An overview of SNP interactions in genome-wide association studies

Pei Li, Maozu Guo, Chunyu Wang, Xiaoyan Liu and Quan Zou

Advance Access publication date 19 September 2014

Abstract

With the recent explosion in high-throughput genotyping technology, the amount and quality of single-nucleotide polymorphism (SNP) data has increased exponentially. Therefore, the identification of SNP interactions that are associated with common diseases is playing an increasing and important role in interpreting the genetic basis of disease susceptibility and in devising new diagnostic tests and treatments. However, because these data sets are large, although they typically have small sample sizes and low signal-to-noise ratios, there has been no major breakthrough despite many efforts, making this a major focus in the field of bioinformatics. In this article, we review the two main aspects of SNP interaction studies in recent years—the simulation and identification of SNP interactions—and then discuss the principles, efficiency and differences between these methods.

Keywords: genome-wide association studies; SNP interactions; data simulation; detection methods

INTRODUCTION

With the completion of the human genome project and the international haplotype map project (HapMap), as well as the rapid development of high-throughput sequencing technology, the research focus on complex diseases has gradually shifted to genome-wide association studies (GWAS). GWAS are mainly case-control studies that examine single-nucleotide polymorphisms (SNPs) to determine genetic factors associated with complex diseases. The first article using GWAS in age-related macular degeneration was published in *Science* in 2005 [1]. In the past 10 years, GWAS have been performed on many different common diseases, including cancer, cardiovascular disease, gastrointestinal disease, rheumatism, skin diseases and infectious diseases, with a number of notable results [1–11], and investigation into the role of SNPs in complex phenotypes has become a major focus in bioinformatics.

Although GWAS have achieved a number of successes [12, 13], few loci identified have a high or moderate disease risk, and some well-known genetic risk factors have been missed. The relative risk of most new loci is only 1.1–1.2, which suggests that these individual SNPs have a small effect on the heritability of complex diseases [14], and that a large subset of SNPs associated with complex diseases has still not been identified. This is owing to a number of factors. First, pathogenic SNPs have a low population frequency, making them difficult to identify by GWAS using relatively small sample sets [15]. Another reason is that many studies use single-locus tests, in which each locus is tested independently for association with a phenotype, ignoring the combined effect of multiple loci on disease susceptibility [16].

In fact, complex diseases or traits are often affected by the combined action of many genes, each with a small effect, which can be determined by examining...
the interaction of multiple SNPs [17]. In recent years, an increasing number of studies have also shown that SNP interactions are one of the major genetic components of complex diseases [17-20]. Many loci associated with complex diseases identified by GWAS cannot be replicated with different data sets, suggesting SNP interactions within affected individuals [21]. Researchers have found that SNP interactions play a role in Alzheimer’s disease [22], breast cancer [23, 24], schizophrenia [25] and other diseases [26]. Moreover, as the interactions can be multidimensional in addition to two-dimensional, the examination of SNP interactions has great significance to the study of complex diseases.

However, because SNP data are typically derived from small sample sets with large numbers of reads and a low signal-to-noise ratio, SNP interaction detection is typically a ‘needle-in-a-haystack’ problem [27]. There are two primary problems to address when examining SNP interactions. First, because of a lack of benchmark data, it is difficult to design and assess them analytically; data simulation is therefore the most practical means for evaluating performance [28-30]. Second, traditional methods are not highly efficient and miss many loci associated with complex diseases [31], requiring a more holistic approach for detecting SNP interactions. In this article, we review the research and discuss the achievements in both the simulation and identification of SNP interactions.

BACKGROUND: SNP INTERACTIONS
SNP interactions are based on the concept of epistasis [32], first defined by Bateson in 1909, which refers to a phenotype controlled by two pairs of alleles, whereby one allele can suppress the other. Although Bateson’s epistasis is similar to the genetics concept of allele interaction, it does not explain the genetic aetiology of complex diseases. In 1918, Fisher [33] proposed that the epistasis effect is caused by the interaction of two non-alleles, the contribution of which is the deviation from the additive effect of a single locus, which is better at explaining the aetiology of complex disease [34]. In 1954, based on this definition, Cockerham [35] divided epistasis into three types: additive × additive, additive × dominant and dominant × dominant. In 1998, Philips [36] further extended this into six types: dominant epistasis, recessive epistasis, duplicate genes with cumulative effect, duplicate dominant genes, duplicate recessive genes and dominant and recessive interaction. Recently, epistasis has been widely used in biology, and the connotation has been expanded to three types [33]: genetic epistasis, biological epistasis and statistical epistasis.

Because of the shifting definitions of epistasis used by different scholars in different fields and our limited understanding of the effects of epistasis, it has been suggested that the term epistasis should be limited to its original meaning—genetic epistasis [36]. Thus, biological epistasis should be replaced by the term protein–protein interaction, and statistical epistasis replaced by SNP interactions, defined as the total effect of non-linear interactions of multiple SNPs associated with a phenotype.

If one SNP has a marginal effect on a phenotype, it is known as an SNP interaction displaying marginal effects. In some cases, however, each individual SNP has no effect on the phenotype, but the combination has a strong effect; this is known as SNP interactions displaying no marginal effects (INME). Figure 1 shows an example of INME; SNP1 and SNP2 have the same frequency in cases and controls for different genotypes and thus have no effect on the phenotype. However, the combination of both (e.g. AABB, Aabb, aabb) has a profound effect.

DATA SIMULATION OF SNP INTERACTIONS
Data simulation of SNP interaction plays an important role in assessing and designing analytical methods, as well as in studying hypotheses on genomic variation affecting disease. Ultimately, such simulations are a bridge to real SNP interaction data; only when methods for identifying SNP interactions can be validated using benchmark data can they accurately and effectively identify interactions in real data. For these reasons, it is important to develop simulators with the capability of accurately and efficiently generating complex genomic data under realistic scenarios, including the effects of natural selection, recombination and sophisticated demographics and environmental factors. Currently, although there are a variety of software packages for SNP interaction data simulation, many can be placed into three basic types [29, 30, 37]: coalescent, forward and resample. We will examine each in turn and discuss their primary features and differences. All available software in this section can be found in Table 1.
Coalescent simulation

Coalescent theory was introduced by Kingman [47, 48] in 1982 and has since been extended to include recombination, selection and other complex evolutionary models. The essential characteristic of the coalescent model is that it only models the ancestry of genes with neutral alleles, providing a way to efficiently simulate the ancestry of a sample of neutral alleles in large populations. In general, a coalescent simulation has two processes: coalescent and permutation (Figure 2).

In coalescent process, all SNP alleles are traced in seed data to their most recent common ancestor (MRCA). This is usually implemented using the Wright–Fisher model, which provides a dynamic description of the evolution of an idealized population and the transmission of genes from one generation to the next. Based on this model, for a population of $N$ diploid samples, $i$ copies of allele $a$, and $(2N-i)$ copies of allele $A$ in the population, the distribution of the frequency of allele $a$ in the next generation is calculated as follows [49]:

$$Pr(j|i) = \binom{2N}{j} \left( \frac{i}{2N} \right)^j \left( 1 - \frac{i}{2N} \right)^{2N-j}$$

To trace samples to their MRCA, a path is designed to simulate the trajectory starting from the

---

**Figure 1**: An example of the INME disease model. The left represents the frequency in cases and controls for each individual SNP, and each individual SNP has no effect on the phenotype. The right represents the frequency in cases and controls for the combinations of these two SNPs. The dark-shaded cells represent combinations that have a strong effect, whereas the light-shaded cells represent combinations that have a small effect. (A colour version of this figure is available online at: http://bfg.oxfordjournals.org)

**Table 1**: Data simulation software for SNP interactions

<table>
<thead>
<tr>
<th>Framework</th>
<th>Purpose</th>
<th>Simulator</th>
<th>Web link</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>Coalescent</td>
<td>Modelling the genealogy of the observed data</td>
<td>CoaSim</td>
<td><a href="http://www.daimi.au.dk/~mailund/CoaSim/">http://www.daimi.au.dk/~mailund/CoaSim/</a></td>
<td>[38]</td>
</tr>
<tr>
<td></td>
<td></td>
<td>GENOME</td>
<td><a href="http://www.wsph.umich.edu/csg/liang/genome/">http://www.wsph.umich.edu/csg/liang/genome/</a></td>
<td>[39]</td>
</tr>
<tr>
<td></td>
<td></td>
<td>MuCS</td>
<td><a href="http://www.hsc.usc.edu/~garykche/">http://www.hsc.usc.edu/~garykche/</a></td>
<td>[40]</td>
</tr>
<tr>
<td>Forward-time</td>
<td>Modelling any complex evolutionary and</td>
<td>SimuPOP</td>
<td><a href="http://simupop.sourceforge.net/">http://simupop.sourceforge.net/</a></td>
<td>[41]</td>
</tr>
<tr>
<td></td>
<td>demographic scenarios</td>
<td>Quantinemo</td>
<td><a href="http://www2.unil.ch/popgen/softwares/quantinemo/">http://www2.unil.ch/popgen/softwares/quantinemo/</a></td>
<td>[42]</td>
</tr>
<tr>
<td></td>
<td></td>
<td>genomeSIMLA</td>
<td><a href="http://ritchielab.psu.edu/ritchielab/current-research/genomesimla/">http://ritchielab.psu.edu/ritchielab/current-research/genomesimla/</a></td>
<td>[43]</td>
</tr>
<tr>
<td>Resampling</td>
<td>Retaining patterns of real data</td>
<td>HAPGEN</td>
<td><a href="http://www.stats.ox.ac.uk/~marchini/software/gwas/gwas.html">http://www.stats.ox.ac.uk/~marchini/software/gwas/gwas.html</a></td>
<td>[44]</td>
</tr>
<tr>
<td></td>
<td></td>
<td>GWAsimulator</td>
<td><a href="http://biostat.mc.vanderbilt.edu/wiki/Main/GWAsimulator">http://biostat.mc.vanderbilt.edu/wiki/Main/GWAsimulator</a></td>
<td>[45]</td>
</tr>
<tr>
<td></td>
<td></td>
<td>HAP-SAMPLE</td>
<td><a href="http://wwwhapsample.org/">http://wwwhapsample.org/</a></td>
<td>[46]</td>
</tr>
</tbody>
</table>
current generation and going backward until the allele is lost (denoted by $H = \{i_T, i_{T-1}, i_{T-2}...i_2, i_1\}$, where $i_T = i, ... i_1 = 1$ and $i_0 = 0$). Assuming a Markov chain for transitions from one generation to the next, the transition probability from generation $t$ to $t-1$ can be calculated as follows [49]:

$$
\Pr(i_{t-1} | i_t) = \frac{2N_{t-1}}{i_{t-1}} \left( \frac{i_t}{2N_t} \right)^{i_t} \left( 1 - \frac{i_t}{2N_t} \right)^{2N_{t-1} - i_{t-1}}
$$

where $N_t$ denotes the population size and $i_t$ denotes the copies of the allele at generation $t$. The probability of a given sample path can be found by multiplying the transition probabilities as follows:

$$
\Pr(H) = \prod_{t=1}^{T} \Pr(i_{t-1} | i_t)
$$

The permutation process describes when and how alleles mutate over time and generate simulated data through evolutionary scenarios of mutation originating from the MRCA. Assuming that each mutation occurs uniformly and independently, mutations follow a Poisson distribution, and the expected number of mutations on the $i$-th branch of the genealogy tree is calculated as follows [50]:

$$
E(M_i) = \theta \tau_i
$$

where the parameter $\theta = 4N_e \mu$, $N_e$ is the effective population size, $\mu$ is the chromosomal mutation rate and $\tau_i$ is the length of the $i$-th branch.

Because coalescent simulations are based on the history of lineages with offspring surviving in the current population, they are computationally efficient and can thus be applied to simulating large DNA regions. Consequently, coalescent simulations play an integral role in investigating sample properties and evaluating association studies. Such simulations have also been extended for modelling more realistic scenarios of evolution and demography, and a number of simulator packages are now available. These include CoaSim [38], which generates simulation data based on a Monte Carlo approach, the GENOME software package [39], which can accurately simulate SNP data with linkage disequilibrium (LD) and the minor allele frequency (MAF) via recombination parameters, and the MaCS [40] software package, which can simulate evolutionary scenarios of a haplotype block based on Markov chain data.

The main weakness of these simulators is that they can only model the genealogy of observed samples, thereby ignoring unsampled members of the population. Accordingly, they are not suitable for tracking complete ancestral information. Furthermore, it is difficult for the coalescent framework to be incorporated into some selection scenarios, such as multilocus selection, and thus it cannot simulate complex SNP interactions in those data sets.
**Forward-time simulation**

Unlike coalescent-based simulation, forward-time simulation [51] evolves a population forward in time using arbitrary genetic and demographic factors. This method usually starts from initial seed data and draws solutions from either the final or last several generations after many generations under various evolutionary and demographic scenarios. Within this framework, complete ancestral information can be tracked, and sample properties can be investigated at any generation. Because it closely mimics the complex evolutionary histories of human populations, this method can, in theory, simulate arbitrarily complex genetic samples and can thus be incorporated in complex disease models to simulate any evolutionary and demographic scenarios.

Because evolutionary scenarios such as natural selection, mutation and recombination have a substantial effect on this method, the implementation of these evolutionary scenarios is critical and has effects on allele frequency and correlation. As alleles follow the Hardy–Weinberg principle (HWP), with a population of \( N_t \) diploid individuals at generation \( t \) and focusing on a locus with major allele \( A \) and minor allele \( a \) with a MAF of \( p_s \) at generation \( t \), the frequency of genotypes \( AA, Aa \) and \( aa \) is \((1 - p_s)^2\), \(2p_sp(1-p_s)\) and \(p_s^2\), respectively. Considering only natural selection and extending Slakkin’s work [49], the frequency of allele \( a \) in the next generation \((t+1)\) is given by

\[
p_{t+1} = \frac{2p_t(1-p_t)(1+s_2) + 2p_t^2}{2(1-p_t)^2(1+s_1) + 4p_t(1-p_t)(1+s_2) + 2p_t^2} + \varepsilon
\]

where \(1 + s_1, 1 + s_2 \) and \(1\) denote the relative fitness of the three genotypes \( AA, Aa \) and \(aa\). \( s_1 \) and \( s_2 \) are \(> -1\), and \( \varepsilon \) is related to the population size \(N_{t+1}\).

If we assume that mutations occur randomly at generation \(t+1\), using a two-allele mutation model, the MAF is given by

\[
p_{t+1} = p_{t+1} + \mu(1-2p_{t+1})
\]

where the variable \( \mu \) denotes the mutation rate. Forward-time simulations evaluate allelic correlation by LD patterns and always use the mean LD over genome distance to characterize LD features, as follows:

\[
LD_{\text{mean}} = \sum_{i=1}^{M-1} \sum_{j=i+1}^{M} \frac{LD_{ij}}{\binom{M}{2}}
\]

where \(LD_{ij}\) denotes the LD coefficient between marker \(i\) and marker \(j\), and \(M\) is the genome distance.

Recombination occurs only when the current generation produces offspring for the next generation, and the probabilities of recombination between any two markers can be calculated by Kosambi’s mapping function. The total number of recombination events between any pair of chromosomes follows a Poisson distribution, allowing the calculation of the expected frequency as follows:

\[
E(R) = \sum_{i=1}^{l-1} \sum_{j=i+1}^{l} RP_{ij}
\]

where \(RP_{ij}\) denotes the recombination probability between marker \(i\) and marker \(j\), and \(l\) is the length of the chromosome.

Although forward simulation is less efficient than coalescent simulation, it has some advantages. First, all the ancestral information of samples can be tracked, though this is computationally intensive; this can be helpful for investigating the evolution of an entire population. Second, forward simulations can incorporate evolutionary scenarios with different types of selections, variable population sizes and recombination and thus can simulate any evolutionary or demographic scenario as well as complex disease models. Hence, there is an increasing interest in forward simulation, and many have been developed. Examples include SimuPOP [41], which can simulate SNP interactions, QuantiNemo [42], which incorporates complex evolutionary scenarios including selection, mutation, recombination and migration, and genomeSIMLA [43], which can simulate SNP interactions using LD patterns in both family-based and case–control data sets. However, challenges above and beyond computation efficiency remain. Indeed, the results of forward simulation depend heavily on the initial population used, making the selection of an appropriate sample...
crucial. Additionally, controlling for the disease allele frequency is still being researched.

**Resampling simulation**

Unlike the coalescent and forward-time methods, which require modelling complex evolutionary and demographic scenarios, resampling simulations generate simulation data by random selection from seed data within a disease model. Therefore, this method can generate arbitrarily large simulated data sets to meet research needs.

The resampling simulation method is based on observed data; thus, there is a relationship between the simulated and observed data, such as allele frequency and LD patterns. As the haplotype frequency of \( ab \) in the samples follows a binomial distribution, the deviation from the expected frequency of haplotype \( ab \) in the simulated data for two SNP markers with minor alleles \( a \) and \( b \) is given by

\[
p_h - \sqrt{\frac{p_a(1-p_a)}{n}} \leq p' \leq p_h + \sqrt{\frac{p_b(1-p_b)}{n}}
\]

where \( p'_h \) denotes the expected frequency of haplotype \( ab \) in the simulated data and \( p_h \) denotes the expected frequency of haplotype \( ab \) in the observed data. Deviation in LD patterns in the simulated data for two SNP markers is given by

\[
r^2 = \frac{(p'_h - p'_{apb})^2}{p'_a(1-p'_a)p'_b(1-p'_b)}
\]

where \( p'_a \) and \( p'_b \) denote the frequencies of alleles \( a \) and \( b \), respectively, in the simulated data.

Because they do not model complex evolutionary and demographic scenarios, resample simulations are more efficient than coalescent or forward-time simulations. Because this method is based on an existing data set, it is useful for validating genotype–phenotype association studies and evaluating the effects of methods for SNP interaction detection. Resample simulation methods include the HAPGEN algorithm [44], GWAsimulator [45] and the HAP-SAMPLE simulator [46], which implement a simulation of a single disease model. However, resampling simulation is not suitable for observing evolutionary processes or studying demographical structures because it generates simulated data by sampling from an existing data set without establishing evolutionary and demographic scenarios.

**METHODS FOR SNP INTERACTION DETECTION**

Two challenges must be overcome to successfully detect loci associated with complex diseases using a genome-wide approach [27, 52]. First, complex traits are often affected by the interaction of many genes with small effects and environmental factors, which often present a non-linear high-dimensional effect [17]. The traditional linear parametric method often focuses on single-locus analysis and the detection of genes with marginal effect, which is insufficient to capture these high-dimensional interactions [31]. Second, the nature of SNP data sets makes SNP interaction identification a combinatorial search problem using small samples sizes with huge amounts of data, resulting in searches that are computationally intensive because of the massive number of possible combinations. Therefore, many traditional methods are difficult to adapt to genome-wide data sets. Given the interest in SNP interaction identification in recent years, various methods have been proposed focusing on either solving one of the two challenges or finding a balance between these two difficulties. All the available methods or tools listed in this section can be found in Table 2.

**Modelling challenge**

Traditional linear parametric methods for detecting SNP interactions have poor performance because of their focus on single-locus analysis. In some cases, each individual locus has no effect on phenotype, but the combination of SNPs produces a strong effect, known as INME. This makes traditional methods useless for SNP interaction detection [31].

To circumvent these limitations, many data mining and machine learning methods have been proposed, including logic regression [54], Random Forest [64, 65], Multifactor-dimensionality reduction (MDR) [53], bayesian epistasis association mapping [66] and neural networks [67, 68]. Although these methods use different models and algorithms, all are focused on detecting high-dimensional multilocus interaction and can be better adapted to complex scenarios with INME disease models. We will focus on two example methods, MDR and logic regression, which are more classical and widely
used in this area, to demonstrate their principles, strengths and weaknesses.

MDR

MDR [53] was proposed to solve the problem that when detecting high-dimensional interactions, traditional linear methods involve many empty table cells, leading to an increase in standard deviation, parameter estimation error and type I error. MDR detects SNP interactions through an exhaustive search of the combinations of $n$ genetic or environmental factors that may be associated with disease or traits, reducing the dimensionality by dividing these factors into low-risk or high-risk combinations according to their effect on disease susceptibility. MDR is usually applied to the experimental design for the study of case–control or phenotypic discordant sib pairs.

The process of MDR is shown in Figure 3. First, the data set must be divided into multiple (usually 10-fold) partitions for cross-validation. Typically, in the case of 10-fold cross-validation, 90% of the data is the training set, and the remaining 10% is the testing set. Second, an exhaustive list of the $n$ combinations of variables, which can be environmental factors or loci, is generated. Third, as shown in Figure 3, each of the $n$ combinations is next organized into a contingency table, with each cell representing multifactorial combinations, and the number of cases and controls for each cell in the table is calculated. In the fourth step, the ratio of cases to controls in each cell is calculated. If the ratio is higher than the threshold (usually 1), this multifactorial combination is labelled as high risk; otherwise, it is categorized as low risk. In the fifth step, each of the $n$ combinations is used to build a model using the labelled training set, and the training accuracy is calculated. The top-scoring model is selected for validation using the testing set, and the testing accuracy is calculated. Finally, these steps are repeated for each cross-validation, and the average testing accuracy is calculated. Then, among all the k-locus models, the model with the highest cross-validation consistency is chosen as the best.

The MDR method is non-parametric and model free, without the necessity of assuming a specific genetic model. This method can effectively identify high-dimensional SNP interactions in the absence of marginal effects and also adapts well to identify SNP interactions in environments with noise, making it robust. However, the MDR method is computationally very intense. Therefore, the amount of data that can be analysed by MDR largely depends on the computing resources available. The results from this method are difficult to interpret, and the data must be balanced [69].

The main technical characteristics of the MDR method are the reduction in data dimensionality and cross-validation; however, these two characteristics are both its strength and its weakness. Data dimensionality reduction can effectively reduce the computation resources needed but is too simple to make the results easy to explain. Cross-validation improves confidence in the results but also increases the computation. Despite these possible weaknesses, comparing the MDR method with the penalized
logistic regression method, the former method can capture a more complex pattern of SNP interactions than the latter method [70].

Since its proposal by Richie in 2001, the MDR method has been modified and extended by many researchers. Moore proposed an entropy-based interpretation method [71] based on an MDR strategy, and Lee [72] and Lou [73] applied regression strategy to the MDR method, which extends its application to non-binary and continuous variables. To identify more multilocus interactions, Pattin [74] used protein–protein interaction networks to reduce the computation of the MDR method. Velez [75] improved the MDR method and extended it to unbalanced data. Furthermore, Namkung [76] modified this model and extended it with the ability to address missing data.

**Logic regression**

Logic regression [54] is a generalized regression model designed for data with many binary predictor variables and is used mainly to detect high-dimensional gene–gene or gene–environment interactions. This method is usually applied to case–control or case–only studies. Although SNP data variables are not binary, they can be converted to binary variables. If each individual SNP has a value of 0, 1 or 2 for the variant alleles, each individual SNP can be converted into two binary variables: $X_{id} = 1$ if the SNP has at least one variant allele, and $X_{ir} = 1$ if the SNP has two variant alleles. The other SNPs are assigned 0. Thus, $X_{id}$ represents the dominant effect of SNP $S_i$, and $X_{ir}$ represents the recessive effect of SNP $S_i$. After transformation into binary variables, the logic regression model can be used to detect gene–gene or gene–environment interactions.

Logic regression constructs new predictors as logical combinations of binary variables based on new predictor variables. In this method, combinations of binary variables are usually represented by logic trees, whereby each leaf represents a binary variable and each knot represents the binary operators ‘and’ or ‘or’. Figure 4 shows a logic tree representing combinations of binary variables. Logic trees can be used not only as a graphical representation of logic expression but also to search for a better model by

---

**Figure 3:** The process of MDR consists of six steps; each step is explained in the text. (A colour version of this figure is available online at: http://bfg.oxfordjournals.org)
modifying the trees to generate new ones. Logic regression can be described as follows:

\[ g(E(Y)) = \beta_0 + \sum_{i=1}^{n} \beta_i L_i \]

where \( g \) is a link function, \( Y \) is the response, \( L_i \) represents a combination of binaries (logic tree) and \( \beta_i \) denotes the regression parameters. In general, \( g \) is always given by \( \text{Logit}(x) = \log\left(\frac{x}{1-x}\right) \), where \( x \in (0, 1) \).

Logic regression uses a simulated annealing algorithm to find a good-fitting model from the large search space of possible models. Typically, after a logic model is initialized, a candidate model is generated at each iteration by randomly changing the current logic trees by replacing, adding or removing a leaf or pruning or growing a branch. The regression parameters of this candidate model can be estimated as follows: \(-2\max g(\hat{\beta})\), where \( \hat{\beta} \) is the regression parameter and \( f \) is the log likelihood. If this model is better, then it is accepted as the current optimal model; otherwise, it is accepted with a certain probability. This algorithm begins with a high probability of accepting a worse result but avoids falling into the local minimum. Later, as the iteration increases, the probability of accepting a worse result also decreases. To avoid over-fitting, logic regression selects the best model by cross-validation.

The logic regression model is able to effectively detect high-dimensional SNP interactions, and its results are easily interpreted. However, this model presents many problems when factors increase, such as increased susceptibility to over-fitting, model recession, too many empty cells in the contingency table for detection of multifactorial interactions and computational intensity. These problems become more significant as the dimensions of interactions increase [77, 78]. To solve these problems, many researchers have recently extended and improved this model. Park [77] used penalty logic regression, which effectively mitigated the problem of too many empty cells in the contingency table. Kooperberg [79] proposed Monte Carlo logic regression by combining logic regression with the Markov chain Monte Carlo method, which effectively mitigates the problems of model recession and over-fitting. In addition, Fritsch [80] modified the logic regression model by combining it with the Bayesian method and proposed a full Bayesian logic regression method. Combining the logic regression method with bootstrap analysis, Schwender [81] designed the LogicFS method. Finally, Purcell [82] designed the Plink software to detect SNP interactions on a genome-wide scale.

**Computational challenges**

Methods to circumvent the computational challenges inherent in determining SNP interactions can be divided into two groups: attribute selection [27] and parallel processing. The attribute selection strategy reduces the size of the data set to improve computational efficiency by prescreening SNPs using a priori information. Parallel processing improves computational efficiency by increasing computational resources, effectively reducing search time and allowing searches in the complete space of multifactorial combinations.

**Attribute selection**

There are two primary approaches for attribute selection [27]—filter and wrapper approaches. The filter approach uses algorithms to assess the quality of each variable and then uses this information to select a subset for analysis. The wrapper approach randomly selects a subset of variables to classify and evaluate quality at each iteration until a good-quality subset of variables is obtained. With some learning algorithms, the wrapper approach can select a better subset of variables than the filter approach but is less efficient.

Many methods have used the attribute selection strategy. The INTERSNP programme [55] uses a priori information of statistical evidence (single marker association at a moderate level), genetic
Parallel processing

In recent years, significant effort has focused on the development of parallel SNP interaction detection methods, with some notable progress. Plink [59] has been extended for parallelization to create FastEpistasis for the detection of epistatic effects of a continuous phenotype. This has been scaled to data sets of 5000 individuals with 500 000 SNP sites each. Ma [60] modified Mao’s [84] method by parallel extension to create EPISNPmpi, which required ~20 h to complete the detection of two-dimensional interactions of 100 000 SNPs. In comparison, serial processing of the same data set required 18 h to complete the detection of two-dimensional interactions of 5700 SNPs. eCCEO [61] uses cloud computing to detect epistatic effects, demonstrating the effectiveness and efficiency of the model on a 40-node cluster. Greene [62] modified the MDR method by combining it with the parallel computing advantage of graphics processing unit (GPU) to design the implementation of MDR for GPUs and demonstrated its effectiveness in GWAS of sporadic amyotrophic lateral sclerosis. Combining the advantages of the GPU’s parallel computation and attribute selection strategy, SHEsisEpi [63] effectively improves the efficiency of SNP interaction detection.

Although traditional methods cannot detect at the whole-genome scale because of computational complexity, parallel extensions effectively solve this problem. Parallel processing can effectively enhance the computing power and reduce the time cost of many algorithms. However, substantial computing resources are required, increasing costs, especially for whole-genome data sets.

SUMMARY AND CONCLUSIONS

In the past 10 years, the utilization of SNP markers for GWAS on complex diseases and traits has produced many achievements [1–11] and has become an important tool for determining the genetic basis of many diseases. The research on SNP interactions has rapidly expanded, not only in terms of methodological development and practical applications but also in translating statistical interactions into biological interactions [85]. However, the ‘needle-in-a-haystack’ nature of SNP interactions [27] makes such research challenging, requiring advances in the simulation and detection of SNP interactions.

Three basic simulation frameworks [29, 30, 37]—coalescent, forward and resampling—have been used to study SNP interactions. Coalescent-based simulations model the genealogy of the observed data and can be applied to simulate large DNA regions efficiently; however, complex evolutionary scenarios, such as natural selection and recombination, are difficult to model. Forward-time simulation can model many complex evolutionary and demographic scenarios but is less efficient. Resampling simulations retain patterns of real data, such as allele frequency and LD patterns, and are more efficient.

The identification of SNP interactions requires advances in modelling and computation [27, 52]. Two methods, MDR and logic regression, can effectively model high-dimensional SNP interactions but are less efficient. New computational methods based on attribute selection or parallel processing...
attempt to balance the computational cost and effects to increase SNP interaction detection.

Further advances in both computing power and algorithm efficiency will allow for the analysis of increasingly large SNP data sets. This will increase the sensitivity and specificity of SNP interaction detection, providing new insights into the molecular pathogenesis of a number of diseases.

**Key points**

- We have concluded the simulation methods of SNP interaction in GWAS analysis, which is important for evaluating the SNP interaction detecting method.
- We have reviewed the latest SNP interaction identification methods, and discuss the principles, efficiency and differences between them.
- We discuss about the future research of GWAS research in bioinformatics and give our suggestion.

**FUNDING**

Q.Z. and P.L. are supported by Natural Science Foundation of China (61370010) and the Natural Science Foundation of Fujian Province of China (2014J01253). M.G. and C.W. are supported by Natural Science Foundation of China (61271346), Specialized Research Fund for the Doctoral Program of Higher Education of China (20112302110040) and Fundamental Research Funds for the Central Universities (HIT.KISTP.201418). X.L. is supported by Natural Science Foundation of China (61172098 and 91335112).

**References**

24. Chuang LY, Lin YD, Chang HW, et al. SNP-SNP Interaction Using Gauss Chaotic Map Particle Swarm...


