature, different criteria are used for evaluation of parallel algorithms including, efficiency analysis, speedup and analysis of inefficiencies caused by communication overhead and unequal load balancing etc. The parallel algorithms are tested both for time efficiency and accuracy of the results. In this section, we discuss both parallel algorithms in light of the aforementioned key issues.

4.1 Experimental Environment

The parallel algorithms were implemented using SHARCNET (Shared Hierarchical Academic Computing Network). SHARCNET is a grid of high performance interconnected clusters, which spans eleven leading academic institutions across southern

Ontario and provides support for leading-edge research. The algorithms were compiled and run mainly on narwhal, megaladon, tiger, bala, and bruce clusters. The system is ideal for serial/parallel MPI code development and large computations. All three algorithms, serial algorithm for consensus method and the parallel algorithms for consensus and enhanced consensus methods, were implemented and run in above environment and results are based on that.

4.2 Experimental Data

The experimental data set consists of unambiguous genotype pairs for 80 individuals. There are a total of 17 underlying haplotypes involving 9 experimentally determined polymorphic sites. Among 80 individuals 33 individuals have unambiguous genotype pairs, so that 11 out of 17 real haplotypes can be inferred from unambiguous genotypes. Whereas 47 individuals have ambiguous genotype pairs, Table 4.1 shows grouping of ambiguous pairs depending on the number of polymorphic sites.

Table 4.1: Grouping of ambiguous genotypes.

Number of SNPs	Number of Individuals
2	17
3	20
4	6
5	4

Genotype pairs for all 80 individuals were experimentally determined and details available in [Orzack 03]. The data set can be obtained at http://www.genetics.org.

4.3 Parallel Consensus Algorithm

Table 4.2 shows the runtime for the sequential as well as the new parallel consensus algorithm in seconds. For each value the algorithm was run 5 times and the average is obtained, along with the standard deviation for each set of values. The small standard deviation indicates the narrow distribution of data.

Also Figure 4.1 and Figure 4.2 show probability density function for data sets for two values of Table 4.2 with higher data dispersion (as indicated by standard deviations). The coefficient of variation for Weibull distribution depends on beta, the shape factor, and gives a relative measure of data dispersion compared to the mean. The beta values for both plots are very high indicating small variations in the data.

Table 4.2: Runtimes for sequential and parallel consensus algorithms.

Number of Iterations		Number of Processors with Elapsed Time (sec)										
		1	**2**	**3**	**5**	**9**	**17**	**33**	**65**	**129**	**257**	**512**
10^2	AVG	0.15	1.22	1.15	1.32	1.28	1.31	1.22	1.39	-	-	-
	STDV	0.002	0.015	0.015	0.121	0.159	0.244	0.142	0.241	-	-	-
10^3	AVG	1.43	2.52	1.84	1.65	1.52	1.28	1.32	1.56	2.11	3.91	4.80
	STDV	0.036	0.070	0.028	0.183	0.011	0.110	0.067	0.312	0.092	0.812	0.594
10^4	AVG	14.7	16.37	8.60	5.40	3.47	2.52	1.97	1.66	2.95	4.11	5.71
	STDV	0.521	0.029	0.029	0.170	0.147	0.068	0.088	0.215	0.911	1.112	0.534
10^5	AVG	146.30	148.32	77.11	39.55	19.61	10.93	5.98	4.71	4.12	6.32	8.34
	STDV	4.140	6.428	2.115	1.334	1.209	0.080	0.184	0.511	0.932	0.889	1.548
10^6	AVG	1458.69	1466.16	745.87	375.42	196.20	96.74	51.97	26.1	15.32	9.42	15.32
	STDV	32.679	28.038	8.540	2.831	3.776	0.378	0.782	0.382	0.522	0.881	0.490
10^7	AVG	14558.71	14661.49	7750.09	3864.98	1887.81	965.09	480.12	254.12	146.91	119.18	162.91
	STDV	333.214	13.021	12.812	4.308	9.453	7.213	5.431	3.211	3.932	1.129	3.564

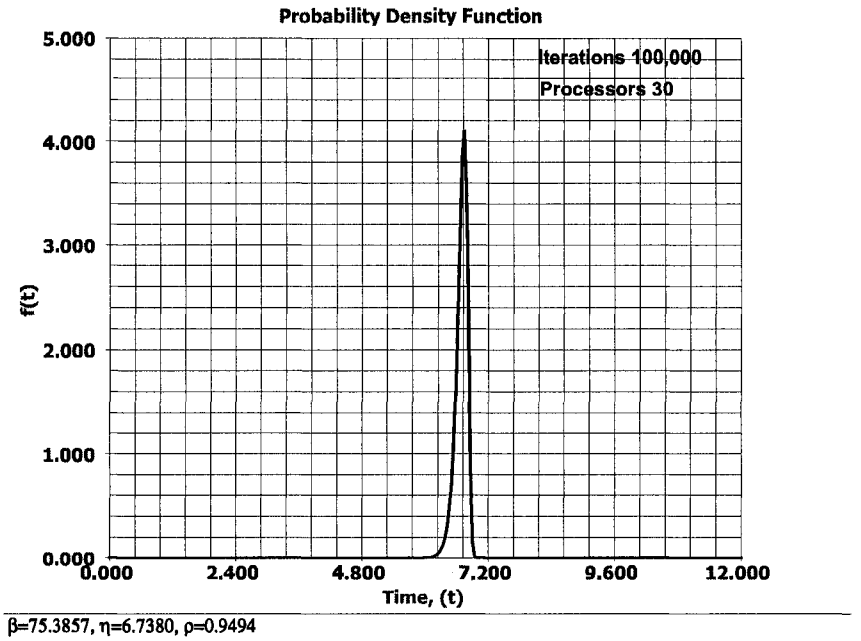

Figure 4.1: Weibull Probability Density Distribution for data set of runtime values for consensus method with 100,000 iterations and 33 processors.

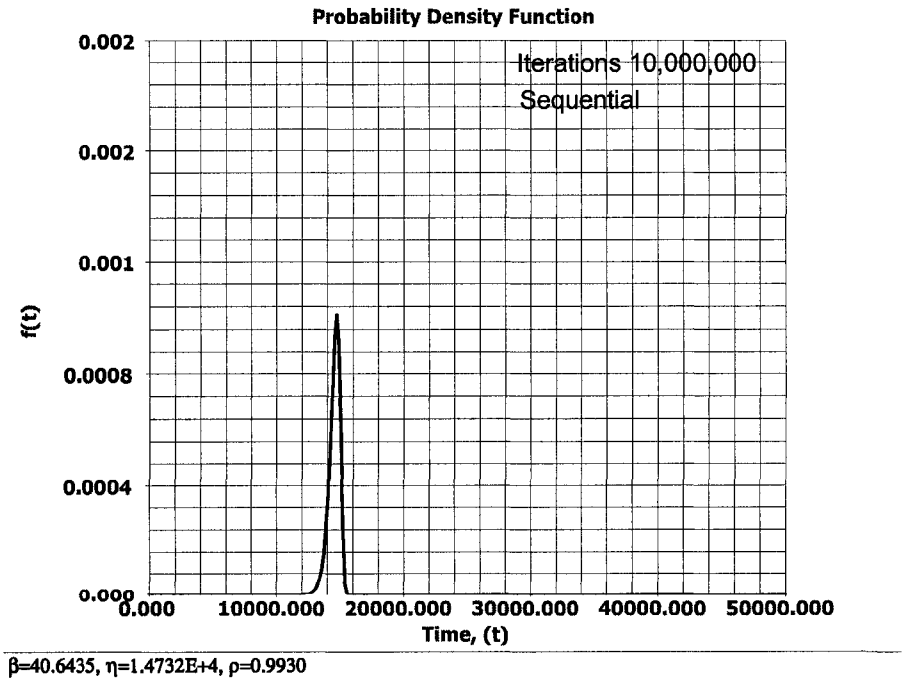

Figure 4.2: Weibull Probability Density Distribution for data set of runtime values for consensus method with sequential 10,000,000 iterations.

The values in Table 4.2 indicate that in case of sequential algorithm, which benefits from a single universal abstract machine model (the RAM), the runtime shows a linear growth with increase in the number of iterations. The runtimes for parallel algorithm show only a slight increase in time for mapping from sequential to parallel approach, as depicted in Figure 4.3. The drop in runtime for a given number of iterations becomes more pronounced with increase in number of iterations as shown in Figure 4.4 and Figure 4.5.

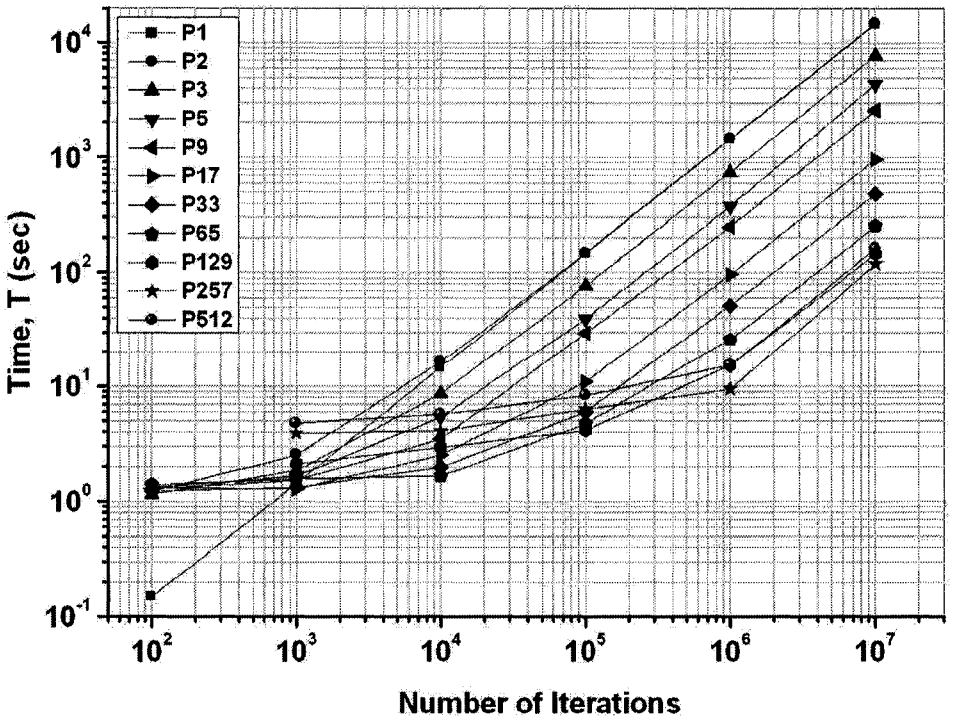

Figure 4.3: Runtimes for parallel algorithm with different processors.

The runtime decreases for a given number of iterations until an optimal increase in number of slave processors, after which decrease in runtime reaches a plateau and starts to increase as the increase in runtime due to communication overhead becomes greater than time saved due to division of task as shown in Figure 4.4.

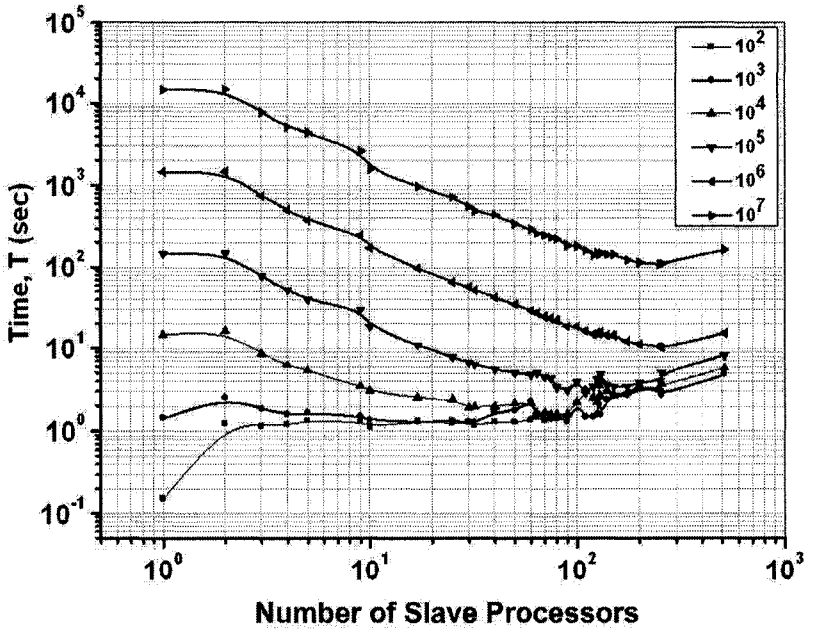

Figure 4.4: Runtimes for parallel algorithm for different number of iterations.

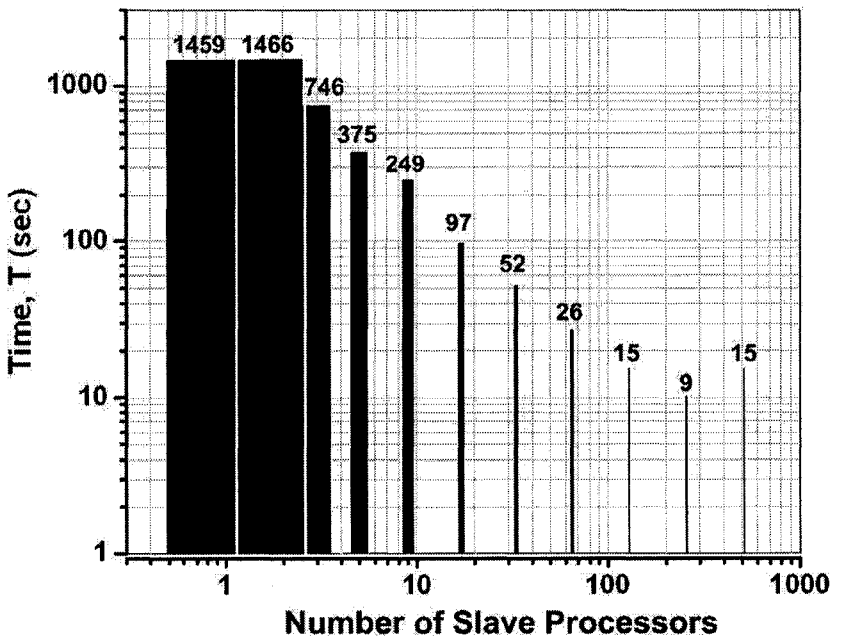

Figure 4.5: Decrease in runtime with increase in number of processors.

4.3.1 Efficiency and Speedup Analysis for Parallel Consensus Algorithm

Efficiency is a performance metric, defined as an estimate of how well utilized the processors are in solving the problem, compared to efforts wasted in communication and synchronization. Typically, it is a value between zero and one. Algorithms with linear speedup and algorithms running on a single processor have an efficiency of 1, whereas efficiency approaches zero for inherently serial problems with increase in number of processors. Given p number of slave processors, χ number of iterations, T_s the execution time of the sequential algorithm, T_p the execution time of the parallel algorithm with p processors efficiency $\in$ is calculated as follows:

$$\varepsilon = \frac{T_S(x)}{p \times T_P(x)} \tag{4.1}$$

Using the results obtained from parallel consensus algorithm its efficiency is evaluation is shown in Figure 4.6. In general, the efficiency values are very high, 0.96; however, as shown in Figure 4.6, for a task of given size, there is a maximum increase in efficiency for certain number of processors after which efficiency starts decreasing.

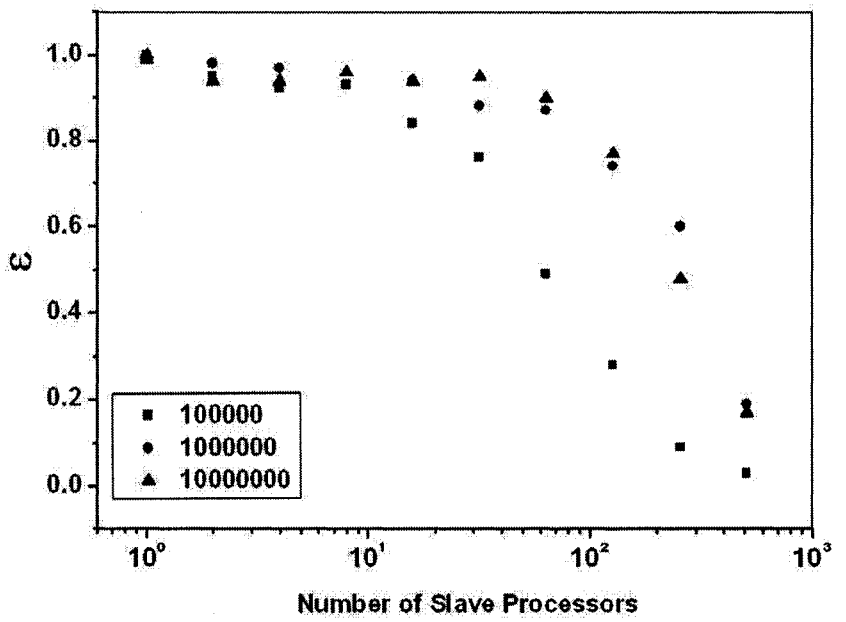

Figure 4.6: Efficiency diagram for parallel consensus algorithm.

Speedup is defined as the factor by which parallel algorithm is faster than the corresponding sequential algorithm. Given p number of slave processors, χ number of iterations, T_s the execution time of the sequential algorithm, T_p the execution time of the parallel algorithm with p processors, speedup S_p can be calculated as below:

$$S_x = \frac{T_S(x)}{T_P(x)} \tag{4.2}$$

Results for speedup evaluation S_x for parallel consensus method is shown in Figure 4.7. Evaluations results show that this problem is inherently parallelisable and the speedup increases linearly with number of processors. There is an optimal increase in number of processors for each given number of iterations, which achieves maximum speedup. Further increase in number of processors although causes a decrease in runtime however overall efficiency starts to drop, for instance the optimal number of processors for 100000 iterations is 128, whereas for 1000000 and 10000000 iterations the optimal number of processors is 256, as shown in Figure 4.7. This drop in efficiency/speedup is caused due to smaller saving of time due to division of task among the processors, as compared to the increase in time due to the communication overhead.

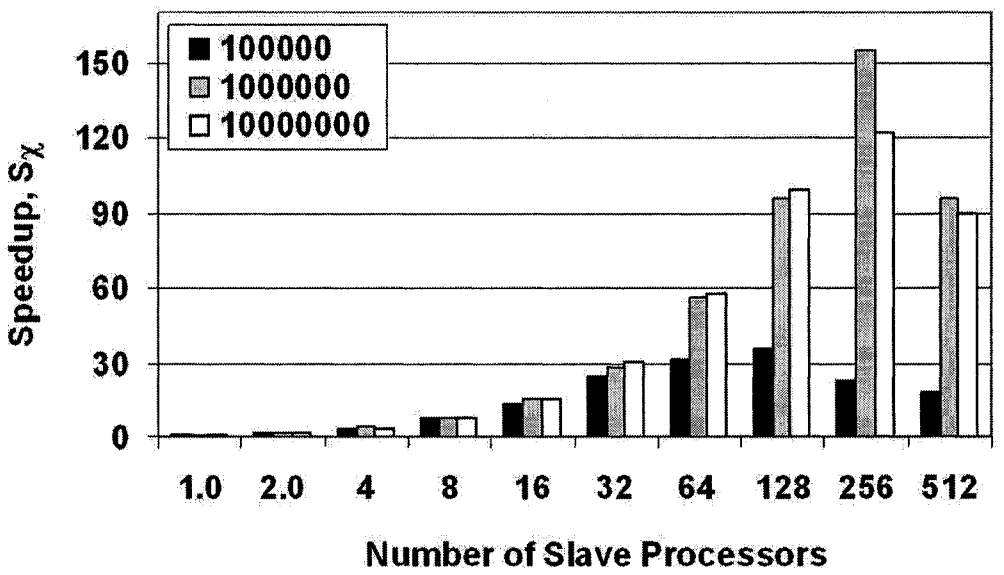

Figure 4.7: Speedup diagram for parallel consensus algorithm.

4.3.2 Analysis of Consensus Results for Parallel Consensus Algorithm

Average accuracy for the Consensus results for parallel consensus algorithm is similar to the average accuracy of the sequential algorithm. As shown in Figures 4.8 – 4.10 the average value of the correctly inferred genotypes with consensus method in general remains fairly consistent regardless of the increase in number of slave processors, except for very large number of slave processors as compared to the number of iterations.

Parallel Consensus algorithm was also employed to investigate if consensus results were affected by number of iterations of Clark algorithm run before consensus method was applied. A consistent increase in accuracy of consensus results (i.e. number of correctly inferred genotypes) was observed with increase in number of iterations, however a threshold value was observed at 10,000 iterations, after which an increase in the computation time for consensus method was observed without any increase in average accuracy of the consensus results.

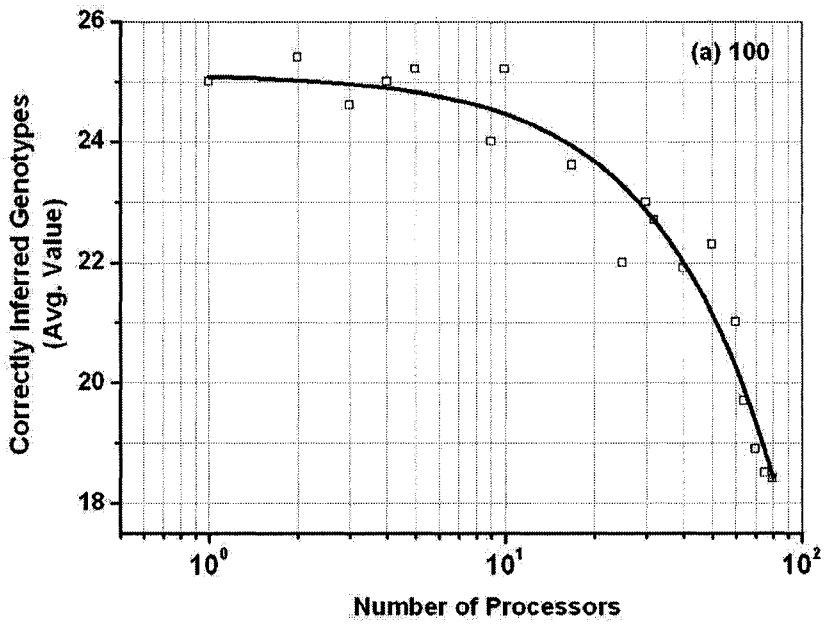

Figure 4.8: Accuracy of Consensus Results (100 iterations) of parallel consensus method with different number of processors.

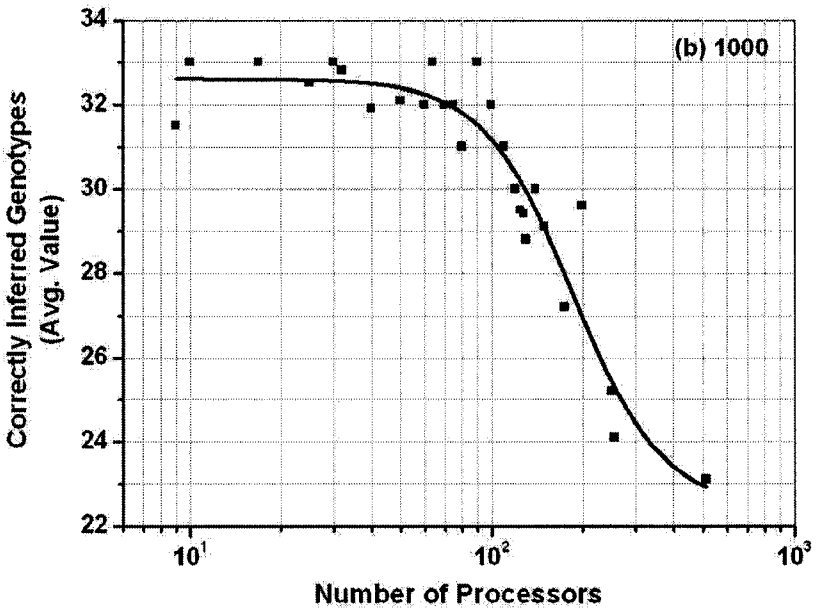

Figure 4.9: Accuracy of Consensus Results (1000 iterations) of parallel consensus method with different number of processors.

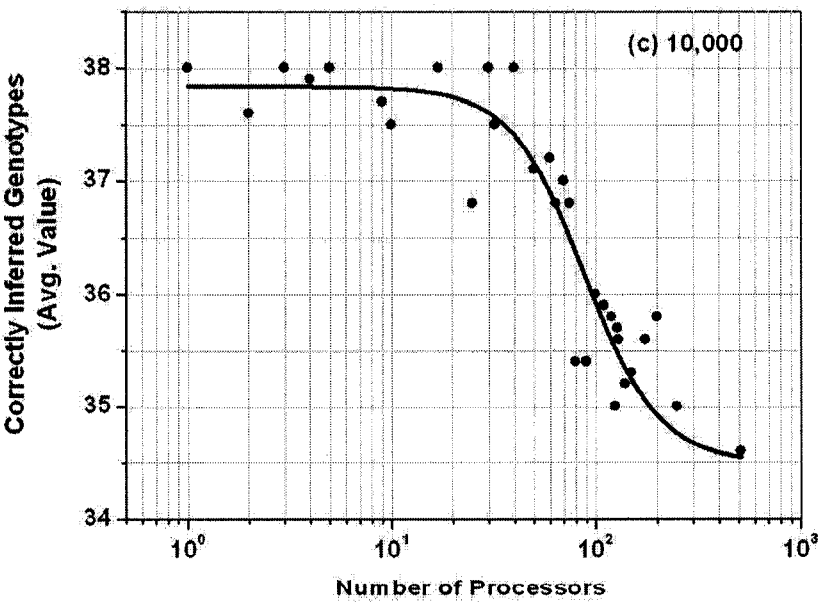

Figure 4.10: Accuracy of Consensus Results (10,000 iterations) of parallel consensus method with different number of processors.

4.4 Parallel Enhanced Consensus Algorithm

Table 4.3 shows the runtime for the new parallel algorithm for enhanced consensus method. The runtimes for enhanced consensus algorithm are comparable with those of parallel consensus algorithm. For instance runtime for parallel consensus algorithm with 10,00,000 iterations and one slave processor is approximately 1458.69 sec, whereas the runtime for parallel enhanced consensus algorithm for 1000 iterations and 1000 Consreps (which is equivalent to 1000000 iterations) is approximately 1001.33 sec.

Table 4.3: Runtime for parallel enhanced consensus algorithm.

Iterations	Consreps	Function	Number of Processors with Elapsed Time, T (sec)									
			1	5	25	50	75	100	125	150	175	200
1000	100	AVG	101.22	21.19	4.35	2.29	24.77	2.38	-	-	-	-
		STDV	1.16	1.69	0.22	0.10	1.08	0.09	-	-	-	-
1000	1000	AVG	1001.33	211.81	42.21	21.34	36.35	12.29	9.41	117.25	130.83	7.84
		STDV	8.72	6.54	1.10	1.02	1.22	1.01	0.76	3.50	3.02	0.32
1000	10000	AVG	10091.21	2114.51	463.35	214.23	161.27	114.34	88.21	164.88	86.657	60.05
		STDV	16.22	11.23	4.82	3.43	2.10	3.25	2.22	3.20	2.95	1.52

The trend for decrease in runtime for parallel enhanced consensus algorithm for a given number of iterations and Consreps is same as that of parallel consensus algorithm. Runtime decreases until an optimal increase in number of slave processors, after which decrease in runtime reaches a plateau and starts to increase as the increase in runtime due to communication overhead becomes greater than time saved due to division of task as shown in Figure 4.11 for iteration with 100 Consreps.

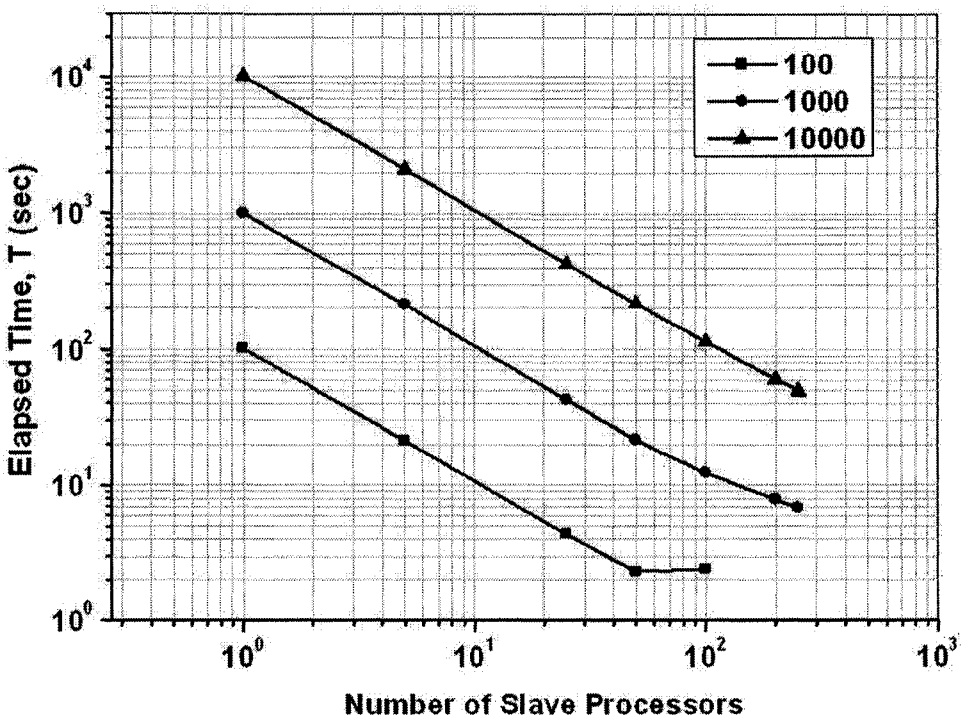

Figure 4.11: Runtimes for parallel enhanced algorithm with different number of processors.

4.4.1 Efficiency and Speedup Analysis for Parallel Enhanced Consensus Algorithm

Using the results obtained from parallel enhanced consensus algorithm its efficiency evaluation is shown in Fig 4.12. In general the efficiency values are very high, however as shown in Figure 4.12, for a task of given size, there is a maximum increase in efficiency for certain number of processors after which efficiency starts decreasing.

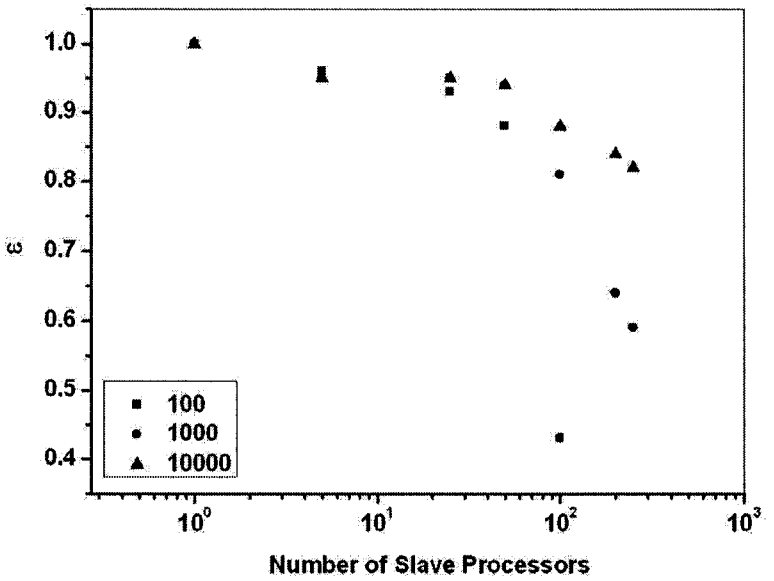

Figure 4.12: Efficiency diagram for parallel enhanced consensus algorithm.

Results for speedup evaluation S_x for parallel enhanced consensus method is shown in Fig 4.13. Evaluation results show that this problem is well suited for parallelization and the speedup increases linearly with number of processors.

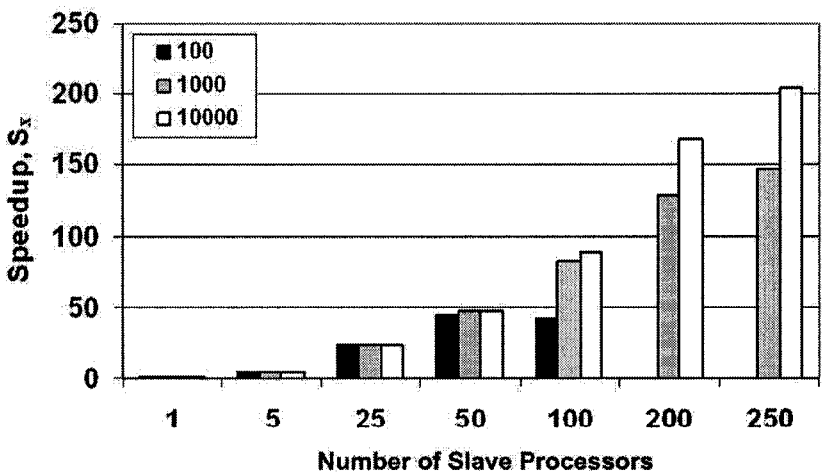

Figure 4.13: Speedup diagram for parallel enhanced consensus algorithm.

4.4.2 Analysis of Consensus Results for Parallel Enhanced Consensus Algorithm

The data listed in Table 4.4, and shown in Figures 4.14 and 4.15, the average accuracy of enhanced consensus results is consistently better than that of average consensus results, and closer in accuracy to the best consensus results among total number of iterations. The Enhanced Consensus algorithm, thus allows selection of better and more accurate result in economical time.

The enhanced consensus algorithm was tested with different number of iterations and consreps, however a threshold value was observed at 10,000 number of iterations and 1,0000 number of consreps, after which an increase in the computation time for consensus method was observed without any increase in average accuracy of the enhanced consensus results.

Table 4.4: Enhanced consensus results for variations 4a.

Enhanced Consensus for Variation 4a				
# of Procs.	**Iterations 1000 & Consreps 1000**		**Iterations 1000 & Consreps 10,000**	
	Avg of Cons	**ConsCons**	**Avg of Cons**	**ConsCons**
25	33.0	36.0	34.3	35.0
50	31.4	39.0	34.1	37.0
75	34.1	39.0	34.1	39.0
100	30.9	36.0	33.8	39.0
125	28.2	34.0	33.6	36.0
150	28.8	37.0	33.5	36.0
175	26.3	37.0	33.3	38.0
200	27.2	33.0	33.3	36.0
250	25.8	34.0	32.9	39.0

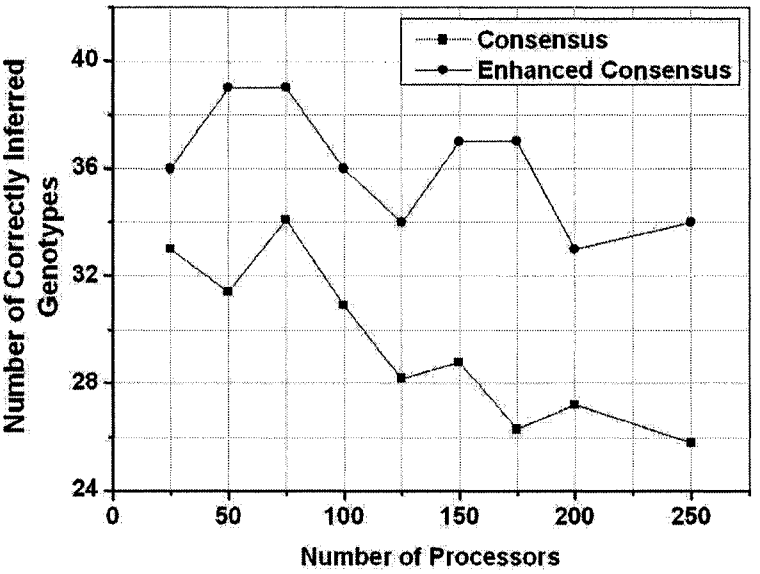

Figure 4.14: Enhance consensus results for variation 4a

(Iterations 1000 & Consreps 1000).

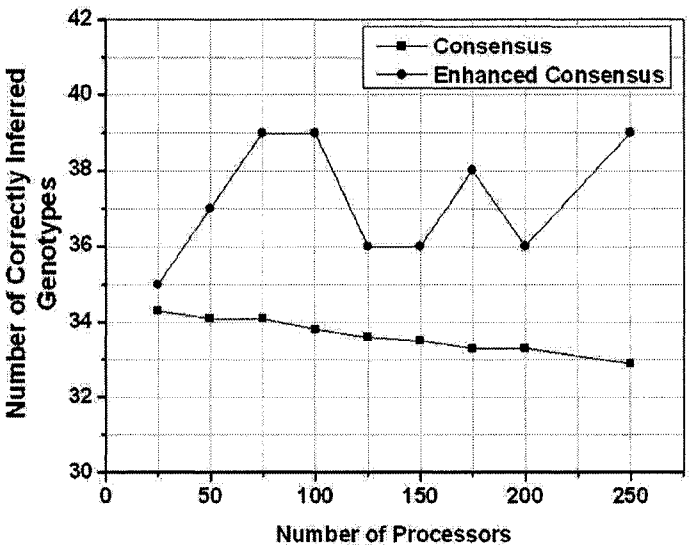

Figure 4.15: Enhance consensus results for variation 4a

(Iterations 1000 & Consreps 10,000).

Chapter 5

Conclusion and Future Directions

Importance of haplotype inference in linkage disequilibrium and numerous other studies, in addition to the high cost and difficulty of their experimental determination, has continually driven the development of computational methods for haplotype determination. Clark's Inference Rule is one of the simpler and well-known approaches for haplotype inference for groups of unrelated individuals. When combined with consensus method approach introduced in [Orzack 03] Clark's algorithm's results are comparable to the results of some of the best computational methods available for haplotype inference [Orzack 03], [Stephens 01].

This thesis implements two new parallel algorithms based on consensus method. First approach parallelizes the consensus method; whereas, the second parallel algorithm implements an enhanced consensus method that improves upon the accuracy of the consensus results. Experimental evaluation, of both algorithms indicates high scalability and good speedup/efficiency on different SHARCNET clusters. The runtimes for parallel algorithm show only a slight increase for mapping from sequential to parallel approach. The communication cost is very small, as each slave processor communicates only twice with master processor, once to receive its assigned task and second to send the results back to master processor. There is no communication among the slave processors.

Further, the results obtained using parallel algorithm indicated that on average the accuracy of consensus results improves with increase in the number of iterations until a

threshold value is reached. Any further increase in number of iterations increases the computational time without significant improvement in the quality of results. This is because any further improvement then requires more refined selective methods. The average accuracy of the parallel consensus method remains fairly consistent with increase in the number of slave processors is increased. However, as the number of slave processors increases beyond a threshold value, there is a noticeable decrease in the accuracy of the results. The threshold value depends upon the ratio of number of iterations to the number of slave processors, and is generally observed at very high number of slave processors, that causes the task assigned to each processor to become very small.

The new parallel enhanced consensus method shows promising results, both in terms of time efficiency and quality of results. As shown by the results in the previous section, the accuracy of the results generated by parallel consensus method is consistently higher than the average consensus results. This is an important advantage considering that the important task in the analyses and integration of the multiple results generated by a stochastic rule based algorithm is the ability to consistently and confidently generate more accurate results from the numerous results available.

Future Research Directions

The results presented in this thesis indicate that the parallel algorithms can be successfully used, to further refine the consensus approach, and to obtain more accurate results.

The parallel algorithms presented in this thesis are based on variation 4a of Clark algorithm described in [Orzack 03], as this variation closely follows original Clark's Inference rule. However, both parallel approaches can be easily extended to study the other variations explained in [Orzack 03]. They can also be applied easily to study other novel variations of Clark's algorithm based on other genetic models.

The parallel approach can also be employed to investigate the stochastic behavior of other promising algorithmic inferral methods, such as, the method of [Stephens 01] as

any algorithmic inferral method can potentially generate different solutions. Thus advanced techniques for integration of contending solutions become necessary.

Finally although the parallel approach combined with consensus method shows great potential and it has been shown that the optimal result is possible for some variations of Clark algorithm [Orzack 03]. However, we were unable to retrieve the optimal result with confidence and reliability, something that is, for now, possible only with experimental results.

This failure urges continuing research to best implement current methods to achieve optimal performance and to develop advanced techniques that allow best possible integration of available results, to generate results that are comparable to experimental results, for haplotype inference, in terms of inferral accuracy.

Bibliography

[Adkins 04] R. M. Adkins, "Comparison of the Accuracy of Methods of Computational Haplotype Inference using a Large Empirical Dataset" BMC Genetics, vol. 5, No. 22, 2004.

[Bafna 02] V. Bafna, Gusfield Dan, Lancia Giuseppe, and Yooseph Shibu, "Haplotyping as Perfect Phylogeny: A direct approach (Technical Report)" US Davis Computer Science Technical Report (CSE- 2002-21), 17, July 2002.

[Bafna 03] V. Bafna, D. Gusfield, G. Lancia, and S. Yooseph, "Haplotyping as Perfect Phylogeny: A Direct Approach," Journal of Computational Biology, vol.10, No. ¾, pp. 323-340, 2003.

[Barzuza 04] T. Barzuza, S. Beckmann Jacques, Shamir Ron, and Pe'er Itsik, "Computational Problems in Perfect Phylogeny Haplotyping: Xor-Genotypes and Tag SNPs" Springer-Verlag, Berlin, Heidelberg, pp. 14-31, 2004.

[Beaumont 04] M. A. Beaumont, and B. Rannala, "The Bayesian Revolution in Genetics" *Nature Reviews Genetics*, vol. 5, pp. 251 –261, 2004.

[Bonozzoni 03] P. Bonizzoni, G. D. Vedova, R. Dondi, and J. Li, "The Haplotyping Problem: An Overview of Computational Models and Solutions," *J. Comp. Sci. Tech.* vol. 18, no. 6, pp. 675-688, 2003.

[Casey 03] W. Casey, and M. Bud, "A Nearly Linear-Time General Algorithm for Bi-Allele Haplotype Phasing" International Conf. On HiPC, Dec. 2003.

[Chakravarti 98] A. Chakravarti, "It's raining SNPs; hallelujah?" *Nature Genetics*, vol. 19, pp. 216-217, Jul 1998.

[Chung 03a]. R. H. Chung, and D. Gusfield, "Perfect Phylogeny Haplotyper: Haplotype Inferral Using a Tree Model" Bioinformatics, 19 (6): 780, 2003.

[Chung 03b] R. H. Chung, and D. Gusfield, "Empirical Exploration of Perfect Phylogeny Haplotyping and Haplotypers" Proceedings of the 2003 Cocoon Conference, July. 2003.

[Clark 90] A. G. Clark, "Inference of Haplotypes from PCR-amplified Samples of Diploid Populations" *Mol. Biol.* vol. 7, No. 2, pp.111-122, 1990.

[Clark 01] V. J. Clark, Metheny Noah, Dean Michael, and J. Peterson Raymond, "Statistical estimation and pedigree analysis of CCR2-CCR5 haplotypes" Hum Genet vol.108, pp. 484–493, 2001.

[Clark 04] A.G. Clark, E. T. Dermitzakis, and S. E. Antonarakis, "Trisomic Phase Inference" LNBI 2983, pp. 1-8, 2004.

[Daly 01] M. J. Daly, Rioux, D. Schaffner S. F. J., J. Hudson Thomas, and S. Lander Eric, "High-resolution haplotype structure in the human genome" Nature Genetics, vol. 29, October 2001.

[Ding 03] C. Ding, and C. R. Cantor, "Direct molecular haplotyping of long-range genomic DNA with M1-PCR" *PNAS* vol. 100 No. 13, pp. 7449–7453, June 2003.

[Doi 03] K. Doi, J. Li, and T. Jiang, "Minimum-Recombinant Haplotype Configuration on Tree Pedigrees" Proc. WABI'03, pp. 339-353, 2003.

[Eskin 04a] E. Eskin, E. Halperin, and R. Karp, "Large Scale Reconstruction of Haplotypes from Genotype Data." In Proceedings of the Seventh Annual International Conference on Research in Computational Molecular Biology (RECOMB-2003), San Diego, CA: March 27-31, 2004.

[Eskin 04b] E. Eskin, E. Halperin, and R. Sharan, "Optimally Phasing Long Genomic Regions using Local Haplotype Predictions" In Proceedings of the Second RECOMB Satellite Workshop on Computational Methods for SNPs and Haplotypes, Pittsburg, PA: February 20-21, 2004.

[Excoffier 95] L. Excofier, and M Slatkin, "Maximum-Likelihood Estimation of Molecular Haplotype Frequencies in a Diploid Population" Mol. Biol. vol.12 (5): pp. 921-927, 1995.

[Fallin 00] D Fallin, and N. J. Schork, "Accuracy of Haplotype Frequency Estimation for Biallelic Loci, via the Expectation-Maximization Algorithm for Unphased Diploid Genotype Data" Am. J. Hum. Genet. vol. 67, pp. 947–959, 2000.

[Francis 03] S. C. Francis, M. Michael, and P. Aristides, "The Human Genome Project: Lessons from Large-Scale Biology", Science 11, vol. 300, no. 5617, pp. 286 – 290, 2003.

[Gusfield 91] D. Gusfield, "Efficient Algorithms for Inferring Evolutionary Trees" Networks 21: 19&emdash; 28. 1991.

[Gusfield 00] D. Gusfield, "A Practical Algorithm for Optimal Inference of Haplotypes from Diploid Populations," ISMB, pp. 183-189, 2000.

[Gusfield 01] D. Gusfield, "Inference of Haplotypes from Samples of Diploid Populations: Complexity and Algorithms," Journal of Computational Biology, vol. 8, No. 3, pp. 305-323, 2001.

[Gusfield 02] D. Gusfield, "Haplotyping as Perfect Phylogeny: Conceptual Framework and Efficient Solutions," *RECOMB*, 166-175, 2002.

[Gusfield 03] D. Gusfield, "Haplotype Inference by Pure Parsimony" Conference CPM, June 25-27. 2003.

[Gusfield 04] D. Gusfield, "An Overview of Combinatorial Methods for Haplotype Inference," *A survey of combinatorial algorithms and software for haplotype inference developed at UC Davis,* 2004.

[Hajiaghayi 05] M. T. Hajiaghayi, K. Jain, K. Konwar, C. Lau, Mandoiu I. I. I., A. Russell, A. Shvartsman, and V. V. Vazirani, "The Minimum k-Colored Subgraph Problem in Haplotyping and DNA Primer Selection" To be submitted to APPROX, 2005.

[Halldorsson 03] B. V. Halldorsson, V. Bafna, N. Edwards, R. Lippert, S. Yooseph, and S. Istrail, "Combinatorial Problems Arising in SNP and Haplotype Analysis," *In DMTCS*, pp. 26-47, 2003.

[Halldorsson 04] B. V. Halldorsson, V. Bafna, N. Edwards, R. Lippert, S. Yooseph, and S. Istrail, "A Survey of Computational Methods for Determining Haplotypes" *LNBI 2983*, pp. 26–47, 2004.

[Halperin 04a] E. Halperin, and E. Eskin, "Haplotype reconstruction from genotype data using Imperfect Phylogeny" *Bioinformatics* vol. 20, No. 12, pp. 1842–1849, 2004.

[Halperin 04b] E. Halperin, and R. M. Karp, "Perfect Phylogeny and Haplotype Assignment" Proceedings of the eighth annual international conference on Computational molecular biology table of contents, pp. 10-19, 2004.

[Hoh 03] J. Hoh, F. Matsuda, X. Peng, D. Markovic, M. G. Lathrop, and J. Ott, "SNP haplotype tagging from DNA pools of two individuals" BMC Bioinformatics, vol. 4, No. 14, 2003.

[Istrail 97] S. Istrail, G. G. Sutton, L. Florea, A. L. Halpern, C. M. Mobarry, Lippert Ross, B. Walenz, H. Shatkay, I. Dew, R. Miller Jason, J. Flanigan Michael, J. Edwards Nathan, Bolanos Randall, Fasulo Daniel, V. Halldorsson Bjarni, Hannenhalli Shridhar, Turner Russell, Yooseph Shibu, Lu Fu, R. Nusskern Deborah, Chris Shue Bixiong, Holly Zheng Xiangqun, Zhong Fei, L. Delcher Arthur, H. Huson Daniel, A. Kravitz Saul, Mouchard Laurent, Reinert Knut, A. Remington Karin, G. Clark Andrew, S. Waterman Michael, E. Eichler Evan, D. Adams Mark, W. Hunkapiller Michael, W. Myers Eugene, and Craig Venter J., "Whole-genome shotgun assembly and comparison of human genome assemblies" PNAS vol. 101, No. 7, pp. 1916–1921, February , 2004 Kuleshov, V.; Banerjee, S.; "Minimal-disturbance topology reconfiguration in all-optical networks," SPIE International Symposium on Voice, Video, and Data Communication, Dallas Texas, 1997 vol. 3230 paper No. 15, 1997.

[Jelinek 82] W. R. Jelinek and C. W. Schmid and "The Structure and Organization of Genetic Material", *Journal of Integrative and Comparative Biology*, 1982.

[Johnson 01] G. C. L. Johnson, L. Esposito, B. J. Barratt, A. N. Smith, J. Heward, G. Di Genova, H. Ueda, H. J. Cordell, I. A. Eaves, F. Dudbridge, R. C. J. Twells, F. Payne, W. Hughes, S. Nutland, H. Stevens, P. Carr, E. Tuomilehto-Wolf, J. Tuomilehto, S. C. L. Gough, D. G. Clayton, and J. A. Todd, "Haplotype tagging for the identification of common disease genes" *Nature Genetics*, vol. 29, October 2001.

[Jones 02] G. Jones, "Alfred Russel Wallace, Robert Owen and the Theory of Natural Selection", *BJHS*, vol. 35, pp. 73-96, 2002.

[Jorde 01] L. B. Jorde, W. S. Watkins, and M.J. Bamshad1 "Population Genomics: A Bridge from Evolutionary History to Genetic Medicine", *Human Molecular Genetics*, vol. 10, no. 20, pp. 2199-2207, 2001.

[Kelly 04] E. D. Kelly, F. Sievers, and R. McManus, "Haplotype frequency estimation error analysis in the presence of missing genotype data" *BMC Bioinformatics*, vol. 5:188, 2004.

[Kimmel 04] G. Kimmel, and R. Sharan, "The Incomplete Perfect Phylogeny Haplotype Problem," In Proceedings of the Second RECOMB Satellite Workshop on Computational Methods for SNPs and Haplotypes, pp. 59-70, 2004.

[Lancia 01] G. Lancia, V. Bafna, S. Istrail, R. Lippert, and R. Schwartz, "SNPs Problems; Complexity and Algorithms" *Proceedings of the 9th Annual European Symposium on Algorithms*, p.182-193, August 28-31, 2001.

[Lam 00] J. C. Lam, K. Roeder, and B. Devlin, "Haplotype Fine Mapping by Evolutionary Trees" *Am. J. Hum. Genet.* Vol. 66, pp. 659–673, 2000.

[Li 03a] J. Li, and T. Jiang, "Efficient Inference of Haplotypes from Genotypes on a Pedigree" *Journal of Bioinformatics and Computational Biology*, Vol. 1, No. 1, pp. 41-69, 2003.

[Li 03b] J. Li and T. Jiang "Efficient Rule-Based Haplotyping Algorithm for Pedigree Data," In Proc. 7th Annual Conference on Research in Computational Molecular Biology (RECOMB03), pp. 197-206, 2003.

[Li 03c] J. Li; and T. Jiang "PedPhase: Haplotype Inference for Pedigree Data," In progress, 2003.

[Li 04] L. M. Li, J. H. Kim, M. S. and Waterman, "Haplotype Reconstruction from SNP Alignment," *Journal of Computational Biology,* Vol. 11, pp. 505-516, 2004.

[Lin 97] S. Lin, and T. P. Speed, "An Algorithm for Haplotype Analysis" *In Proc. of the Annual Conference on Research in Computational Molecular Biology (RECOMB97)*, 1997.

[Lin 02] S. Lin, D. J. Cutler, E. Michael, M. E., and A. Chakravarti, "Haplotype Inference in Random Population Samples" *Am. J. Hum. Genet.* Vol. 71, pp.1129–1137, 2002.

[Lippert 02] R. Lippert, R. Schwartz, G. Lancia, and S. Istrail , "Algorithmic strategies for the single nucleotide polymorphism haplotype assembly problem" *Briefings in Bioinformatics*, vol. 3, no. 1, pp. 23-31(9), March. 2002

[Liu 01] J. S. Liu, C. Sabatti, J. Teng, B. J. B. Keats, and N. Risch, "Bayesian Analysis of Haplotypes for Linkage Disequilibrium Mapping" Genome Research, vol. 11, pp.1716–1724, 2001.

[Mayr 91] E. Mayr, "*One Long Argument: Charles Darwin and the Genesis of Modern Evolutionary Thought*", Harvard Univ. Press, Cambridge Massachusetts, pp. 575 - dc20, 1991.

[Niu 02] T. Niu, S. Qin Zhaohui, Xu Xiping, and S. Liu Jun, "Bayesian Haplotype Inference for Multiple Linked Single-Nucleotide Polymorphisms" Am J Hum Genet, vol. 70, pp. 157-169, 2002.

[Niu 04a] T. Niu, Lu Xin, K. Hosung, S. Zhaohui, and S. J. Jun, "Haplotype Inference and Its Application in Linkage Disequilibrium Mapping" *LNBI 2983*, pp. 48-61, 2004.

[Niu 04b] T. Niu, "Algorithms for Inferring Haplotypes" *Genetic Epidemiology*, vol.27, pp. 334–347, 2004

[Orzack 03] S. Orzack, D. Gusfield, J. Olson, S. Nesbitt, L. Subrahmanyan, and Vincent P. Stanton Jr., "Analysis and Exploration of the Use of Rule-Based Algorithms and Consensus Methods for the Inferral of Haplotypes", *Genetics*, vol. 165, pp. 915–928, October 2003

[Peer 03] I. Pe'er, and J. S. Beckmann, "Resolution of Haplotypes and Haplotype Frequencies from SNP Genotypes of Pooled Samples", *RECOMB 03*, pp. 237-246, 2003

[Qin 02] Z. S. Qin, Niu Tianhua, and S. Liu Jun, "Partition-Ligation–Expectation-Maximization Algorithm for Haplotype Inference with Single-Nucleotide Polymorphisms" Am J Hum Genet, vol. 71, pp. 1242-1247, 2002.

[Rhode 01] K. Rhode, and R. Fuerst "Haplotyping and Estimation of Haplotype Frequencies for Closely Linked Biallelic Multilocus Genetic Phenotypes Including Nuclear Family Information" Human Mutation vol. 17, pp. 289-295, 2001.

[Rizzi 02] R. Rizzi, V. Bafna, S. Istrail, and G. Lancia, "Practical Algorithms and Fixed-Parameter Tractability for the Single Individual SNP Haplotyping Problem", *WABI*, pp. 29-43, 2002

[Ruano 90] G. Ruano, K. K. Kennet, and J. C. Stephens, "Haplotype of multiple polymorphisms resolved by enzymatic amplification of single DNA molecules" *Proc. Natl. Acad. Sci.* vol. 87, pp. 6296-6300, 1990.

[Saiki 85] R. K. Saiki, S. Scharf, F. Faloona, K. B. Mullis, G. T. Horn, H. A. Ehrlich, and N. Arnheim, "Enzymatic Amplication of Beta-globin Genomic Sequences and Restriction Site Analysis for Diagnosis of Sickle Cell Anaemia. *Science* vol. 230, pp. 1350-1354, 1985.

[Schaid 02] D. J. Schaid, S. K. McDonnell, L. Wang, J. M. Cunningham, and S. N. Thibodeau, "Caution on Pedigree Haplotype Inference with Software That Assumes Linkage Equilibrium" Am. J. Hum. Genet. Vol. 71, pp. 992–995, 2002.

[Scharf 86] S. J. Scharf, G. T. Horn, and H. A. Erlich, "Direct Cloning and Sequence Analysis of Enzymatically Amplified Genomic Sequences. Science vol. 233, pp. 1076- 1078, 1986.

[Service 99] S. K. Service, W. Temple Lang D., B. Freimer N., and A. Sandkuijl L., "Linkage-Disequilibrium Mapping of Disease Genes by Reconstruction of Ancestral Haplotypes in Founder Populations" Am. J. Hum. Genet. Vol. 64, pp. 1728–1738, 1999.

[Sobel 96] E. Sobel, and K. Lange, "Descent graphs in pedigree analysis: applications to haplotyping; location scores; and marker-sharing statistics." American J. Hum Genet, vol. 58, No. 6, pp. 1323-37, June, 1996.

[Stephens 00] M. Stephens, and P. Donnelly, " Inference in molecular population genetics" J. R. Stat. Soc. Vol. 64, No.4, pp. 605-655, 2000.

[Stephens 01] M. Stephens, N. J. Smith, and P. Donnelly, "A New Statistical Method for Haplotype Reconstruction from Population Data" Am. J. Hum. Genet. Vol. 68, pp.978–989, 2001.

[Stephens 03] M. Stephens, and P. Donnelly, "A Comparison of Bayesian Methods for Haplotype Reconstruction from Population Genotype Data" Am J Hum Genet. Vol. 73, pp. 1162–1169, 2003.

[Shabarova 94] Z. Shabarova and A. Bogdanov, "Advanced Organic Chemistry of Nucleic Acids", *VCH Verlagsgesellshaft mbH Weinheim*, 1994.

[Shermer 06] M. Shermer, "Astonishing Mind: Francis Crick 1916–2004", *Skeptics Society*, Retrieved on 2006.

[Tapadar 00] P. Tapadar, S. Ghosh, and P. P. Majumder, "Haplotyping in Pedigrees via a Genetic Algorithm" Hum Hered, vol. 50, pp. 43-56, 2000.

[Terhalle 03] W. Terhalle, S. Willi, and K. Karla, "A New Method for Predicting Haplotype Pairs for Genotypes" Currents in Computational Molecular Biology 2003. Berlin: MPI for Molecular Genetics and BCB, pp.95-96, 2003.

[Tost 02] J. Tost, O. Brandt, F. Boussicault, D. Derbala, C. Caloustian, Lechner, and D. I. G. Gut, "Molecular Haplotyping at High Throughput" *Nucleic Acids Research*, vol. 30, No. 19, 2002.

[Wang 03] L. Wang and Y. Xu, "Haplotype Inference by Maximum Parsimony" *Bioinformatics*, vol. 14, 2003.

[Watson 80] J. D. Watson, "The Double Helix: A Personal Account of the Discovery of the Structure of DNA", *W.W. Norton and Co., New York*, 1980.

[Yang 03] Y. Yang, Zhang Jingshan, Hoh Josephine, Matsuda Fumihiko, Xu Peng, Lathrop Mark, and Ott Jurg, "Efficiency of single-nucleotide polymorphism haplotype estimation from pooled DNA" PNAS, vol. 100, No. 12pp. 7225–7230, June, 2003.

VITAE AUCTORIS

NAME:	Qamar Saeed
COUNTRY OF BIRTH:	Pakistan
YEAR OF BIRTH:	1973
EDUCATION:	Bachelor of Computer Science University of Windsor Windsor, ON, Canada, 2002
	B.Sc. Honours [Majors: Biological Sciences and Computer Science] University of Windsor Windsor, ON, Canada, 2004
	Master of Science in Computer Science University of Windsor Windsor, ON, Canada, 2007