ParaHaploJava Manual

A program for whole genome association study using parallel computing
Version 1.1

2009/04/30

Kazuharu Misawa [a] and Naoyuki Kamatani [b]

[a] Research Program for Computational Science, Research and Development Group for Next-Generation Integrated Living Matter Simulation, Fusion of Data and Analysis Research and Development Team, RIKEN, 4-6-1 Shirokane-dai, Minato-ku, Tokyo 108-8639, Japan. [b] Laboratory for Statistical Analysis, RIKEN Center for Genomic Medicine, Tokyo, Japan

Table of Contents

Introduction

The purpose of this program package

Recent advances in high-throughput genotyping technologies have allowed us to test allele frequency differences between case and control populations on a genome-wide scale (Hirschhorn and Daly 2005). More than one million single nucleotide polymorphisms (SNPs) for which accurate and complete genotypes have been obtained Thousands of people are now being genotyped (Nakamura 2007; Yamaguchi-Kabata et al. 2008).

One of the crucial problems in GWAS is correction for multiple comparisons. Usually Bonferroni's correction for the P value is used to account for multiple testing. When SNP loci are in linkage disequilibrium, however, Bonferroni's correction is known to be too conservative and may drop truly significant SNPs (Kimmel and Shamir 2006; Misawa et al. 2008).

To cope with multiple comparison problem in GWAS, Misawa et al. (2008) have developed new algorithms to correct for the multiple comparisons at multiple SNP loci in linkage disequilibrium by treating linked loci as one haplotype block. They developed the method to calculate the exact probability of the type I error under the condition that the haplotype frequencies in the population are known and the number of haplotype copies in sample follows a multinomial distribution. The permutation test also can handle this problem (Kimmel and Shamir 2006).

Running time is a problem in calculating the exact probability (Misawa et al. 2008) as well as in performing permutation tests (Kimmel and Shamir 2006). In this study, we developed ParaHaplo (Misawa and Kamatani submitted) , parallel computation programs for calculating exact probability (Misawa et al. 2008) as well as for permutation tests (Kimmel and Shamir 2006) for GWAS. ParaHaplo is based on data parallelism, a programming technique for splitting a large data set into small data sets that can be operated on in parallel (Culler, Gupta, and Singh 1997) p44. ParaHaplo is developed based on the Intel Message Passing Interface (MPI) and runs on PC clusters.

To measure the efficiency of ParaHaplo in computational time, difference in haplotype frequency between CHB and JPT hapmap (The_International_HapMap_Consortium 2005) on chromosome 22 is analyzed by using the new program. The result showed that more than 100 times speed up was achieved by ParaHaplo.

Program overivew

ParaHaplo tests the difference in allele frequency between case and control as well as that between two populations. ParaHaplo outputs the Pearson score for chi-square test. The users can make ParaHaplo output Fst by using command line option. ParaHaplo calculates the rates of type I errors of the test on the allele frequency by the exact method (Misawa et al. 2008). The algorithm to calculate asymptotically the type I error rates using a Markov-chain Monte Carlo sampler (Misawa et al. 2008) is also implemented in this program. This program also supports The standard permutation test (SPT) and RAT (Kimmel and Shamir 2006) .

ParaHaplo is implemented in a MPI-C multithreaded package. MPI package allows us to construct parallel computing programs on multiprocessors. The genome-wide polymorphism data is broken into haplotype blocks defined by users. Then, the MPI-Bcast function is used to distribute a single set of haplotype block data into each processor. Then haplotype frequency data of one haplotype block are analyzed by a single-processor. In this step, the probability of local type I error given the significance level at each SNP locus is calculated.

After the analysis on each haplotype block is complete, the results are joined into a single genome-wide data by using the MPI-Gatherv function. Then, the global type I error is obtained from the local type I error by using Bonferroni's correction, because different haplotype blocks are considered to be independent of each other although SNPs within haplotype block are not independent.

ParaHaplo requires an input file of haplotype block boundary, and two data for population data. When data files are provided for each chromosome, ParaHaplo HapMap data format and BioBank Japan data format are supported. ParaHaplo is compatible to OpenMPI version 1.2.5 as well as to MPICH version 1.2.7p1.The users can compile the source by GCC compiler as well as by Intel C compiler. C programs as well as Java programs are also available for single-processor machines.This program package contains the following programs:

(i) A Markov-chain Monte Carlo (MCMC) algorithm to calculate asymptotically the probability of the type I error (Misawa et al. 2008)

(ii) Calculation of exact type I error rates (Misawa et al. 2008)

(iii) Permutation test based on RAT algorithm (Kimmel and Shamir 2006)

(iv) Standard Permutation test (SPT)

(v) Data conversion from BioBank Japan format to HapMap format

Operating systems and platforms

We have tested our java program on Java 1.5 or later.

Install

Download and install

The source programs and executable binaries are available at
http://sourceforge.jp/projects/parallelgwas/?_sl=1

Then, put allTest.jar on your folder.

Usage of programs in the packages

How to run the non-parallel version in Java

Usage:

```
%> jar –jar allTests.jar $1 $2 $3 $4 $5 $6 $7
```

Command-line options

$1 :	The first data file, as the case population
$2 :	The second data file, as the control population
$3:	The start position of a haplotype block
$4 :	The end position of a haplotype block
$5:	Number of burnin generations
$6 :	Number of MCMC generations
$7 :	Name of Method, MCMC, RAT or SPT

MCMC: Calculation of type I error rates by using MCMC method

RAT: Calculation of P value by RAT algorithm

SPT: Calculation of P value by standard permutaion algorithm

Data formats of input and output files

ParaHaplo supports data format of the HapMap (The_International_HapMap_Consortium 2005) and that of BioBank Japan (Nakamura 2007). The example data are included in SNP/data.

.

Data conversion from BioBank Japan format to HapMap format

To convert genotype file in format of BioBank Japan to HapMap format, use illumina2hapmap.exe.

Usage:

```
%> illumina2hapmap.exe $1 $2 $3
```

illumina2hapmap.exe is in SNP/tools/illumina2hapmap directory. When the user run this script outside of this directory, the path must be set appropriately.

Command-line options

$1:	Data file in BioBank Japan style.
$2:	Information file of SNPs
$3:	Output data file in HapMap format

Haplotype block file

In this program package, users can specify the haplotype-block file; Haplotype blocks defined by user or sliding window method.

Haplotype blocks defined by user

When haplotype blocks defined by user are used, users must list the boundaries of haplotype blocks in the haplotype-block file. The last line must be empty. When this option is used, haplotype blocks are not overlapping. The following is an example of the haplotype-block file.

```
14000000
15000000
16000000
17000000

48000000
49000000
50000000
```

Sliding window method

When sliding window method is used, users must write the window size and the step size in the haplotype-block file. The first line of the haplotype block file must be the window size, and the second line must be the step size. Both of the window size and the step size are measured by the number of SNPs. When the step size is smaller than the window size, each window is overlapping. When the step size is larger than the window size, SNPs between windows are ignored. If each SNP is need to be analyzed separately, Both of window size and step size must be set to 1. The third line must be empty. The following is an example of the haplotype-block file.

```
 100
 200
```

Output file

The programs within the program package output the result of analyses in the similar format. The output files consist of two parts; the header part and the result part. The difference among the output files of different programs is in the header part.

The header part

The first and second lines show the names of case and control data file. The first column shows the range of the haplotype block on the chromosomes. The second column shows the number of SNPs in the haplotype block. The third line shows the number of runs of the MCMC chains. The fourth line shows the number of generations of the MCMC chains. The fifth line is the headline. All lines after the headline are result lines.

The result part

The result lines show the result of the analyses. The third column and the fourth column show the rs number and the position on the chromosome of the SNP that has the highest score, respectively. The fifth column shows the highest score. The sixth score shows the P value. When there is no polymorphic site in the haplotype block, the program outputs “NoData.”.

The first column shows the range of the haplotype block on the chromosomes. The second column shows the number of SNPs in the haplotype block. The third line shows the number of runs of the MCMC chains. The fourth line shows the number of generations of the MCMC chains. The fifth line is the headline. All lines after the headline are result lines.

```
CaseData      = /phased/JPT/chr22.txt
ControlData   = /phased/CHB/chr22.txt
Repeat        = 4
Generation    = 100000
BlockArea        SNPNum  rsNumber Position  Score         Type I error
14657412-14893245  100  rsxxxxx1  14870204  7.9158529679  0.0161200000
14976512-16148952  100  rsxxxxx2  15853229  10.0341711103 0.4292800000
16485214-17065485  100  rsxxxxx3  16643268  16.3205869262 0.0189700000
17165489-18221578  100  NoData
18345862-19147853  100  rsxxxxx5  18582729  28.5802577195 0.0000000000
19348562-19965482  100  rsxxxxx6  19660266  14.2802131465 0.0318300000
```

Calculating the global P value

Usually the data files for case and control populations are separated into chromosomes. Thus, the users must calculate the global P value by gathering the results from all chromosomes. To calculate the global P value, use global.awk. global.awk is in SNP/Tools.

Usage:

```
%> gawk –f global.awk datafile1.txt datafile2.txt … >all.txt
```

Literature cited

Culler DE, Gupta A, Singh JP. 1997. Parallel Computer Architecture: A Hardware/Software Approach. Morgan Kaufmann Publishers, San Francisco, CA.

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

Kimmel G, Shamir R. 2006. A fast method for computing high-significance disease association in large population-based studies. Am J Hum Genet **79**:481-492.

Misawa K, Fujii S, Yamazaki T, Takahashi A, Takasaki J, Yanagisawa M, Ohnishi Y, Nakamura Y, Kamatani N. 2008. New correction algorithms for multiple comparisons in case-control multilocus association studies based on haplotypes and diplotype configurations. J Hum Genet **53**:789-801.

Misawa K, Kamatani N. submitted. ParaHaplo: A program package for whole-genome association study using parallel computing J. Comp. BIol.

Nakamura Y. 2007. The BioBank Japan Project. Clin Adv Hematol Oncol **5**:696-697.

The_International_HapMap_Consortium. 2005. A haplotype map of the human genome. Nature **437**:1299-1320.

Yamaguchi-Kabata Y, Nakazono K, Takahashi A, Saito S, Hosono N, Kubo M, Nakamura Y, Kamatani N. 2008. Japanese population structure, based on SNP genotypes from 7003 individuals compared to other ethnic groups: effects on population-based association studies. Am J Hum Genet **83**:445-456.