BIOINFORMATICS ORIGINAL PAPER Vol. 25 no. 4 2009, pages 504–511
doi:10.1093/bioinformatics/btn652

Genetics and population analysis

SNPHarvester: a filtering-based approach for detecting epistatic interactions in genome-wide association studies

Can Yang[1,*], Zengyou He[1], Xiang Wan[1], Qiang Yang[2], Hong Xue[3] and Weichuan Yu[1]

[1]Laboratory for Bioinformatics and Computational Biology, Department of Electronic and Computer Engineering, [2]Department of Computer Science and Engineering and [3]Department of Biochemistry, The Hong Kong University of Science and Technology, Clear Water Bay, Kowloon, Hong Kong, China

Received on September 19, 2008; revised on November 15, 2008; accepted on December 17, 2008

Advance Access publication December 19, 2008

Associate Editor: Alex Bateman

ABSTRACT

Motivation: Hundreds of thousands of single nucleotide polymorphisms (SNPs) are available for genome-wide association (GWA) studies nowadays. The epistatic interactions of SNPs are believed to be very important in determining individual susceptibility to complex diseases. However, existing methods for SNP interaction discovery either suffer from high computation complexity or perform poorly when marginal effects of disease loci are weak or absent. Hence, it is desirable to develop an effective method to search epistatic interactions in genome-wide scale.

Results: We propose a new method SNPHarvester to detect SNP–SNP interactions in GWA studies. SNPHarvester creates multiple paths in which the visited SNP groups tend to be statistically associated with diseases, and then harvests those significant SNP groups which pass the statistical tests. It greatly reduces the number of SNPs. Consequently, existing tools can be directly used to detect epistatic interactions. By using a wide range of simulated data and a real genome-wide data, we demonstrate that SNPHarvester outperforms its recent competitor significantly and is promising for practical disease prognosis.

Availability: http://bioinformatics.ust.hk/SNPHarvester.html

Contact: eeyang@ust.hk

Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 INTRODUCTION

Despite the great success in identifying genes responsible for Mendelian diseases, the nature of non-Mendelian (or complex) diseases remains mysterious. Because of the sophisticated regulatory mechanism encoded in the human genome, it is widely agreed that complex traits are typically caused by the joint effects of multiple genetic variations instead of a single genetic variation. These multiple genetic variations may show very little effect individually but strong interactions jointly, which is known as epistasis or multilocus interaction (Cordell, 2002). Recently, an increasing number of research has reported the presence of epistatic interactions in complex diseases (Griffiths *et al*., 2008), such as breast cancer (Ritchie *et al*., 2001) and type-2 diabetes (Cho *et al*., 2004).

*To whom correspondence should be addressed.

Nowadays, the genome-wide association (GWA) studies have produced hundreds of thousands of single nucleotide polymorphisms (SNPs) (WTCCC, 2007) and become a powerful approach to identify genes involved in common human diseases (Hirschhorn and Daly, 2005; McCarthy *et al*., 2008; Wang *et al*., 2005). Recently, some computational methods have been proposed to address this issue [see Liang and Kelemen (2008) and Musani *et al*. (2007) for comprehensive reviews]. Based on their optimization strategies, they can be roughly divided into three categories:

(1) Brute-force search methods.

(2) Greedy search methods.

(3) Stochastic search methods.

In this article, we propose a new stochastic search method named SNPHarvester to search for significant SNP groups in large-scale association studies. The main advantage of SNPHarvester is that it can select a set of significant SNP groups from hundreds of thousands of SNPs efficiently. As a result, existing methods can be applied directly to the selected SNP groups for epistatic interaction detection. We will discuss the relationship between our method and these existing methods in Section 3.

The rest of the article is organized as follows: Section 2 describes the SNPHarvester algorithm in detail. Section 3 discusses the relationship between SNPHarvester and existing methods. Section 4 demonstrates the effectiveness of our method with experiments. Section 5 discusses the issue of incorporating expert knowledge. Section 6 concludes this article.

2 METHOD

In GWA studies, SNPs are high-density bi-allelic markers. We use capital letters (e.g. *A, B,* …) and lowercase letter (e.g. *a, b,* …) to denote the major and minor alleles, respectively. For each SNP, there are three genotypes: *AA, Aa,* and *aa*.

Suppose N_d case samples and N_u control samples have been genotyped at L SNP makers for an association study. The L SNP markers can be partitioned into three classes.

- Class 0: SNPs are unassociated to the disease.
- Class 1: SNPs influence the disease risk independently, i.e. they show marginal effects.
- Class 2: SNPs contribute little effects to the disease risk individually but influence the disease risk jointly. This kind of behavior is known

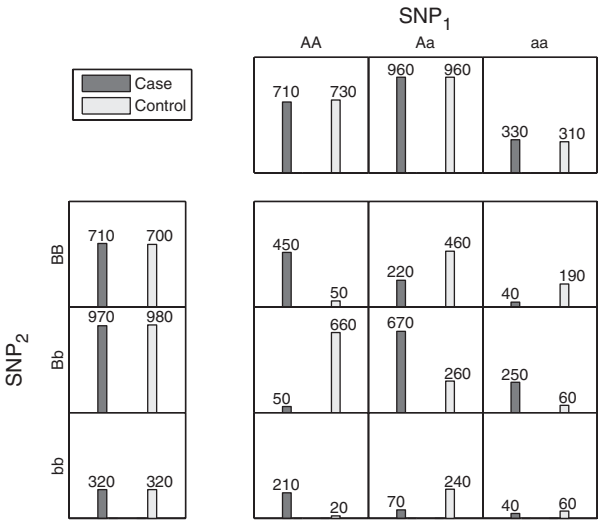

Fig. 1. An illustration of a typical pattern of SNPs in class 2. There are 2000 cases and 2000 controls. Both SNP_1 and SNP_2 have the same distribution in case and control, but their joint distribution is significantly different between case and control.

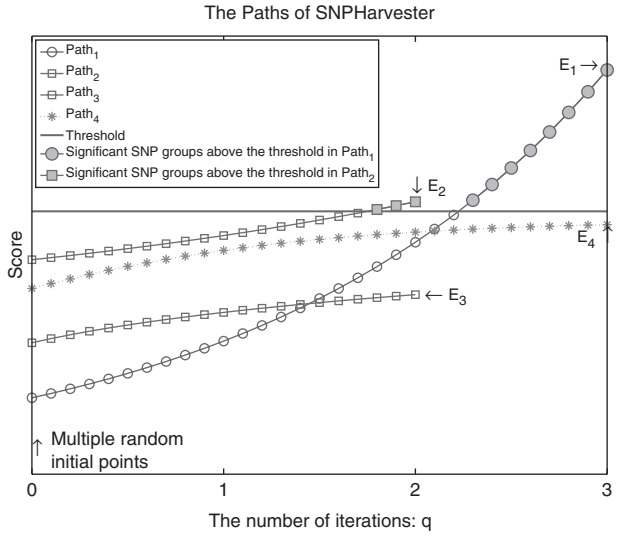

Fig. 2. An illustration of multiple paths generated by SNPHarvester: each path is initialized randomly and the following entire path is created by a local search algorithm PathSeeker. In one iteration, PathSeeker scans the index of all SNPs once. During the iteration, PathSeeker updates the current visited SNP group (denoted by the markers in the figure) so that the score of the new group always increases. Typically, PathSeeker algorithm converges in two or three iterations. All significant SNP groups found by PathSeeker will be recorded as long as their scores are above the statistical threshold. Here four paths are shown and E_i represents the SNP group at the end of $Path_i$.

as epistatic interaction in genetic analysis. Identification of the SNPs in class 2 is computationally challenging.

Figure 1 gives a toy example showing a typical pattern for SNPs in class 2. In practice, the boundary between different classes may not be clear. The membership of a SNP is often determined by thresholding some statistical test scores.

SNPHarvester first identifies disease-associated SNP groups from thousands of SNPs to reduce the number of SNPs. Then, we use L_2 penalized logistic regression models (Park and Hastie, 2008) as a post-processing step to extract SNP interactions from identified SNP groups.

SNPHarvester is based on multiple path generation with a generic score function. It consists of the following key parts:

(1) Multiple paths: multiple epistatic interactions rather than a single one are expected to be found in GWA studies due to the sophisticated regulatory mechanism encoded in the human genome. Therefore, the idea of multiple paths is motivated by the identification of multiple significant SNP groups, as shown in Figure 2.
   - SNPHarvester randomly initializes the starting point of each path and generates the path by a local search algorithm. Random initial points of paths lead to a diverse path-family which contributes to the success of SNPHarvester.
   - We develop a local search algorithm called PathSeeker to generate each path from a random initialization. We are interested not only in the local optimum at the end of a specific path but also in the significant SNP groups along the path. PathSeeker evaluates the score function of each visited SNP group, and records the SNP group whose score passes a fixed threshold (the score function and the threshold will be discussed in the following paragraph). Then, we obtain the significant SNP groups during path generation.

(2) Score function: the score function is defined to measure the association between a k-SNP group and the phenotype. There are a number of reasonable score functions for this purpose, such as the χ^2-value, the classification accuracy (Ritchie *et al*., 2001) and the B-statistic value (Zhang and Liu, 2007). In this article, we use the most popular χ^2-value as our score function and the threshold is determined by Bonferroni corrections. Note that the χ^2-value with degree of freedom 3^k-1 only measures the association between a k-SNP group and the phenotype, a post-processing step is necessary to distinguish SNPs that have interaction effects on the phenotype from those SNPs with marginal effects on the phenotype.

In the following, we first introduce our PathSeeker algorithm which is the core of SNPHarvester. Then we explain the details of SNPHarvester algorithm. Finally, we describe the method of extracting epistatic interactions from the significant SNP groups.

2.1 PathSeeker algorithm

2.1.1 Notation The current visited SNP-group (the active set) is denoted as A whose score is denoted as $Score(A)$. Let SNP_{s_j} be the selected SNP in active set A with index j, where $j=1,\ldots,k$ for k-SNP groups. Let SNP_i be the i-th SNP, where $i=1,\ldots,L$.

2.1.2 PathSeeker The idea of PathSeeker algorithm is to increase $Score(A)$ by updating only one SNP in active set A at a time, as illustrated in Figure 3. Suppose the active set $A=\{SNP_{s_1}, SNP_{s_2},\ldots,SNP_{s_k}\}$ and the SNP to be checked is SNP_i ($SNP_i \notin A$). PathSeeker generates k sets $A_1,\ldots,A_k$, where A_j is obtained by swapping SNP_i with $SNP_{s_j} \in A$. Let $A^* = \arg\max_{A_j, j=1,\ldots,k} Score(A_j)$. If $Score(A^*) < Score(A)$, then keep A unchanged; otherwise, let $A \leftarrow A^*$. PathSeeker will record the active set A if it passes the statistical test. PathSeeker continues the same procedure for SNP_{i+1}. The detail of PathSeeker algorithm is given in Algorithm 1.

The convergence of PathSeeker algorithm is guaranteed in Theorem 1.

THEOREM 1. *PathSeeker algorithm converges to a local optimum E in a finite number of iterations.*

PROOF. There are only a finite number of possible subset of k-way interactions. Each possible subset A appears at most once since the sequence $Score(A)$ is strictly increasing. Hence, the result follows. ■

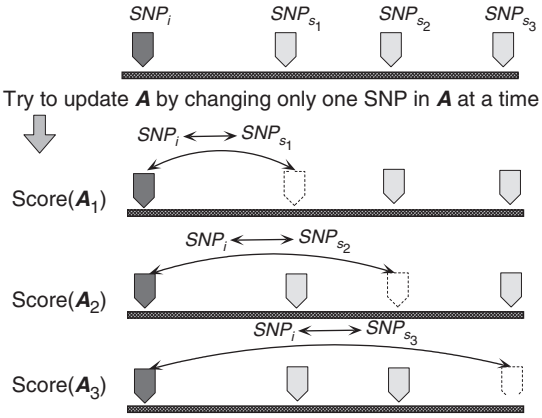

Fig. 3. An illustration of the idea of PathSeeker: PathSeeker tries to update $A=\{SNP_{s_1}, SNP_{s_2}, SNP_{s_3}\}$ by swapping only one SNP in A at a time, where the SNP swapped out of A is denoted by dashed block. For example, $A_1=\{SNP_i, SNP_{s_2}, SNP_{s_3}\}$.

Algorithm 1. PathSeeker algorithm

Notation: q: iteration number.
Input:
D: a dataset of N_d case samples and N_u control samples genotyped at L SNP makers.
k: k-SNP groups.
T: statistical significance threshold.
Output:
M: a collection of k-SNP groups which pass the statistical testing.
E: the local optimum (i.e. the active set at the end of a path).

/* **Phase 1-Initialization** */
Randomly select k SNPs to form an active set A

/* **Phase 2-Iteration** */
Initialize $q=0$ and $IsSwap \leftarrow$ True
while $IsSwap$=True **do**
  $IsSwap \leftarrow$ False
  for $i=1$ to L **do**
    if SNP_i does not belong to A **then**
      $h \leftarrow \arg\max_{j=1,\ldots,k} Score(A+\{SNP_i\}-\{SNP_{s_j}\})$
      if $Score(A+\{SNP_i\}-\{SNP_{s_h}\}) > Score(A)$ **then**
        Update A as $A \leftarrow A+\{SNP_i\}-\{SNP_{s_h}\}$
        $IsSwap \leftarrow$ True
        /* **Harvest the significant interacting SNPs** */
        if $Score(A) > T$ **then**
          Put A into M
        end if
      end if
    end if
  end for
  $q{+}{+}$ /***Record iteration number** */
end while
$E \leftarrow A$ /***Record the local optimum after convergence** */
return M, E

The time complexity of score calculation at each swap evaluation step is $O(kN)$, where k is the number of SNPs in active set A and $N=N_d+N_u$. Accordingly, the PathSeeker algorithm has a time complexity of $O(qkLN)$, where L is the total number of SNP markers and q is the number of iterations. Here one iteration means that index i goes from 1 to L, as shown in Algorithm 1. The time complexity is linear to all parameters. Thus, PathSeeker has good scalability. Empirically, we observe that $q=2$ or 3 for most cases in our experiments.

2.2 SNPHarvester algorithm

SNPHarvester calls PathSeeker multiple times to detect significant associated SNP groups. Specifically, we design SNPHarvester based on the following considerations:

- We first scan L SNP markers once to detect the single significant SNPs based on 2-df χ^2-value after Bonferroni corrections. We then remove all these significant SNPs since we are more interested in searching for epistatic interactions.
- For a fixed k, we generate multiple paths by running PathSeeker multiple times to identify k-way SNP interactions. We use the χ^2-value with 3^k-1 degree of freedom to measure the association between a k-SNP group and the phenotype. Our threshold is determined by the significance threshold $\alpha=0.01$ after Bonferroni corrections. This threshold is known to be conservative.
- We need to remove local optima during path generation. Let E_i be the active set at the end of i-th path. The score of E_i cannot be increased by swapping one SNP $\in E_i$ with SNP $\notin E_i$. Based on this fact, we remove SNPs contained in E_i. This means that the number of SNPs becomes smaller in the later stage of our algorithm.
- We need to decide the range of k. As we use χ^2 to measure the significance of SNP groups, we have to deal with the data sparsity problem as discussed in Musani *et al*. (2007), i.e. many cells in the multi-SNP contingence table only have very small number of samples when k is relatively large. This will lead to inaccurate calculation of the χ^2-value. Therefore, we restrict $k \leq \ln_3 N_d - 1$ in the SNPHarvester algorithm.

SNPHarvester algorithm is given in Algorithm 2.

2.3 Post-processing

Let S be the set of significant SNP groups identified by SNPHarvester. The SNP groups in S do not necessarily have interaction effects on disease status due to the following reasons:

- In the first step of SNPHarvester, we only remove those SNPs having very strong marginal effects based on Bonferroni corrections. Thus, the SNPs with relative small marginal effects remain to be evaluated for high-order interactions. Strong association may be caused by this kind of SNPs.
- The significance of a k-SNP group may be caused by its sub-group.

Therefore, we need a post-processing step to filter out those spurious interactions. We use logistic regression models because marginal effects and high-order interactions can be processed elegantly in the framework of logistic regression models.

Specifically, for a k-SNP group in S, there are $2^k=1+C_k^1+C_k^2+\ldots+C_k^k$ terms to be fitted in the logistic regression model. Since we restrict $k \leq \log_3 N_d - 1$ (typically $k \leq 5$), 2^k would not be a large number. However, it is possible that some cells in the multi-SNP contingency table are zeros, which means that standard logistic regression models cannot be directly applied. Thus, we adopt the L_2 penalized logistic regression (Park and Hastie, 2008) to filter out spurious interactions. Applying this statistical tool is computationally feasible because the number of SNPs has been greatly reduced. Our post-processing steps are described as follows:

(1) For a k-SNP group in S, we use three dummy variables to code each SNP and 3^j dummy variables to code j-way interactions, where $2 \leq j \leq k$.

Algorithm 2. SNPHarvester Algorithm

Input:
D: a data set of N_d case samples and N_u control samples genotyped at L SNP makers.
SuccessiveRun: Stop condition – If no significant interactions are identified within successive *SuccessiveRun* runs of PathSeeker, then SNPHarvester terminates.
T: statistical significance threshold.
Output:
S: a collection of k-SNP groups which pass the statistical testing, where $k = 1, \ldots, \lceil \ln_3 N_d - 1 \rceil$.

```
for k = 1 to ⌈ln_3 N_d − 1⌉ do
  if k == 1 then
    scan L SNP markers, put the significant SNPs into S, and remove those SNPs
  else
    NumRandomRun ← 0
    while NumRandomRun < SuccessiveRun do
      (M,E) = PathSeeker(D,k,T)
      Remove the local optimum E
      if M is empty then
        NumRandomRun++
      else
        Put M into S, and NumRandomRun ← 0
      end if
    end while
  end if
end for
return S
```

(2) We fit a L_2 penalized logistic regression to minimize

$$L(\beta_0, \boldsymbol{\beta}, \lambda) = -l(\beta_0, \boldsymbol{\beta}) + \frac{\lambda}{2}||\boldsymbol{\beta}||_2^2, \quad (1)$$

where $l(\beta_0, \boldsymbol{\beta})$ is the binomial log-likelihood and λ is a regularization parameter. Here, we exactly follow the method proposed in Park and Hastie (2008):

(a) We run a classical forward-backward variable selection to extract interactions.

(b) We use the Bayesian information criterion (BIC) to measure the model complexity and choose λ by cross-validation.

(3) We report the interactions selected by the L_2 penalized logistic regression as epistatic interactions.

3 RELATIONSHIP BETWEEN SNPHARVESTER AND EXISTING METHODS

Existing interaction identification methods can be roughly divided into three categories:

(1) Exhaustive search methods such as Multifactor-Dimensionality Reduction (MDR) work well on small size problem. In GWA studies, direct application of these methods is computationally prohibitive. An effective filtering method is needed to significantly reduce the number of SNPs so that exhaustive search is computationally feasible on the reduced SNP set.

(2) Greedy search methods: they select the first SNP to best discriminate cases and controls at the first step, and select the second one such that the selected two SNPs maximally reduce the classification error. This process continues until specified model complexity is achieved. For example, CART splits on a SNP which can optimize some criterion (entropy or gini index) for classification, and then performs recursive partition until a large tree is grown. If no marginal effects are present, the choice of split variable is just like a random guess. Stepwise methods (e.g. Marchini *et al*., 2005) also suffer from the same issue, which is confirmed in Zhang and Liu (2007) by simulation studies. Therefore, these methods will fail when marginal effects of disease loci are absent.

(3) Stochastic search methods. BEAM (Zhang and Liu, 2007) designs a Bayesian marker partition model and uses Markov chain Monte Carlo (MCMC) sampling strategy to maximize the posterior probability of the model; grammatical evolution neural networks (Motsinger-Reif *et al*., 2008) use a variation of genetic programming to optimize the structure of neural network and select associated SNPs; random handfuls (Province and Borecki, 2008) randomly select handfuls of SNPs and model their effects by a linear regression model, then update the posterior probability of each SNP based on the linear regression model. Expert knowledge is explored in genetic programming (Moore, 2007; Moore and White, 2006, 2007) and ant colony optimization (Greene *et al*., 2008) to improve the performance of stochastic search methods.

Our SNPHarvester algorithm belongs to the category of stochastic search methods. Despite of the conceptual similarity, we would like to highlight the following key differences:

- Global optima versus local optima. Those existing methods such as genetic programming are interested in global optima. For our SNPHarvester algorithm, we use the PathSeeker to find a local optimal solution instead of global one. In GWA studies, there are usually multiple interaction patterns. Each of them corresponds to a local optimal solution. Every local optimal solution is practically meaningful and should be returned to the users. Hence, we did not employ any control strategies to prevent the algorithm from reaching local optima or use other search algorithms, such as simulated annealing to jump out of local optima. Instead, we use a hill climbing search to reach the local optima as soon as possible.
- Sequential optimization versus parallel optimization. Those existing methods, such as genetic programming and ant colony optimization try to identify multiple interactions in a parallel manner. One distinct feature of SNPHarvester is that it detects significant SNPs in a sequential manner. It removes local optima in the search process. Consequently, search space becomes smaller in the later stage.
- Model-based approaches versus model-free approaches. BEAM builds a simplified model M to explain the whole genome-wide data D and tries to maximize the loglikelihood $\ln p(D|M)$ by MCMC. Grammatical evolution neural networks select associated SNPs based on the network architecture. Random handfuls update the posterior probability of each SNP based on a linear regression model. SNPHarvester does not try to build a model but directly creates paths to detect significant associations. It uses a simple score function (the χ^2-value) to measure the association between SNPs and the phenotype for computational efficiency.

4 RESULTS

In this section, we evaluate the performance of SNPHarvester using both simulated and real data. In simulation studies, we compare SNPHarvester with some recent competitors under a wide spectrum of epistatic models. For the real case–control study, we run SNPHarvester on the rheumatoid arthritis (RA) data (about 500K SNPs, 3504 samples) from the Wellcome Trust Case Control Consortium (WTCCC).

4.1 Experiments on simulation studies

In simulation studies, we mainly compare SNPHarvester with BEAM (Zhang and Liu, 2007) since BEAM is very powerful for detecting epistatic interactions. We also use Random Forest (Breiman, 2001) as a representative of the methods that require marginal effects (the comparison results are given in the Supplementary Material). We simulate 100 datasets for each parameter setting. We use the ratio of the number of successful identifications to the number of datasets to measure the power of each method. We conduct our experiments in three cases:

- Case 1: disease loci with marginal effects.
- Case 2: disease loci without marginal effects.
- Case 3: multiple epistatic interactions.

The methods requiring marginal effects are expected to perform reasonably well in Case 1, while they would perform poorly in Case 2. Case 3 is designed to mimic multiple causal epistasis.

4.1.1 Case 1: disease loci with marginal effects There are many interaction models with weak marginal effects. We use the three models in Marchini *et al*. (2005) for comparison. Model 1 is an additive model, and Models 2 and 3 define epistatic interactions with multiplicative effects and threshold effects, respectively. The marginal effects are measured in effect size λ as defined in Marchini *et al*. (2005), and linkage disequilibrium (LD) between SNPs is measured by r^2. Because Zhang and Liu (2007) has carried out comprehensive comparison studies of BEAM, the stepwise logistic regression, logic regression and MDR, here we only show the comparison between SNPHarvester and BEAM. More results can be found in the Supplementary Material.

We simulate data based on the three models under different parameter settings to study the power of SNPHarvester. These settings are designed for the practical concerns as discussed in Wang *et al*. (2005):

- Sample size is very important for case–control studies. The statistical power is often increased by increasing sample size. Under each parameter setting, 2000 samples ($N_d = 1000, N_u = 1000$) and 4000 samples ($N_d = 2000, N_u = 2000$) are simulated.
- The statistical power is influenced by minor allele frequency (MAF) greatly. Here MAF is chosen from 0.1 to 0.5.
- Effective size λ is chosen to be relatively small: $\lambda = 0.3$ for Model 1 and $\lambda = 0.2$ for Models 2 and 3.
- Disease loci may or may not be genotyped in reality. The cases $r^2 = 1$ are simulated for the disease loci directly genotyped, and the cases $r^2 = 0.7$ are simulated for the disease loci ungenotyped but their LD markers with $r^2 = 0.7$ genotyped.
- In each setting, 2000 SNP markers are simulated.

The results in Figure 4 show that:

(1) SNPHarvester performs slightly better than BEAM when disease loci present marginal effects. Random Forest is comparable with SNPHarvester and BEAM (see the Supplementary Material).
(2) The power of both methods can be increased by increasing the sample size.
(3) If the disease locus is unobserved, then it becomes more difficult to identify the locus by the LD markers (the performances of both methods are worse in cases $r^2 = 0.7$ than that in cases $r^2 = 1.0$).

4.1.2 Case 2: disease loci without marginal effects A wide spectrum of interaction models without marginal effects have been discussed in Culverhouse *et al*. (2002). Here, we consider the 60 pure epistatic models in Velez *et al*. (2007) to compare the performance between SNPHarvester and BEAM. The details of these models are available in the Supplementary Material. The heritability h^2 [see definition in Culverhouse *et al*. (2002)] of these 60 models ranges from 0.025 to 0.4, and the MAF ranges from 0.2 to 0.4. We use 100 datasets for each disease model. There are 200 cases, 200 controls and 1000 SNPs in each dataset.

Random Forests work poorly when no marginal effects are present (please refer to the Supplementary Material). The comparison between SNPHarvester and BEAM in Figure 5 shows that SNPHarvester is superior to BEAM for detecting epistatic interactions without marginal effects. In our experiments, we only allow SNPHarvester to generate 50 paths to save computation time. But this small number already enables SNPHarvester to outperform BEAM. For the models with MAF $= 0.2, 0.4$ and $h^2 \geq 0.1$, the power of SNPHarvester is about 70%, while that of BEAM is roughly 20%. The performances of the two methods degrade as the heritability h^2 decreases: BEAM almost totally loses its power for the models with MAF $= 0.2$ and $h^2 \leq 0.05$, while SNPHarvester still keeps its power at about 40% for some of these models and does better for the models with MAF $= 0.4$. We also explore why SNPHarvester degrades its performance as the heritability h^2 decreases. The reason is that the χ^2-value of the two disease loci is no longer significant compared with random match of any two loci.

4.1.3 Case 3: multiple disease loci without marginal effects In practice, there might exist multiple SNP–SNP interactions in the association studies. We use eight hybrid models (HM) to mimic multiple interactions, and compare SNPHarvester and BEAM on these models. Each of the eight HMs is constructed by a mixture of five pure epistatic models but with the same heritability and MAF (details of the eight HMs are given in the Supplementary Material). For example, HM1 consists of Model epi1 $\sim$ epi5. We simulate the first interaction based on Model epi1, and the second interaction based on Model epi2, and so on. Thus, there are five interactions in the HM but they are simulated independently. We simulated 100 datasets for each HM and each dataset contains 200 cases, 200 controls and 1000 SNPs.

The comparison between SNPHarvester and BEAM is shown in Figure 6. BEAM only identifies one of five interactions in most cases and can identify at most two of five interactions simultaneously, while SNPHarvester often identifies more than

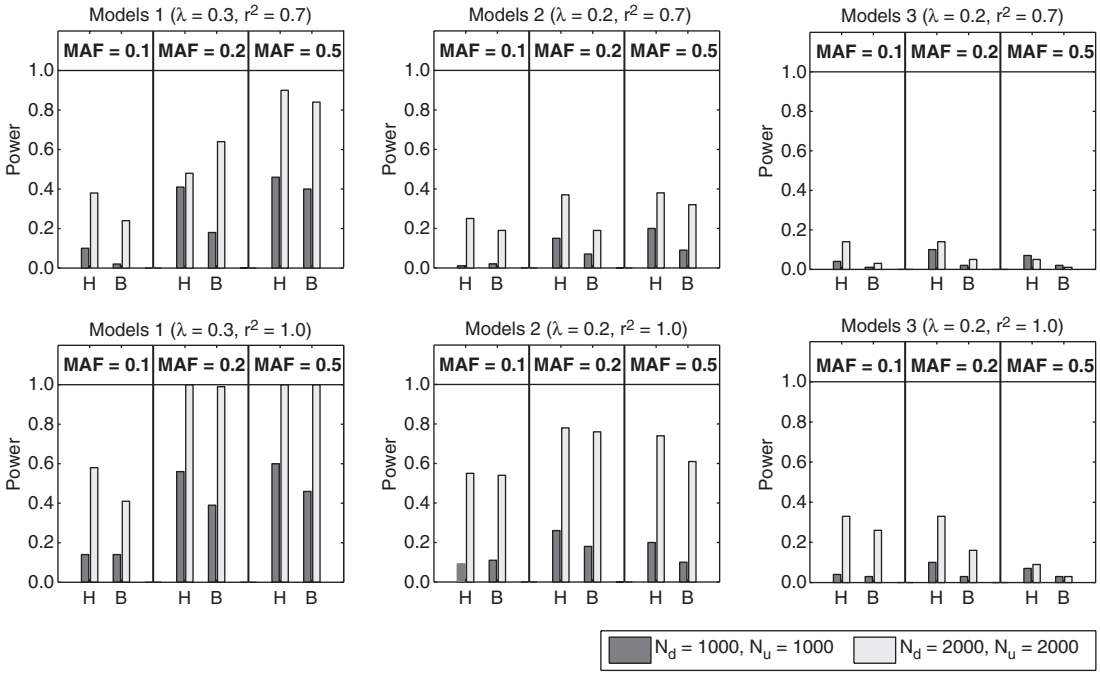

Fig. 4. The performance comparison between SNPHarvester (H) and BEAM (B) on three models with marginal effects. For each model 100 datasets are generated. Under each parameter setting, 2000 samples ($N_u = 1000, N_d = 1000$) and 4000 samples ($N_u = 2000, N_d = 2000$) are simulated. SNPHarvester generates 50 paths (roughly 2.5×10^5 operations) and BEAM runs 5×10^6 MCMC iterations. The comparison shows that SNPHarvester outperforms BEAM slightly on the three models with marginal effects.

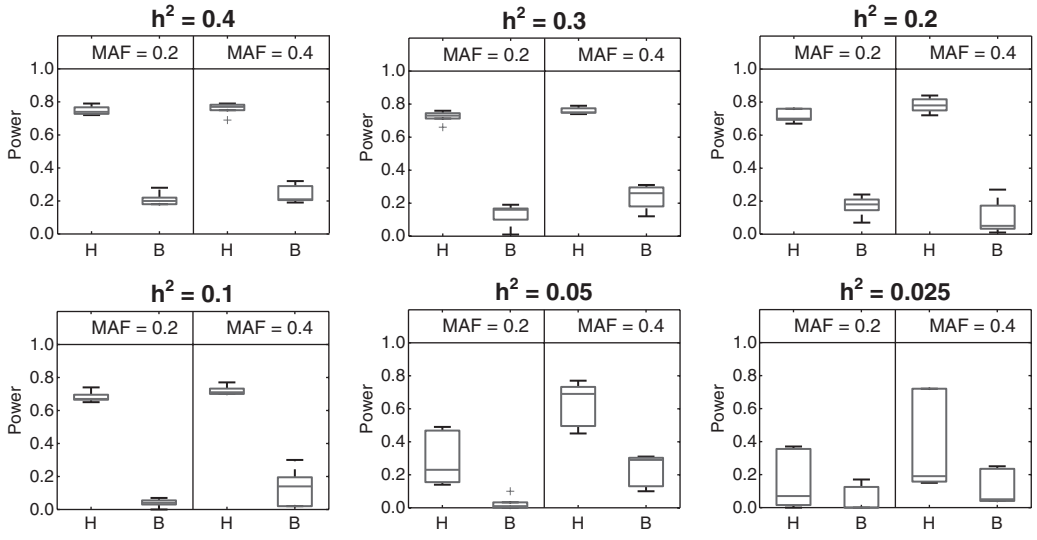

Fig. 5. The performance comparison between SNPHarvester (H) and BEAM (B) on 60 pure epistatic models (without marginal effects). For each model 100 datasets are generated. Each dataset contains 400 samples ($N_u = 200$, $N_d = 200$) and 1000 SNPs. SNPHarvester generates 50 paths (roughly 2.5×10^5) and BEAM runs 5×10^6 MCMC iterations. The comparison shows that SNPHarvester outperforms BEAM on the 60 pure epistatic models.

two interactions simultaneously. Clearly, SNPHarvester outperforms BEAM significantly in terms of identifying multiple interactions.

4.2 Parameter setting

The issue of how many paths should be generated is not resolved theoretically. Instead, we conduct experiments to show the performance of SNPHarvester under various parameter setting. Here, we consider the 10 pure epistatic models (model epi6 ∼ model epi10 and model epi16 ∼ model epi20 given in the Supplementary Material). We use 100 datasets for each model and each dataset contains $L = 1000$ SNPs. The result is shown in Figure 7.

Figure 7 shows that the power increases as the parameter *SuccessiveRun* increases. We also record the computation time under

different value of *SuccessiveRun* and different number of samples N on a PC (CPU: Intel 3.0GHz and RAM 8GB). Figure 7 shows that the computation time of SNPHarvester increases linearly with respect to N.

To obtain a good compromise between the power of SNPHarvester and its computational efficiency, we suggest setting $SuccessiveRun = 40 \sim 50$ for medium-scale problems (i.e. 10^4 SNPs) and $SuccessiveRun = 20 \sim 30$ for large-scale problems (i.e. 10^5 SNPs).

4.3 Experiments on WTCCC RA study

We use SNPHarvester to perform GWA study on WTCCC RA data. Most SNP markers identified by WTCCC are also identified by SNPHarvester, since they shows remarkable marginal effects. For example, the SNP markers rs582757, rs5029938 and rs5029939 for the gene TNFAIP3 which are shown to be closely related to RA (Thomson *et al*., 2007; WTCCC, 2007) are identified. We report some SNP interactions in Table 1.

- An association between RA and gene HLA-DRB1 (locating at 6p21.3) has been established in Gregersen *et al*. (1987). Our analysis indicates the strong association between RA and 6p21.3. The top 10 significant SNP groups in that region are reported in Table 1 (more details can be found in the Supplementary Material).
- SNP markers rs1358169, rs6460831 and rs2526100 are related to gene THSD7A on chromosome 7, which have been reported to be associated with bone mineral density recently (Mori *et al*., 2008). This shows plausible biological relevance.
- We also report some SNP groups which show weak marginal effects but strong interactions (see the Supplementary Material). Their biological interpretation needs to be further investigated.

Regarding to the computation time, it takes about 2 weeks for SNPHarvester to handle WTCCC data on a PC (CPU: Intel 3.0GHz and RAM 8GB). For the results shown in this article,

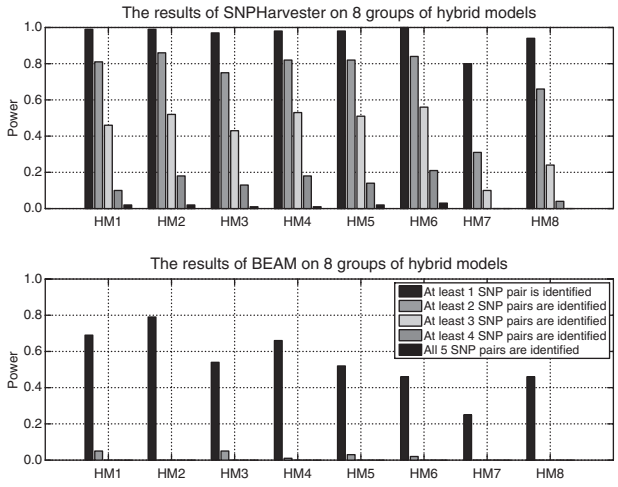

Fig. 6. The performance comparison between SNPHarvester (H) and BEAM (B) on eight HMs. 'HMi' represents 'Hybrid Model i', where $i = 1, \ldots, 8$. The absence of bar indicates zero power. For each model 100 datasets are generated. Each dataset contains 400 samples ($N_d = 200, N_u = 200$) and 1000 SNPs. SNPHarvester generates 50 paths and BEAM runs 5×10^6 MCMC iterations. The comparison shows that SNPHarvester outperforms BEAM in terms of identifying multiple causal epistatic interactions on the eight hybrid epistatic models.

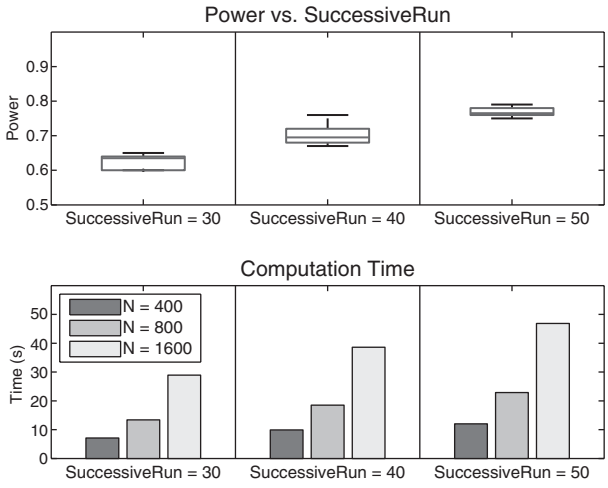

Fig. 7. The power and computation time of SNPHarvester when *SuccessiveRun* varies from 30 to 50.

Table 1. Some significant SNP groups identified by SNPHarvester on WTCCC RA data

SNP groups	Location	Related genes	P-value
(rs2621419,rs2857130)	6p21.3-p21.3	HLA-DQA2, HLA-DQB2	2.0849×10^{-27}
(rs2621419,rs2857154)	6p21.3-p21.3	HLA-DQB1, HLA-DQB2	1.3548×10^{-22}
(rs910050,rs9268877)	6p21.3-p21.3	HLA-DQB1, HLA-DQB2	1.4266×10^{-22}
(rs2621384,rs2857154)	6p21.3-p21.3	HLA-DQB1, HLA-DQB2	1.6249×10^{-22}
(rs2857173,rs2857154)	6p21.3-p21.3	HLA-DQB1, HLA-DQB2	3.4497×10^{-22}
(rs3135342,rs9268877)	6p21.3-p21.3	BTNL2, HLA-DRB9	4.1657×10^{-22}
(rs5000563,rs9268877)	6p21.3-p21.3	HLA-DRA, HLA-DRB9	4.4302×10^{-22}
(rs3129877,rs9268877)	6p21.3-p21.3	HLA-DRA, HLA-DRB9	1.1479×10^{-21}
(rs2857173,rs7382347)	6p21.3-p21.3	HLA-DQB1, HLA-DQB 2	1.2622×10^{-21}
(rs1358169,rs6460831)	7p21.3-7p21.3	(THSD7A,THSD7A)	$<10^{-30}$
(rs2526100,rs6460831)	7p21.3-7p21.3	(THSD7A,THSD7A)	$<10^{-30}$

we set *SuccessiveRun* = 20 for WTCCC data analysis. The available software of BEAM fails to handle the whole genome-wide data.

5 EXTENSION TO INCORPORATING EXPERT KNOWLEDGE

Despite of the success of SNPHarvester, we realize that its performance is inferior to the methods by exploring expert knowledge (Moore, 2007; Moore and White, 2006, 2007). We believe that expert knowledge is critical for the greater success of genetic analysis. Expert knowledge may come from biological information, e.g. pathway information (Wang *et al*., 2007), or some other computational resource, e.g. Tuned ReliefF (Moore and White, 2007). Assuming some 'good' SNPs have been given by expert knowledge, below are some possible ways to extend SNPHarvester by incorporating expert knowledge:

(1) Initial points of paths should be guided by expert knowledge. 'Good' SNPs should be selected as initial points with higher probability.

(2) Updating active set can also be guided by expert knowledge. 'Good' SNPs can take precedence over other SNPs.

(3) If pathway information is available (Wang *et al*., 2007), SNPHarvester could spend more computation time within the same pathway than across pathways. This will help to identify biological meaningful epistatic interactions.

We plan to explore this issue in the future to improve our current method.

6 CONCLUSION

In this article, we proposed a simple but effective method named SNPHarvester for GWA studies. SNPHarvester efficiently reduces the number of SNPs and enables the direct applications of existing statistical tools in interaction detection. We show that SNPHarvester outperforms its nearest competitors in both extensive simulation studies and real application.

Funding: This work was supported with the Grant GRF621707, from the Hong Kong Research Grant Council, grant RPC07/08.EG25, RPC06/07.EG09 and a postdoctoral fellowship award from the Hong Kong University of Science and Technology.

Conflict of Interest: none declared.

REFERENCES

Breiman,L. *et al*. (2001) Random forests. *Mach. Learn.*, **45**, 5–32.

Cho,Y. *et al*. (2004) Multifactor-dimensionality reduction shows a two-locus interaction associated with type 2 diabetes mellitus. *Diabetologia*, **47**, 549–554.

Cordell,H.J. (2002) Epistasis: what it means, what it doesn't mean, and statistical methods to detect it in humans. *Hum. Mol. Genet.*, **11**, 2463–2468.

Culverhouse,R. *et al*. (2002) A perspective on epistasis: limits of models displaying no main effect. *Am. J. Hum. Genet.*, **70**, 461–471.

Greene,C.S. *et al*. (2008) Ant colony optimization for genome-wide genetic analysis. In Dorigo,M. *et al*. (eds), *Procedings of the 6th International Conference on Ant Colony Optimization and Swarm Intelligence (ANTS 2008)*, vol. 5217 of *Lecture Notes in Computer Science*. Springer, Brussels, Belgium, pp. 37–47.

Gregersen,P.K. *et al*. (1987) The shared epitope hypothesis. an approach to understanding the molecular genetics of susceptibility to rheumatoid arthritis. *Arthritis Rheum.*, **30**, 1205–1213.

Griffiths,A.J. *et al*. (2008) *Introduction to Genetic Analysis*. W.H.Freeman and Co Ltd., New York, USA.

Hirschhorn,J.N. and Daly,M.J. (2005) Genome-wide association studies for common diseases and complex traits. *Nat. Rev. Genet.*, **6**, 95–108.

Liang,Y. and Kelemen,A. (2008) Statistical advances and challenges for analyzing correlated high dimensional SNP data in genomic study for complex diseases. *Stat. Surv.*, **2**, 43–60.

Marchini,J. *et al*. (2005) Genome-wide strategies for detecting multiple loci that influence complex diseases. *Nat. Genet.*, **37**, 413–417.

McCarthy,M. *et al*. (2008) Genome-wide association studies for complex traits: consensus, uncertainty and challenges. *Nat. Rev.*, **9**, 356–369.

Moore,J.H. (2007) Genome-wide analysis of epistasis using multifactor dimensionality reduction: feature selection and construction in the domain of human genetics. In Zhu,X. and Davidson,I. (eds), *Knowledge Discovery and Data Mining: Challenges and Realities with Real World Data*. IGI Global, Hershey, USA, pp. 17–30.

Moore,J.H. and White,B.C. (2006) Exploiting expert knowledge in genetic programming for genome-wide genetic analysis. In Runarsson,T.P. *et al.* (eds), *Procedings of the 9th International Conference on Parallel Problem Solving from Nature*, vol. 4193 of *Lecture Notes in Computer Science*. Springer, Reykjavik, Iceland, pp. 969–977.

Moore,J.H. and White,B.C. (2007) Genome-wide genetic analysis using genetic programming: the critical need for expert knowledge. In Riolo,R. *et al.* (eds), *Genetic Programming Theory and Practice IV*. Springer, New York, USA, pp. 11–28.

Mori,S. *et al.* (2008) Association of genetic variations of genes encoding thrombospondin, type 1, domain-containing 4 and 7a with low bone mineral density in Japanese women with osteoporosis. *J. Hum. Genet.*, **53**, 694–697.

Motsinger-Reif,A.A. *et al.* (2008) Power of grammatical evolution neural networks to detect gene-gene interactions in the presence of error. *BMC Res. Notes*, **1**, 65.

Musani,S. *et al*. (2007) Detection of gene-gene interactions in genome-wide association studies of human population data. *Hum. Hered.*, **63**, 67–84.

Park,M. and Hastie,T. (2008) Penalized logistic regression for detecting gene interactions. *Biostatistics*, **9**, 30–50.

Province,M.A. and Borecki,I.B. (2008) Gathering the gold dust: methods for assessing the aggregate impact of small effect genes in genomic scans. In *Proceedings of Pacific Symposium on Biocomputing*, Maui, Hawaii, USA.

Ritchie,M. *et al*. (2001) Multifactor-dimensionality reduction reveals high-order interactions among estrogen-metabolism genes in sporadic breast cancer. *Am. J. Hum. Genet.*, **69**, 138–147.

Thomson,W. *et al*. (2007) Rheumatoid arthritis association at 6q23. *Nat. Genet.*, **39**, 1431–1433.

Velez,D. *et al*. (2007) A balanced accuracy function for epistasis modeling in imbalanced datasets using multifactor dimensionality reduction. *Genet. Epidemiol.*, **31**, 306–315.

Wang,K. *et al*. (2007) Pathway-based approaches for analysis of genomewide association studies. *Am. J. Hum. Genet.*, **81**, 1278–1283.

Wang,W. *et al*. (2005) Genome-wide association studies: theoretical and practical concerns. *Nat. Rev. Genet.*, **6**, 109–118.

WTCCC (2007) Genome-wide association study of 14,000 cases of seven common diseases and 3,000 shared controls. *Nature*, **447**, 661–678.

Zhang,Y. and Liu,J.S. (2007) Bayesian inference of epistatic interactions in case-control studies. *Nat. Genet.*, **39**, 1167–1173.