Genes Genom (2016) 38:1–12
DOI 10.1007/s13258-015-0342-x

REVIEW

Survey of computational haplotype determination methods for single individual

Je-Keun Rhee[1] · Honglan Li[2] · Je-Gun Joung[3] · Kyu-Baek Hwang[2] · Byoung-Tak Zhang[4] · Soo-Yong Shin[5]

Received: 17 July 2015 / Accepted: 6 October 2015 / Published online: 15 October 2015

Abstract Genome-wide association studies have expanded our understanding of the relationship between the human genome and disease. However, because of current technical limitations, it is still challenging to clearly resolve diploid sequences, that is, two copies for each chromosome. One copy of each chromosome is inherited from each parent and the genomic function is determined by the interplay between the alleles represented as genotypes in the diploid sequences. Thus, to understand the nature of genetic variation in biological processes, including disease, it is necessary to determine the complete genomic sequence of each haplotype. Although there are experimental approaches for haplotype sequencing that physically separate the chromosomes, these methods are expensive and laborious and require special equipment. Here, we review the computational approaches that can be used to determine the haplotype phase. Since 1990, many researchers have tried to reconstruct the haplotype phase using a variety of computational methods, and some researches have been successfully help to determine the haplotype phase. In this review, we investigate how the computational haplotype determination methods have been developed, and we present the remaining problems affecting the determination of the haplotype of single individual using next-generation sequencing methods.

Keywords Haplotype determination · Next-generation sequencing · Computational genomics

✉ Soo-Yong Shin
sooyong.shin@gmail.com; sooyong.shin@amc.seoul.kr

[1] Department of Medical Informatics, College of Medicine, The Catholic University of Korea, Seoul 06591, Korea

[2] School of Computer Science and Engineering, Soongsil University, Seoul 06978, Korea

[3] Translational Bioinformatics Lab., Samsung Genome Institute, Samsung Medical Center, Seoul 06351, Korea

[4] School of Computer Science and Engineering, Seoul National University, Seoul 08826, Korea

[5] Department of Biomedical Informatics, Asan Medical Center, Seoul 05505, Korea

Introduction

The study of DNA sequence variations is one of the main research topics in genetics. Among the diverse variations, single nucleotide polymorphisms (SNPs) frequently occurs in the human genome (Sachidanandam et al. 2001), and their association with disease has been widely investigated. Recently, with the development of high-throughput data generation technologies, it has become possible to carry out genome-wide association studies (GWASs) in the human genome using a huge number of SNPs (Mardis 2008; Feero et al. 2010). In particular, next-generation sequencing (NGS) technologies have helped to identify sequence variations and their characteristics, leading to numerous studies of the associations between SNPs and phenotype, including obesity, diabetes, heart attack and other diseases (Hirschhorn and Daly 2005). Some studies successfully identified associations but, in many cases, GWAS did not have sufficient power and SNP presence did not guarantee a change in the phenotype (Galvan et al. 2010).

In these circumstances, haplotype analysis has been highlighted because it may be more advantageous than traditional genotype analysis in the identification of the presence of genomic sites that affect disease susceptibility (Consortium et al. 2005; Morris and Kaplan 2002). The human genome is diploid and the two copies of each

chromosome have largely identical sequences, except for the X and Y chromosomes. Usually, the human genome is considered to as homozygous because the same alleles are found at each specific locus. However, there are some variations between the pairs in a small portion of the genome, and if there are different alleles at the same positions of the homologous chromosomes, they are referred to as heterozygous alleles. Generally, SNP detection merely reveals if there is a variation at a specific position and it is not determined which of the two chromosome copies contains this variation. Thus, ideally, it needs to list the SNPs belonging in each chromosome copy. Haplotype can be represented as a combination of heterozygous alleles at multiple loci, that is, the sequential combination of the SNPs in a single copy among pairs of chromosomes. The haplotype information can not only identify associations, but can also help in the study of gene function, cis-regulatory roles for gene expression, linkage and inheritance analysis, and evolutionary selection analysis (Bansal et al. 2011; Browning and Browning 2011; Tewhey et al. 2011). Therefore, determination of the haplotype, usually referred as phasing, is important for the full characterization of a single individual genome. Some techniques have been developed to determine haplotypes through molecular experiments, such as the microfluidic whole genome haplotyping approach (Ma et al. 2010; Fan et al. 2011). This method detects the haplotype information by separating the individual chromosomes physically at the cell division process. However, despite its high cost, it is difficult to obtain accurate results (Browning and Browning 2011).

Computational methods are an alternative approach that can reduce the cost of haplotype determination. Here, we review the computational methods that can be used to determine the haplotype phase. Most of the haplotype determination problems can be included in the NP-hard problem, especially when there are some errors (Lancia et al. 2001; Lippert et al. 2002; Cilibrasi et al. 2005). Therefore, many computational approaches have been proposed. There are two computational ways to identify haplotypes: haplotype inference and haplotype assembly. Haplotype inference has been traditionally used in computational genetics areas. It attempts to identify the haplotypes of the samples based on sharing information within the samples from genotype information in the population. Although the same nucleotides are assigned to both chromosomal copies in homozygous alleles, there is only one copy of each nucleotide in heterozygous alleles. If there are m heterozygous sites on the genome, there are 2^m possible haplotypes when we partition it into two groups. This means that direct phasing of the genotype information is computationally expensive. In particular, the haplotype inference methods are much more difficult in the case of samples with rare variants.

Haplotype assembly aims to determine the haplotype of a single individual by directly using sequence reads or fragments originating from one chromosome, meaning that the method assembles sequence fragments or reads from identical chromosomal copies. Previously, DNA microarray analysis has produced genotype information for a set of individuals, so the haplotype has usually been determined by haplotype inference algorithms. However, with the development of high-throughput sequencing technologies, the haplotype assembly methods have become more precise. Generally, sequence reads are mapped to a reference genome, and the origin of the reads cannot be determined. But because each sequence read is derived from a single copy, it is possible to determine the phase information using two or more variants. The initial version of the haplotype assembly method aims to obtain a pair of haplotypes by connecting overlapping fragments with minimum errors. The haplotype assembly methods provide good results in error-free cases, but there are many limitations to its practical use with real datasets. In the current review, we investigate the haplotype determination approaches, explaining the advantages and disadvantages of each method. We also summarize the recent direction and implication of the haplotype determination approaches for the sequencing reads of single individuals.

The structure of this paper is as follows: First, we briefly review the previous haplotype inference methods. The next section defines the single individual haplotyping problem. Then, we review the previous computational haplotype assembly algorithms used at the population level and the recent studies of the single individual haplotyping with various sequencing techniques. The final section discusses the remaining limitations and proposes a future direction.

Traditional haplotype inference algorithms

Before reviewing the haplotype determination methods for single individuals, the traditional haplotype inference algorithms will be introduced. Computational methods for haplotype inference have been studied since 1990 as the microarray technologies have been developed. Because most sequence variation detection methods provide genotype information, the traditional haplotype inference approaches mainly aim to infer the haplotype from the genotypes in the population (Niu 2004; Weale 2004; Salem et al. 2005; Marchini et al. 2006). The haplotype inference methods determine the haplotype under the assumption that genetically closely located loci are linked and that some common haplotypes occupy most of the genetic variations

Table 1 Haplotype inference methods from genotypes in the population

Approach	Related work	Year	References
Rule-based approach	Clark's algorithm	1990	Clark (1990)
Expectation–maximization (EM)	Excoffier and Slatkin	1995	Excoffier and Slatkin (1995)
	HAPLO	1995	Hawley and Kidd (1995)
	Tregouet et al.	2004	Tregouet et al. (2004)
Bayesian method	PHASE	2001	Stephens et al. (2001)
	Haplotyper	2002	Niu et al. (2002)
	Arlequin	2003	Excoffier and Lischer (2010)
	HaploBlock	2004	Greenspan and Geiger (2004)
	DP-Haplotyper	2007	Xing et al. (2007)
Hidden markov model (HMM)	fastPhase	2006	Scheet and Stephens (2006)
	BEAGLE	2007	Browning and Browning (2007)
	IMPUTE2	2009	Howie et al. (2009)
	MACH	2010	Li et al. (2010)
	HapSeq	2012	Zhi et al. (2012)
Tree	HAP	2003	Halperin and Eskin (2004)
	Li et al.	2005	Li et al. (2005)
	PPHS	2012	Efros and Halperin (2012)
Others	PL_EM	2002	Qin et al. (2002)
	hap	2002	Lin et al. (2002)
	hap2	2004	Lin et al. (2004)
	Halldórsson et al.	2011	Halldórsson et al. (2011)

in the population. Table 1 summarizes the previous haplotype inference approaches.

Clark's algorithm was the first computational method for haplotype phasing (Clark 1990). It uses a set of rules to resolve the haplotypes underlying the genotype information. This method starts from a fragment with the clearest information, that is, at one heterozygous site. By starting from the known haplotype, the algorithm searches through the remaining unresolved genotypes and attempts to derive the haplotype from the genotypes by checking if the resolved haplotype can be constructed from a combination of the ambiguous sites of the genotypes. Once the haplotype is inferred from the ambiguous genotypes, it is added to the known haplotype set. The algorithm infers the haplotype by performing the procedures iteratively. The method is easy to use and can be intuitively understood. However, although it performs well with small sets, the performance can be diminished, if the SNPs are not densely connected. Moreover, if there are no initial unambiguous genotypes, the method does not work.

Expectation–Maximization (EM) algorithm is sometimes used to overcome the limitations of Clark's algorithm (Excoffier and Slatkin 1995; Hawley and Kidd 1995). EM-based methods can infer the haplotype phase by assigning the alleles to a haplotype with high probability using the initially estimated frequency values. For the haplotype inference problem, the E-step computes the expected values of the haplotype frequency based on the data and the M-step maximizes the likelihood of the frequency obtained at the E-step. By iterating the E-step and M-step, it can find a possible haplotype for each genotype combination. However, the standard EM-based method does not handle the assumptions regarding genetic recombination and mutations. Also, these can be trapped into local maxima and the results are sensitive to the initial estimation of parameters such as allele frequencies. Moreover, the standard EM-based methods struggle to handle a large number of loci.

Subsequently, many algorithms were developed with various approaches. PHASE (Stephens et al. 2001), Haplotyper (Niu et al. 2002), and HaploBlock (Greenspan and Geiger 2004) use Bayesian method. MACH (Li et al. 2010), fastPhase (Scheet and Stephens 2006), IMPUTE2 (Howie et al. 2009), and BEAGLE (Browning and Browning 2007) were implemented based on the hidden Markov model (HMM). These methods show better performance than the previous parsimony or maximum likelihood approaches. PHASE was the first coalescent theory-based method using the joint probability distribution of haplotypes. Although it is difficult to carry out genome-wide studies using this method because it is only available for a limited number of SNP markers, fastPHASE and BEAGLE made genome-wide studies possible by using a haplotype cluster model, and IMPUTE2 and MACH provide results much faster than PHASE. Also, several methods infer the haplotype using the tree structure (Li

et al. 2005), whereas Arlequin (Excoffier and Lischer 2010) is made using Excoffier-Laval Balding algorithm. In this method, the phase is updated based on a window of neighboring loci and the window size is adaptively determined by the local level of linkage disequilibrium. Arlequin provides good local estimation of the phase in inter-population analysis and searches for the shared haplotypes between populations by comparing them in the inter-population analysis category. In addition, PL-EM was proposed based on a partition-ligation strategy and the EM algorithm (Qin et al. 2002). This method partitions the whole haplotype into small segments and then constructs the partial haplotypes and assembles the segments, using Gibbs sampling or an EM-based approach.

These approaches to statistical inference of the haplotype were largely successful, but the results were sometimes incorrect (Geraci 2010; Browning and Browning 2011). For example, when a haplotype specifically exists in a particular sample or there are rare or novel variants, the haplotype cannot be correctly determined. Moreover, to infer the correct haplotype, a relatively large number of individual genotypes are required. In addition, it is difficult to know if the haplotype is correctly inferred or not.

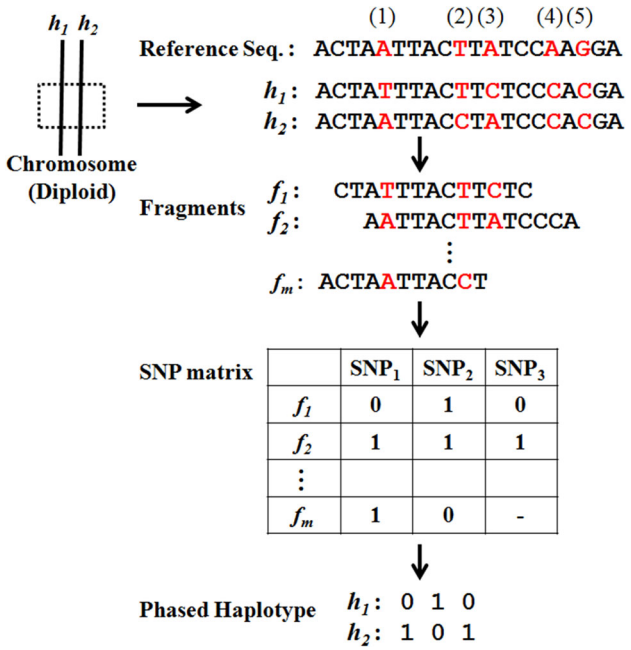

Fig. 1 Haplotype determination problem. SNP sites that are different from the reference sequence are indicated by (1) to (5). The positions (1) to (3) are hetero SNPs and (4) to (5) are homo SNPs. *f* is a fragment and *h* is a haplotype

Definition of the haplotype determination of a single individual

Haplotype determination was defined by Lancia et al. (2001). Figure 1 shows an example of haplotype assembly. The sequence fragments are aligned to the reference genome sequence. The sequence fragment can be a single sequence read or concatenated sequence reads. If a paired-end or mate-pair technique is applied, the fragment can be a combination of two reads. Using this information, we can call sequence variants and determine the homozygous (homo) and heterozygous (hetero) sites in the genome. Homozygous sites have identical alleles, whereas heterozygous sites have different alleles. One genomic site can usually produce three different genotypes, a homo wildtype, a homo mutant type (homo SNP), and a hetero form of the wildtype and mutant type (hetero SNP). It is enough to consider only hetero SNPs for the haplotype determination problem because the homo SNPs and the other identical positions are exactly the same in both chromosomes.

To determine the haplotype, we need to constitute the fragment *f*, which is represented as a set of heterozygous sites in a DNA fragment. The alphabet of the fragment is presented as "1", "0", and "−". If a base in a specific site is identical to the reference, the alphabet is 1. If there is a sequence variant, then the alphabet is 0. If there is no the matched form, the alphabet is presented as '−'. We can then construct $n \times m$ fragment matrix M, where n is the total number of heterozygous sites and m is the number of fragments. From the fragment matrix M, haplotype determination algorithms try to detect a pair of haplotypes. As an example in Fig. 1, we assumed that there are five SNPs in the genome sequence. However, there were hetero SNPs at positions 1, 2, and 3; and homo SNPs at positions 4 and 5. In this case, only three hetero SNPs were used.

Several notations were defined for the following sections: h is haplotype, and f is fragment. So, j-th fragments are represented as f_j and the k-th haplotype is h_k. Then, f_{ji} and h_{ji} are i-th hetero SNP sites at the fragment f_j and haplotype h_j, respectively. The next section will review the previous approaches to haplotype assembly.

Computational approaches for haplotype assembly

After the mid-2000s, many researchers tried to solve the haplotype assembly problem by the taking advantage of developments in sequencing technology (Geraci 2010). The approaches can be categorized by their objective function model or their computational approach. In this paper, the previous methods are mainly categorized using their objective functions: minimum error correction (MEC), weighted minimum letter flip (WMLF), maximum fragment cut (MFC), and others (Table 2).

Table 2 Haplotype assembly methods for single individuals

Model	Approach	Related work	Year	References
MEC[a]	Branch and bound algorithm	Wang et al.	2005	Wang et al. (2005)
		Lim et al.	2012	Lim et al. (2012)
	Genetic algorithm	Wang et al.	2005	Wang et al. (2005)
		GA-MEC	2008	Wu et al. (2008)
		GAHap	2012	Wang et al. (2012)
	Satisfiability problem (SAT)	He et al.	2010	He et al. (2010)
		HapSat	2011	Mousavi et al. (2011)
		xGenHapSat	2012	Mousavi (2012)
	Probabilistic approach	SHR	2008	Chen et al. (2008)
		HASH	2008	Bansal et al. (2008)
		HAPCUT	2008	Bansal and Bafna (2008)
		ProbHap	2014	Kuleshov (2014)
	Others	2D-MEC	2007	Wang et al. (2007)
		Wu et al.	2009	Wu et al. (2009)
		SSK	2012	Xu and Li (2012)
		Deng et al.	2013	Deng et al. (2013)
		HapAssembly	2013	Chen et al. (2013)
WMLF[b]		Zhao et al.	2005	Zhao et al. (2005)
		Kang et al.	2008	Kang et al. (2008)
		Xie et al.	2008	Xie et al. (2008)
		HapAssembler	2010	Kang et al. (2010)
		Wu et al.	2013	Wu et al. (2013)
MFC[c]	Graph	RefHap	2010	Duitama et al. (2010)
Others	Heuristic algorithm	Levy et al.	2007	Levy et al. (2007)
		SpeedHap	2007	Genovese et al. (2008)
	Graph (spanning tree)	HapCompass	2012	Aguiar and Istrail (2012)
	Heuristic dynamic programming	H-BOP	2012	Xie et al. (2012)
	Mixture model	MixSIH	2013	Matsumoto and Kiryu (2013)
	Fuzzy conflict graphs	FastHap	2014	Mazrouee and Wang (2014)

[a] *MEC* minimum error correction

[b] *MLF* weighted minimum letter flip

[c] *MFC* maximum fragment cut

Minimum error correction

MEC aims to minimize errors by comparing the predicted haplotype and the input SNP matrix M. Figure 2 illustrates an example of haplotype construction using a partition to calculate the errors. The input SNP matrix M is partitioned and two haplotypes are constructed from the fragments in each partition. The objective is to minimize the sum of the differences between the constructed haplotype and the partitioned matrix.

Suppose that there are m fragments at M. The MEC score is represented as follows:

$$MEC(M,H) = \sum_{j=1}^{m} \sum_{k=1}^{2} D_{perFrag}(f_j, h_k) \quad (1)$$

The score is presented as the sum of the differences between the j-th fragment f_j and the k-th haplotype h_k. The differences $D_{perFrag}(f_j, h_k)$ is calculated by Eq. (2). $D_{perSNP}(f_{ji}, h_{ki})$ is 1 if the fragment f_j and haplotype h_k are different at the i-th identical SNP site; otherwise, it is 0. $G(h_k)$ is the k-th subset of the fragments when the fragments are partitioned to k subsets. That is, $F(f_j, h_k)$ is an indicator function representing which partition includes the fragment f_j.

$$D_{perFrag}(f_j, h_k) = \sum_{i=0}^{n} F(f_j, h_k) \times D_{perSNP}(f_{ji}, h_{ki}) \quad (2)$$

$$F(f_j, h_k) = \begin{cases} 1 & \text{if } f_j \in G(h_k) \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

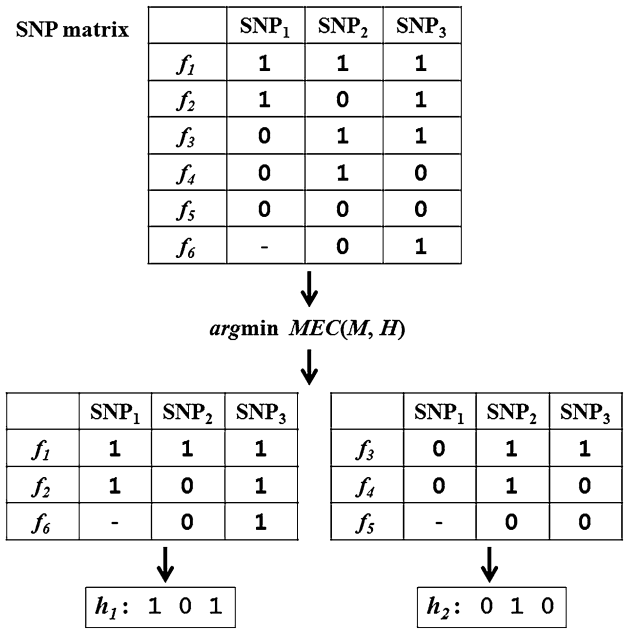

Fig. 2 An example of haplotype determination using the MEC model. M is a SNP fragment matrix, and H indicates the predicted haplotypes

$$D_{perSNP}(f_{ji}, h_{ki}) = \begin{cases} 1 \text{ if } f_{ji} \neq '-', h_{ki} \neq '-', \text{ and } f_{ji} \neq h_{ki} \\ 0 \text{ otherwise} \end{cases} \quad (4)$$

As an example in Fig. 2, three fragments, f_1, f_2, and f_6 are included in h_1 and other fragments in h_2, then $F(f_1, h_1)$ is 1 and $F(f_1, h_2)$ is 0. MEC model is the most popular way to solve the haplotype determination problem and the goal can be easily understood. Therefore, diverse computational approaches have been applied based on MEC. The following subsections will explain the representative computational methods that use MEC as the objective function.

Based on the definition in the previous section, haplotype determination can be viewed as a way to find the optimal path using a binary tree, because the problem can be converted into choosing the side between haplotypes $h1$ and $h2$. Wang et al. (2005) tried to apply a so-called branch and bound algorithm. Each node is a fragment in the tree structure and the edge indicates the index of the haplotype group. From the root node, that is the first fragment, the algorithm adds a fragment and measures the MEC score. Then if the calculated score is bigger than the previous score, it would be divided. The branch and bound algorithm can identify the exact optimal solution, but the time complexity is exponentially increased by the number of fragments. Therefore, its use in a large-scale datasets is difficult. Lim et al. (2012) reduced the search space of the branch and bound algorithm by identifying the initial upper bound using a local search algorithm and solved the MEC problem in practical terms. However, their method was applied to insect species, so further improvement would be necessary to allow it to be used in other species.

Genetic algorithms have been applied due to its reliable performance in NP problems that require a great deal of computational time. Wang et al. (2005) evaluated the haplotype in each generation using a fitness function based on MEC model. The genetic algorithm finds a solution using the following fitness value in each generation:

$$fitness = 1 - \frac{MEC(M, H)}{m \times n} \quad (5)$$

Similar to the conventional genetic algorithms, the initial population is randomly generated. In the evolutionary process, the population is re-generated based on the previous individuals, using crossover and mutation operators. The haplotypes in each generation are evaluated using Eq. (5), and then the new individuals are generated. By repeating the process, the optimal haplotype can be selected. GA-MEC is a variation of the genetic algorithm for the determination of haplotypes (Wu et al. 2008). In GA-MEC, the fragments are partitioned at each generation using a method similar to k-means clustering. After a pair of haplotypes is constructed randomly, each fragment is compared to the two haplotypes. Then, the fragment is classified to the group that has the minimal $D_{perFrag}(\cdot)$ value in Eq. (2). In this way, each fragment is assigned to the two haplotype groups. Next, new haplotypes are constructed using the divided fragment information and the fragment is re-classified by measuring the $D_{perFrag}(\cdot)$ value between the newly generated haplotype and the fragment. By repeating the iterative process until the haplotype does not change, the haplotype is determined. The fitness value is also calculated by Eq. (5). GAHap is another similar approach (Wang et al. 2012), but it uses Hamming distance to calculate the difference between a haplotype and a fragment, without encoding the SNP values to 0, 1, and '−', as introduced in the previous section. Therefore, the method has advantages in that it can handle data that include tri- or tetra-allelic loci. Also, it does not remove homozygous sites from the input matrix, so it reconstructs the haplotype considering the case marked as a homozygous locus by a sequence error in the original data. However, the GA-based approaches usually require considerable time to identify the solution, so their application to large datasets is difficult.

Another way to solve the haplotype determination using a MEC model is to transform the problem into a satisfiability (SAT) problem. He et al. (2010) proposed a partial Max-SAT formulation for haplotype assembly. Mousavi et al. (2011) suggested a HapSat model by converting the haplotype determination problem to a Max-2-SAT problem, which is more general than the partial Max-SAT. This

approach can use the general Max-SAT solver and the formation is more generalized, considering homozygous alleles that can appear due to sequence errors. In addition, it is formulated with fewer variables and clauses. The logical equations are as follows:

$$
\begin{aligned}
&C = \{\} \\
&F'_j = \begin{cases} 0 & \text{if } F(f_j, h_1) = 1 \\ 1 & \text{if } F(f_j, h_2) = 1 \end{cases} \\
&\text{if } f_{ji} = 0 \\
&\quad C = C \cup \{(F'_j v \sim h_{1i}), (\sim F'_j v \sim h_{2i})\} \\
&\text{else if } f_{ji} = 1 \\
&\quad C = C \cup \{(F'_j v h_{1i}), (\sim F'_j v h_{2i})\}
\end{aligned} \tag{6}
$$

Often, the datasets for haplotype determination such as read mapping data and variant calling data are incomplete and include several errors. Some studies tried to overcome this limitation by using probabilistic models. Because the fragments of the input SNP matrix are obtained from two haplotypes, Chen et al. (2008) assumed that the fragments were generated according to two parameters representing errors. They designed a probabilistic function using the error parameters for the haplotype. From the input fragment matrix M, the fragments are divided into two sets and the most frequent character in each SNP site is selected to determine the haplotype sequence. Therefore, the two haplotypes can be reconstructed with a possible high probability. HASH (Bansal et al. 2008) and HAPCUT (Bansal and Bafna 2008) also used probabilistic models based on graph structure. They constructed a graph with the input matrix. The node is the column of the matrix, which is itself each SNP. If there is a fragment that includes the two sites, the two nodes are connected by an edge. The weight of the edge is the difference between the number of fragments matched to the haplotype sites and the unmatched fragment. HASH uses a graph-cut algorithm and constructs Markov chain, but HAPCUT optimizes the MEC score by using a Max-Cut algorithm. In these two methods, the distance to both haplotypes is calculated for all fragments, and the fragments with the lowest distance are selected. By calculating the overall MEC scores and using a greedy algorithm, the best pair of haplotypes is determined by using the best MEC score.

Clustering approaches have also been used. Wang et al. (2007) suggested 2D-MEC model in which the difference between two fragments is measured as shown in Eq. (7). This method iteratively divides the fragments and generates haplotypes.

$$
D_{perFragToFrag}(f_j, h_k) = \sum_{i=0}^{n} D_{perSNP}(f_{ji}, f_{ki}) \tag{7}
$$

Wu et al. (2009) clustered the m fragments to two groups representing haplotypes by self-organizing map (SOM) and Xu and Li (2012) used a semi-supervised k-means clustering method.

In addition, one method uses dynamic programming to determine the haplotype for a single individual (He et al. 2010). Basically short reads are represented as binary strings and assigned to each pair of haplotypes to minimize the conflicts with the reads. However, this approach uses the optimal MEC for partial haplotypes and repeatedly extends the partial haplotypes by one bit to obtain the full-length haplotypes. The results showed that the method can be applied to whole-genome sequencing datasets. However, when there are many SNP sites, the dynamic programming algorithm needs considerable computational time. To solve this drawback, Deng et al. (2013) combined this dynamic programming method with a heuristic approach. This method first obtains a subset of the input matrix M by a randomized sampling approach and carries out the dynamic programming. Once it produces an initial solution from the submatrix, it refines the haplotypes by comparing the initial haplotype with all fragments. By repeating the initial solution and refinement steps, the haplotype is determined.

HapAssembly converts the haplotype assembly problem to an integer linear programming problem for optimization (Chen et al. 2013). The input matrix M is decomposed into small independent blocks and the integer linear programming problem is formulated for each block. It can obtain good results for single optimal solution problems.

Weighted minimum letter flip

The WMLF is a modification of the MEC model. Basically, when the constructed haplotype is different from elements of the input SNP matrix, a letter flip is performed. The total error is measured by the number of letter flips so that the SNP matrix is identical to the constructed haplotype. The WMLF model additionally uses the m-by-n weight matrix representing the confidence of each SNP. If all of the elements on the weight matrix are 1, then the WMLF is identical to the MEC model. The difference between the fragments and haplotype in WMLF is shown in Eq. (9).

$$
D_{WMLF\_SNP}(f_{ji}, h_{ki}) = \begin{cases} W_{ji} \text{ if } f_{ji} \neq '-', \text{ and } f_{ji} \neq h_{ki} \\ 0 \text{ otherwise} \end{cases} \tag{8}
$$

$$
WMLF(M, H, W) = \sum_{i=1}^{m} D_{WMLF\_SNP}(f_{ji}, h_{ki}) \tag{9}
$$

Equation (9) calculates the total distance which is the sum of the differences between all of the fragments and the haplotype.

The WMLF model has been used as an objective function for genetic algorithms (Kang et al. 2010) and heuristic

approaches (Xie et al. 2012). Moreover, there is also a complete WMLF (CWMLF) model, which combines the WMLF, minimum fragment removal (MFR), and minimum SNP removal (MSR) (Zhao et al. 2005). MFR is a method to remove the minimum number of vertices from the fragment conflict graph so that the resulting graph is bipartite, whereas MSR is a method to remove the minimum number of vertices from the SNP conflict graph so that no two vertices are adjacent. Zhao et al. (2005) showed that the CWMLF model is effective for solving the haplotype assembly problem by showing that the SNP errors and fragment error rates are lower than those of the WMLF model in their experimental datasets.

Maximum fragment cut

The MFC converts the haplotype determination problem to a Max-Cut problem. The vertices in the graph structure are the fragments *f*s and the edges are represented by the similarity between two fragments. RefHap is the most popular method that uses MFC (Duitama et al. 2010) and it is one of the practically applicable methods at present. The distance between two the fragments f_j and f_l is defined as follows:

$$D_{MFC\_Frag}(f_{ji},f_{li}) = \begin{cases} 1, & \text{if } f_{ji} \neq '-', f_{li} \neq '-', \text{ and } f_{ji} \neq f_{li} \\ -1, & \text{if } f_{ji} \neq '-', f_{li} \neq '-', \text{ and } f_{ji} = f_{li} \\ 0, & \text{otherwise} \end{cases} \quad (10)$$

The distance between two fragments is defined as the sum of the SNP distance, D_{MFC_Frag}. Note that Eq. (10) has -1 even if these have the same value. In this approach, the fragments are divided into two groups to minimize the sum of the distances. For example, RefHap uses a graph structure and each node represents a fragment in the graph. When the fragments have different characters, the two nodes are connected. The weights for the edges are determined by how many characters are not identical between the two fragments. This method aims to divide the nodes into two subsets with the smallest sum of distances.

Other objective functions

Several researchers have presented another possible ways to determine single individual haplotypes. Levy et al. (2007) and Genovese et al. (2008) proposed heuristic methods for the haplotype determination using greedy methods to construct the haplotype. Aguiar and Istrail (2012) developed a haplotype determination method that uses graph structure. In the model, the SNPs are nodes and the sequence reads are edges. Using this method, the SNPs and sequence reads are converted to spanning trees and then the haplotype is assembled by solving the minimum weighted edge removal optimization problem. Matsumoto and Kiryu (2013) developed a variational Bayes expectation maximization (VBEM) algorithm that has two mixture components representing each haplotype. The authors defined the minimum connectivity score (MC score), which is a quality score evaluating partially assembled haplotype segments that are free from switch errors. By selecting regions with high MC scores, the haplotypes can be accurately assembled. Another approach is to combine the previously defined objective functions. Xie et al (2012) combined the two existing models, MEC and MFC, showing that the model could efficiently solve the haplotype phasing problem. Mazrouee and Wang (2014) constructed a fuzzy conflict graph by defining the inter-fragment distance. Fragment partition was performed using the fuzzy conflict graph, similar to the previous MFC method. The partition is then further refined with the MEC model to achieve improved results.

Recent studies of single individual haplotyping using NGS

Nowadays, with developments in sequencing technologies, the aim of haplotype determination is to determine the haplotype of the entire genome of a single individual. Clark's algorithm, which is the first computational haplotype phasing method developed, was applied to only two genes, Adh and Est-6 of *Drosophila melanogaster* (Clark 1990). These genes contain 43 and 52 SNPs, respectively. Even though some algorithms were developed for single individual haplotyping, they were applied to a small number of SNPs, around 100, by limiting the dataset (Wang et al. 2005, 2007; Zhao et al. 2005; Kang et al. 2010). Successful results were obtained, but it is difficult to confirm that these algorithms would work well in large-scale datasets, such as the entire human genome.

To date, some complete real datasets have been generated for haplotype determination. Table 3 summarizes the sequencing-based datasets produced for haplotype determination with computational approaches. Levy et al. (2007) first constructed the haplotype of a single individual from the sequencing datasets. They collected the sequence fragments of Craig J. Venter (HuRef) that were generated by the Sanger sequencing method and assembled the fragments by using a greedy algorithm. The HuRef dataset has been used to verify the performance of newly developed computational algorithms for some years. For example, the HASH and HAPCUT methods were originally applied to the HuRef datasets (Bansal et al. 2008; Bansal and Bafna 2008) to validate their proposed methods. He et al. (2010) also solved the haplotype

Table 3 Haplotype determination for human individuals using high-throughput sequencing datasets and computational approaches

Dataset	Sequencing method	Computational approaches	Year	Representative references
HuRef	Sanger sequencing	**Greedy algorithm** Levy et al. (2007) HASH Bansal et al. (2008) HAPCUT Bansal and Bafna (2008) He et al. (2010) HapAssembly Chen et al. (2013)	2007	Levy et al. (2007)
MaxPlank one	Fosmid sequencing	**RefHap** Duitama et al. (2010)	2011	Suk et al. (2011)
HapMap sample NA20847	Fosmid sequencing	**HAPCUT** Bansal and Bafna (2008)	2011	Kitzman et al. (2011)
HapMap sample NA12878	Fosmid sequencing	**RefHap** Duitama et al. (2010) HapAssembly Chen et al. (2013) MixSIH Matsumoto and Kiryu (2013) H-BOP Xie et al. (2012)	2012	Duitama et al. (2011)
HapMap sample NA19240, six libraries from European HapMap pedigree 1463, and a single library from personal genome project sample NA20431	Dilution-based sequencing with barcode adapters	**Graph-based custom LRF haplotype algorithm** Peters et al. (2012)	2012	Peters et al. (2012)
HapMap sample NA18506, NA20847, NA18507, HG01377, NA18506, and NA12878	Dilution-based sequencing with barcode adapters	**RefHAP** Duitama et al. (2010)	2013	Kaper et al. (2013)
NA20431 of the personal genome project (designated PGP1)	BAC	**HAPCUT** Bansal and Bafna (2008)	2013	Lo et al. (2013)
HapMap sample NA12878, NA12891, NA12892	Dilution-based sequencing with barcode adapters	**Prism** Kuleshov et al. (2014)	2014	Kuleshov et al. (2014)

Bold indicates a method that is used in the original paper presented in the last column

determination problem using the HuRef datasets and dynamic programming for short sequence reads by converting to Max-SAT problem for paired-end reads.

For single individual haplotyping, one of the biggest problems with NGS data is the short length of the sequence reads because these short reads do not include enough variations. However, several experimental methods have recently been developed to overcome the short length problem. One of the recent representative methods is the fosmid pool-based sequencing approaches. Kitzman et al. (2011) and Suk et al. (2011) produced fosmid-based sequencing datasets for a single individual and determined the haplotype by using the previously developed HAPCUT and RefHap methods, respectively. Duitama et al. (2011) generated fosmid-based sequencing datasets for the NA12878 CEU individual. In recent years, the sequencing data for the HapMap sample NA12878 has often been used for computation phasing experiments and for the evaluation of haplotype phasing results.

Although fosmid-based sequencing can generate long-phased contigs, a large amount of DNA for sequencing and extensive library processing are needed. To overcome this problem, Peters et al. (2012) developed long fragment read (LFR) technology without cloning or physical separation of chromosomes to determine the haplotype. The computational haplotyping methods are based on a graph structure. They constructed a graph with nodes corresponding to the hetero SNPs and with edges corresponding to the expected distances within the hetero SNP pairs. The haplotype was assembled by generating a minimum spanning tree from the datasets. Kaper et al. (2013) also proposed a cost effective method using conceptually similar dilution-amplification-based sequencing techniques. These methods reduce the library preparation time but require ultra-deep sequencing of the samples and partially unphased variants can remain. Kuleshov et al. (2014) proposed a statistically aided, long-read haplotyping (SLRH) method. They constructed haplotypes from relatively short DNA fragments that are amplified by PCR to reduce the amplification bias. The method involves two stages. In the first local assembly step, the fragments are assembled into haplotype blocks by connecting the fragments with overlapping hetero SNPs using a dynamic programming algorithm. The local assembly step is conceptually similar to the previous haplotyping methods, such as RefHap (Duitama et al. 2010) and HapCut (Bansal and Bafna 2008). Then, in the global

assembly step the local blocks are formed into long haplotype contigs based on the hidden Markov model, similar to IMPUTE2 (Howie et al. 2009). This approach is able to produce long haplotype contigs with a high likelihood score and high confidence. In addition, Lo et al. (2013) generated a sequencing results based on bacterial artificial chromosomes (BACs). Their approach could reduce the sequencing costs and generate longer fragments than fosmid-based sequencing and LFR techniques.

Conclusions and future directions

The haplotype information of the entire genomes of a single individual helps to clarify our understanding of the structure of the human genome and its individual-specific function. Moreover, it will provide more accurate translational results and phenotype prediction (Tewhey et al. 2011; Browning and Browning 2011; Hoehe 2003; Glusman et al. 2014). Because it is still expensive to separate the two copies of a chromosome using wet-lab experimental methods, computational methods will play a practically important role in the single individual haplotyping of the whole genome. The present paper reviewed the haplotype determination problem and its computational solution. The methods have been rapidly developed in response to the production of a growing number of datasets by NGS techniques. Many researchers have tried to solve the haplotype determination problem for single individuals using various computational approaches, including statistical and heuristic methods, with several proper objective functions.

These approaches achieved successful results, but it is still a challenge to determine the complete haplotype of a single individual on a genome-scale. One of the main problems to be tackled is the short length of the sequencing reads. Short reads do not include enough sequence variations in a single read, so it is difficult to determine the complete haplotype. Recently, longer reads have been made possible by developments in sequencing methods, for example, fosmid-based sequencing methods (Kitzman et al. 2011; Suk et al. 2011; Duitama et al. 2011). Furthermore, nanopore DNA sequencing methods would also be helpful (Clarke et al. 2009). However, these methods are labor intensive and time-consuming, so a more effective algorithm is necessary to determine haplotype from high-throughput sequence reads. Kuleshov et al. (2014) proposed a statistically aided, long-read haplotyping method to determine single individual haplotypes by combining relatively short reads and fosmid sequencing. However it is still challenging to determine the haplotype from short sequence reads. Moreover, in most cases, it is impossible to fully reconstruct the haplotype, usually because of the coverage problem. Another factor affecting the accuracy in real problems is the error rate. Most recent haplotype assembly algorithms provide effective results if the datasets have low error rates. However, in datasets with high error rates, the accuracy is decreased. To overcome the read length and error rate problems, a new algorithm should be developed to uncover the relationships among multiple fragments, even though the fragments do not overlap. Recently, Chen et al. (2014) proposed a hypergraph-based haplotype assembly algorithm to capture the higher-order relationships among the fragments. However, it needs to be validated using recent real sequence datasets. Moreover, use of family or trio information would help to improve the accuracy of the haplotype determination results (Aguiar and Istrail 2013; Roach et al. 2011). Additionally, Yang et al. (2013) tried to overcome the haplotype assembly problem by identifying most likely haplotype segments based on a probabilistic model, by combining the traditional haplotype inference methods. They combined the information from a reference dataset such as HapMap or the 1000 genome by using a likelihood framework.

Finally, to improve efficiency and achieve a reasonable computational time and memory capacity, parallel computing approaches or the application of MapReduce methods are required. Due to the computational challenges, parallel computing has been increasingly important in the biological research area, especially in high-performance sequencing (Taylor 2010). As mentioned above, haplotype determination is a time-consuming process that requires extensive computational power and memory capacity. High performance computing would overcome this limitation. The MapReduce frameworks have been used for NGS analysis. For example, the Genome analysis toolkit, one of the most popular NGS analysis tools, was designed using the MapReduce framework (McKenna et al. 2010). In addition, Cloudburst, a tool to map the sequence reads to a reference genome, also uses a parallel read-mapping algorithm based on MapReduce (Schatz 2009). Therefore, a new algorithm needs to be developed for haplotype determination based on parallel computing. This new algorithm would help more accurate and effective haplotypes to be constructed and help to improve the biological understanding of the human genome.

Acknowledgments This work was supported by the Basic Science Research Program (2012R1A1A2002804) though National Research Foundation (NRF) grant funded by the Korean government (MISP).

Compliance with ethical standards

Conflict of Interest The authors declare no competing interests in relation to this study.

References

Aguiar D, Istrail S (2012) Hapcompass: a fast cycle basis algorithm for accurate haplotype assembly of sequence data. J Comput Biol 19:577–590

Aguiar D, Istrail S (2013) Haplotype assembly in polyploid genomes and identical by descent shared tracts. Bioinformatics 29:i352–i360

Bansal V, Bafna V (2008) Hapcut: an efficient and accurate algorithm for the haplotype assembly problem. Bioinformatics 24:i153–i159

Bansal V, Halpern AL, Axelrod N, Bafna V (2008) An MCMC algorithm for haplotype assembly from whole-genome sequence data. Genome Res 18:1336–1346

Bansal V, Tewhey R, Topol EJ, Schork NJ (2011) The next phase in human genetics. Nat Biotechnol 29:38–39

Browning SR, Browning BL (2007) Rapid and accurate haplotype phasing and missing-data inference for whole-genome association studies by use of localized haplotype clustering. Am J Hum Genet 81:1084–1097

Browning SR, Browning BL (2011) Haplotype phasing: existing methods and new developments. Nat Rev Genet 12:703–714

Chen X, Peng Q, Han L, Zhong T, Xu T (2014) An effective haplotype assembly algorithm based on hypergraph partitioning. J Theor Biol 358:85–92

Chen Z, Fu B, Schweller R, Yang B, Zhao Z, Zhu B (2008) Linear time probabilistic algorithms for the singular haplotype reconstruction problem from snp fragments. J Comput Biol 15:535–546

Chen ZZ, Deng F, Wang L (2013) Exact algorithms for haplotype assembly from whole-genome sequence data. Bioinformatics 29:1938–1945

Cilibrasi R, Van Iersel L, Kelk S, Tromp J (2005) On the complexity of several haplotyping problems. In: Casadio R, Myers G (eds) Algorithms in bioinformatics. Springer, Heidelberg, pp 128–139

Clark AG (1990) Inference of haplotypes from PCR-amplified samples of diploid populations. Mol Biol Evol 7:111–122

Clarke J, Wu HC, Jayasinghe L, Patel A, Reid S, Bayley H (2009) Continuous base identification for single-molecule nanopore DNA sequencing. Nat Nanotechnol 4:265–270

Consortium IH et al (2005) A haplotype map of the human genome. Nature 437:1299–1320

Deng F, Cui W, Wang L (2013) A highly accurate heuristic algorithm for the haplotype assembly problem. BMC Genom 14(Suppl 2):S2

Duitama J, Huebsch T, McEwen G, Suk EK, Hoehe MR (2010) Refhap: a reliable and fast algorithm for single individual haplotyping. In: Proceedings of the First ACM International Conference on Bioinformatics and Computational Biology, ACM, pp 160–169

Duitama J, McEwen GK, Huebsch T, Palczewski S, Schulz S, Verstrepen K, Suk EK, Hoehe MR (2011) Fosmid-based whole genome haplotyping of a hapmap trio child: evaluation of single individual haplotyping techniques. Nucleic Acids Res 40:2041–2053

Efros A, Halperin E (2012) Haplotype reconstruction using perfect phylogeny and sequence data. BMC Bioinformatics 13(Suppl 6):S3

Excoffier L, Lischer HE (2010) Arlequin suite ver 3.5: a new series of programs to perform population genetics analyses under linux and windows. Mol Ecol Resour 10:564–567

Excoffier L, Slatkin M (1995) Maximum-likelihood estimation of molecular haplotype frequencies in a diploid population. Mol Biol Evol 12:921–927

Fan HC, Wang J, Potanina A, Quake SR (2011) Whole-genome molecular haplotyping of single cells. Nat Biotechnol 29:51–57

Feero WG, Guttmacher AE, Manolio TA (2010) Genomewide association studies and assessment of the risk of disease. New Engl J Med 363:166–176

Galvan A, Ioannidis JP, Dragani TA (2010) Beyond genome-wide association studies: genetic heterogeneity and individual predisposition to cancer. Trends Genet 26:132–141

Genovese LM, Geraci F, Pellegrini M (2008) Speedhap: an accurate heuristic for the single individual snp haplotyping problem with many gaps, high reading error rate and low coverage. IEEE/ACM Trans Comput Biol Bioinform(TCBB) 5:492–502

Geraci F (2010) A comparison of several algorithms for the single individual SNP haplotyping reconstruction problem. Bioinformatics 26:2217–2225

Glusman G, Cox HC, Roach JC (2014) Whole-genome haplotyping approaches and genomic medicine. Genome Med 6:73

Greenspan G, Geiger D (2004) Model-based inference of haplotype block variation. J Comput Biol 11:493–504

Halldórsson BV, Aguiar D, Istrail S (2011) Haplotype phasing by multi-assembly of shared haplotypes: phase-dependent interactions between rare variants. In: Pacific Symposium on Biocomputing, World Scientific, pp 88–99

Halperin E, Eskin E (2004) Haplotype reconstruction from genotype data using imperfect phylogeny. Bioinformatics 20:1842–1849

Hawley M, Kidd K (1995) Haplo: a program using the EM algorithm to estimate the frequencies of multi-site haplotypes. J Hered 86:409–411

He D, Choi A, Pipatsrisawat K, Darwiche A, Eskin E (2010) Optimal algorithms for haplotype assembly from whole-genome sequence data. Bioinformatics 26:i183–i190

Hirschhorn JN, Daly MJ (2005) Genome-wide association studies for common diseases and complex traits. Nat Rev Genet 6:95–108

Hoehe M (2003) Haplotypes and the systematic analysis of genetic variation in genes and genomes. Pharmacogenomics 4:547–570

Howie BN, Donnelly P, Marchini J (2009) A flexible and accurate genotype imputation method for the next generation of genome-wide association studies. PLoS Genet 5:e1000

Kang SH, Jeong IS, Choi MH, Lim HS (2008) Haplotype assembly from weighted SNP fragments and related genotype information. In: Chen J, Hopcroft JE (eds) Frontiers in Algorithmics. Springer, Berlin, pp 45–54

Kang SH, Jeong IS, Cho HG, Lim HS (2010) Hapassembler: a web server for haplotype assembly from SNP fragments using genetic algorithm. Biochem Biophys Res Commun 397:340–344

Kaper F, Swamy S, Klotzle B, Munchel S, Cottrell J, Bibikova M, Chuang HY, Kruglyak S, Ronaghi M, Eberle MA et al (2013) Whole-genome haplotyping by dilution, amplification, and sequencing. Proc Natl Acad Sci USA 110:5552–5557

Kitzman JO, MacKenzie AP, Adey A, Hiatt JB, Patwardhan RP, Sudmant PH, Ng SB, Alkan C, Qiu R, Eichler EE et al (2011) Haplotype-resolved genome sequencing of a gujarati indian individual. Nat Biotechnol 29:59–63

Kuleshov V (2014) Probabilistic single-individual haplotyping. Bioinformatics 30:i379–i385

Kuleshov V, Xie D, Chen R, Pushkarev D, Ma Z, Blauwkamp T, Kertesz M, Snyder M (2014) Whole-genome haplotyping using long reads and statistical methods. Nat Biotechnol 32:261–266

Lancia G, Bafna V, Istrail S, Lippert R, Schwartz R (2001) SNPs problems, complexity, and algorithms. In: Meyer auf der Heide (ed.). Algorithms-ESA 2001, Springer, Heidelberg, pp 182–193

Levy S, Sutton G, Ng PC, Feuk L, Halpern AL, Walenz BP, Axelrod N, Huang J, Kirkness EF, Denisov G et al (2007) The diploid genome sequence of an individual human. PLoS Biol 5:e254

Li Y, Willer CJ, Ding J, Scheet P, Abecasis GR (2010) Mach: using sequence and genotype data to estimate haplotypes and unobserved genotypes. Genet Epidemiol 34:816–834

Li Z, Zhou W, Zhang XS, Chen L (2005) A parsimonious tree-grow method for haplotype inference. Bioinformatics 21:3475–3481
Lim HS, Jeong IS, Kang SH (2012) Individual haplotype assembly of apis mellifera (honeybee) using a practical branch and bound algorithm. J Asia-Pac Entomol 15:375–381
Lin S, Cutler DJ, Zwick ME, Chakravarti A (2002) Haplotype inference in random population samples. Am J Hum Genet 71:1129–1137
Lin S, Chakravarti A, Cutler DJ (2004) Haplotype and missing data inference in nuclear families. Genome Res 14:1624–1632
Lippert R, Schwartz R, Lancia G, Istrail S (2002) Algorithmic strategies for the single nucleotide polymorphism haplotype assembly problem. Brief Bioinform 3:23–31
Lo C, Liu R, Lee J, Robasky K, Byrne S, Lucchesi C, Aach J, Church G, Bafna V, Zhang K (2013) On the design of clone-based haplotyping. Genome Biol 14:R100
Ma L, Xiao Y, Huang H, Wang Q, Rao W, Feng Y, Zhang K, Song Q (2010) Direct determination of molecular haplotypes by chromosome microdissection. Nat Methods 7:299
Marchini J, Cutler D, Patterson N, Stephens M, Eskin E, Halperin E, Lin S, Qin ZS, Munro HM, Abecasis GR et al (2006) A comparison of phasing algorithms for trios and unrelated individuals. Am J Hum Genet 78:437–450
Mardis ER (2008) The impact of next-generation sequencing technology on genetics. Trends Genet 24:133–141
Matsumoto H, Kiryu H (2013) Mixsih: a mixture model for single individual haplotyping. BMC Genom 14(Suppl 2):S5
Mazrouee S, Wang W (2014) Fasthap: fast and accurate single individual haplotype reconstruction using fuzzy conflict graphs. Bioinformatics 30:i371–i378
McKenna A, Hanna M, Banks E, Sivachenko A, Cibulskis K, Kernytsky A, Garimella K, Altshuler D, Gabriel S, Daly M et al (2010) The genome analysis toolkit: a mapreduce framework for analyzing next-generation DNA sequencing data. Genome Res 20:1297–1303
Morris RW, Kaplan NL (2002) On the advantage of haplotype analysis in the presence of multiple disease susceptibility alleles. Genet Epidemiol 23:221–233
Mousavi SR (2012) Improved haplotype assembly using xor genotypes. J Theor Biol 298:122–130
Mousavi SR, Mirabolghasemi M, Bargesteh N, Talebi M (2011) Effective haplotype assembly via maximum Boolean satisfiability. Biochem Biophys Res Commun 404:593–598
Niu T (2004) Algorithms for inferring haplotypes. Genet Epidemiol 27:334–347
Niu T, Qin ZS, Xu X, Liu JS (2002) Bayesian haplotype inference for multiple linked single-nucleotide polymorphisms. Am J Hum Genet 70:157–169
Peters BA, Kermani BG, Sparks AB, Alferov O, Hong P, Alexeev A, Jiang Y, Dahl F, Tang YT, Haas J et al (2012) Accurate whole-genome sequencing and haplotyping from 10 to 20 human cells. Nature 487:190–195
Qin ZS, Niu T, Liu JS (2002) Partition-ligation-expectation-maximization algorithm for haplotype inference with single-nucleotide polymorphisms. Am J Hum Genet 71:1242
Roach JC, Glusman G, Hubley R, Montsaroff SZ, Holloway AK, Mauldin DE, Srivastava D, Garg V, Pollard KS, Galas DJ et al (2011) Chromosomal haplotypes by genetic phasing of human families. Am J Hum Genet 89:382–397
Sachidanandam R, Weissman D, Schmidt SC, Kakol JM, Stein LD, Marth G, Sherry S, Mullikin JC, Mortimore BJ, Willey DL et al (2001) A map of human genome sequence variation containing 1.42 million single nucleotide polymorphisms. Nature 409:928–933
Salem RM, Wessel J, Schork NJ (2005) A comprehensive literature review of haplotyping software and methods for use with unrelated individuals. Hum Genomics 2:39
Schatz MC (2009) Cloudburst: highly sensitive read mapping with mapreduce. Bioinformatics 25:1363–1369
Scheet P, Stephens M (2006) A fast and flexible statistical model for large-scale population genotype data: applications to inferring missing genotypes and haplotypic phase. Am J Hum Genet 78:629–644
Stephens M, Smith NJ, Donnelly P (2001) A new statistical method for haplotype reconstruction from population data. Am J Hum Genet 68:978–989
Suk EK, McEwen GK, Duitama J, Nowick K, Schulz S, Palczewski S, Schreiber S, Holloway DT, McLaughlin S, Peckham H et al (2011) A comprehensively molecular haplotype-resolved genome of a European individual. Genome Res 21:1672–1685
Taylor RC (2010) An overview of the hadoop/mapreduce/hbase framework and its current applications in bioinformatics. BMC Bioinformatics 11(Suppl 12):S1
Tewhey R, Bansal V, Torkamani A, Topol EJ, Schork NJ (2011) The importance of phase information for human genomics. Nat Rev Genet 12:215–223
Tregouet D, Escolano S, Tiret L, Mallet A, Golmard J (2004) A new algorithm for haplotype-based association analysis: the stochastic-EM algorithm. Ann Hum Genet 68:165–177
Wang RS, Wu LY, Li ZP, Zhang XS (2005) Haplotype reconstruction from SNP fragments by minimum error correction. Bioinformatics 21:2456–2462
Wang TC, Taheri J, Zomaya AY (2012) Using genetic algorithm in reconstructing single individual haplotype with minimum error correction. J Biomed Inform 45:922–930
Wang Y, Feng E, Wang R (2007) A clustering algorithm based on two distance functions for MEC model. Comput Biol Chem 31:148–150
Weale ME (2004) A survey of current software for haplotype phase inference. Hum genomics 1:141–145
Wu J, Wang J, Chen J (2008) A genetic algorithm for single individual SNP haplotype assembly. In: Young computer scientists, 2008. ICYCS 2008. The 9th International Conference for, IEEE, pp 1012–1017
Wu J, Wang J, Chen J (2013) A heuristic algorithm for haplotype reconstruction from aligned weighted SNP fragments. Int J Bioinform Res Appl 9:13–24
Wu LY, Li Z, Wang RS, Zhang XS, Chen L (2009) Self-organizing map approaches for the haplotype assembly problem. Math Comput Simulat 79:3026–3037
Xie M, Wang J, Chen J (2008) A model of higher accuracy for the individual haplotyping problem based on weighted snp fragments and genotype with errors. Bioinformatics 24:i105–i113
Xie M, Wang J, Jiang T (2012) A fast and accurate algorithm for single individual haplotyping. BMC Syst Biol 6(Suppl 2):S8
Xing EP, Jordan MI, Sharan R (2007) Bayesian haplotype inference via the Dirichlet process. J Comput Biol 14:267–284
Xu XS, Li YX (2012) Semi-supervised clustering algorithm for haplotype assembly problem based on MEC model. Int J Data Min Bioinform 6:429–446
Yang WY, Hormozdiari F, Wang Z, He D, Pasaniuc B, Eskin E (2013) Leveraging reads that span multiple single nucleotide polymorphisms for haplotype inference from sequencing data. Bioinformatics 29:2245–2252
Zhao YY, Wu LY, Zhang JH, Wang RS, Zhang XS (2005) Haplotype assembly from aligned weighted SNP fragments. Comput Biol Chem 29:281–287
Zhi D, Wu J, Liu N, Zhang K (2012) Genotype calling from next-generation sequencing data using haplotype information of reads. Bioinformatics 28:938–946