

Research

Open Access

A fast algorithm for genome-wide haplotype pattern mining

Søren Besenbacher*^{1,2}, Christian NS Pedersen^{1,2} and Thomas Mailund¹

Address: ¹Bioinformatics Research Center, University of Aarhus, Denmark and ²Department of Computer Science, University of Aarhus, Denmark

Email: Søren Besenbacher* - besen@birc.au.dk; Christian NS Pedersen - cstorm@birc.au.dk; Thomas Mailund - mailund@birc.au.dk

* Corresponding author

from The Seventh Asia Pacific Bioinformatics Conference (APBC 2009)
Beijing, China. 13–16 January 2009

Published: 30 January 2009

BMC Bioinformatics 2009, **10**(Suppl 1):S74 doi:10.1186/1471-2105-10-S1-S74

This article is available from: <http://www.biomedcentral.com/1471-2105/10/S1/S74>

© 2009 Besenbacher et al; licensee BioMed Central Ltd.

This is an open access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/2.0>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Abstract

Background: Identifying the genetic components of common diseases has long been an important area of research. Recently, genotyping technology has reached the level where it is cost effective to genotype single nucleotide polymorphism (SNP) markers covering the entire genome, in thousands of individuals, and analyse such data for markers associated with a diseases. The statistical power to detect association, however, is limited when markers are analysed one at a time. This can be alleviated by considering multiple markers simultaneously. The *Haplotype Pattern Mining* (HPM) method is a machine learning approach to do exactly this.

Results: We present a new, faster algorithm for the HPM method. The new approach use patterns of haplotype diversity in the genome: locally in the genome, the number of observed haplotypes is much smaller than the total number of possible haplotypes. We show that the new approach speeds up the HPM method with a factor of 2 on a genome-wide dataset with 5009 individuals typed in 491208 markers using default parameters and more if the pattern length is increased.

Conclusion: The new algorithm speeds up the HPM method and we show that it is feasible to apply HPM to whole genome association mapping with thousands of individuals and hundreds of thousands of markers.

Background

Identifying the genetic causes of common diseases has long been an important research area in genetics. Where early studies were limited to studying few genes at a time, due to economical and technological constraints, development in genotyping technology has revolutionised the field. It is now cost effective to obtain hundreds of thousands of genotype markers in thousands of individuals for a single study. This makes it possible to scan the entire genome for disease associated markers in a single analysis

and such genome-wide association studies have recently lead to a virtual flood of newly discovered disease genes [1-7].

Most studies search for disease association through a marker-by-marker approach where each marker in turn is tested for association to the disease phenotype, e.g. using a simple Fisher's exact test or a χ^2 -test. However, a marker by marker approach is limited in statistical power due to the indirect testing for association, where so called "tag

SNPs" are used as proxies for unobserved markers, but by using multiple markers, this problem can be alleviated [8,9]. A tradeoff must be made between method sophistication and computation efficiency when developing multi-marker approaches, however.

The Haplotype Pattern Mining method is a multi-marker approach introduced in 2000 by Toivonen *et al.* [10,11]. It is based on the idea of extracting local haplotype similarities and locating areas where haplotypes are correlated with the disease phenotype. Compared to methods based on statistical sampling [12-17] HPM is computationally much more efficient, similar to other heuristic approaches [18-20] capable of analysing genome-wide datasets. In this paper, we develop a faster version of HPM and show that it scales to genome-wide association studies.

Methods

The goal of association mapping is to find disease-predisposing regions of the genome. This can be done by looking for differences in the frequency of genetic variants between cases and controls. Since genome sequencing is expensive the whole genomes of the case and control individuals in a case-control study are usually not sequenced. Instead only single base pairs that are known to frequently differ between humans, called SNP markers, are sequenced.

The association mapping problem

If k SNP markers are typed then we can represent a chromosome by a haplotype vector H of length k , where $H = (h_1, \dots, h_k)$ and $h_i \in \text{alleles}(i)$ for all $i, 1 \leq i \leq k$; $\text{alleles}(i)$ is the domain of the i th marker. The input to an association mapping method then consists of a set $A = \{A_1, \dots, A_p\}$ of disease-associated haplotypes and a set $C = \{C_1 \dots C_q\}$ of control haplotypes.

Haplotype pattern

A haplotype pattern P over k markers is a vector $(p_1 \dots p_k)$, where $p_i \in \text{alleles}(i) \cup \{*\}$ for all $i, 1 \leq i \leq k$, where $*$ is the "don't care" symbol. The haplotype pattern occurs in a given haplotype vector (chromosome) $H = (h_1, \dots, h_k)$ if either $p_i = h_i$ or $p_i = *$ for all $i, 1 \leq i \leq k$. The length of a pattern is defined as the maximum distance between two non-"*" characters in the pattern. Gaps are subsequences of "don't care" symbols in a pattern that are surrounded by non-"*" characters on both sides. Since long patterns are not likely to exist we only want look at a subset of the possible patterns. We call the patterns that we want to consider for legal patterns. A pattern is legal if the pattern length is less than the parameter l , it contains fewer than g gaps, and no gaps are longer than s .

Strongly associated pattern

The signed χ^2 measure $\pm \chi^2(P)$ of a haplotype pattern P is the standard χ^2 measure where the sign is positive if the relative frequency of P is higher in cases than in controls, and negative otherwise. Given a positive association threshold x , we say that P is strongly associated with the disease if $\pm \chi^2(P) \geq x$.

The HPM problem

Given a set of case haplotypes $A = \{A_1, \dots, A_p\}$ and control haplotypes $C = \{C_1 \dots C_q\}$ the goal of the HPM algorithm is to find all strongly associated patterns that are legal.

Localizing disease genes using HPM

Haplotype patterns close to a susceptibility locus are likely to be more associated with the disease than patterns further away. If we have found all strongly associated patterns we can score each marker by counting the number of times that it is contained in a strongly associated pattern. The HPM point prediction is then the marker that is most frequently contained in the strongly associated patterns. Fig. 1 shows an example of the localization of a validated susceptibility allele in the Crohn's disease data set from

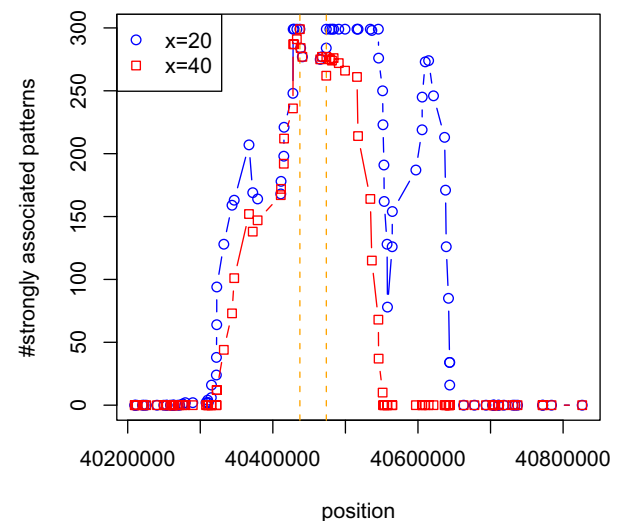


Figure 1
Example of localization. Example of localization of susceptibility alleles using HPM. The plots show the number of strongly associated patterns each marker was included in for two different values of x . The rest of the parameters were fixed at their default values ($l = 7, g = 2, s = 2$). The two vertical lines show the location of the two SNPs in the region that has been validated through replication.

Input: $\pi \times k$ matrix M where $M[i][j]$ = value of chromosome i at marker j , phenotype vector Φ , x , l , g

Output: \mathcal{P}

```

1:  $\mathcal{P} = \emptyset$ 
2:  $\pi_A$  = number of case chromosomes
3:  $\pi_C$  = number of control chromosomes
4:  $lb = \pi_A \cdot \pi \cdot x / (\pi_C \cdot \pi + \pi_A \cdot x)$ 
5:  $p = (p[1], \dots, p[k]) = (*, \dots, *)$ 
6: for  $i = 1$  to  $k$  do
7:   for all  $a \in \text{alleles}(i)$  do
8:      $p[i] = a$ 
9:      $S = \{s \mid M[s][i] == a\}$ 
10:     $\pi_{AP}$  = number of case chromosomes in  $S$ 
11:     $\pi_{CP}$  = number of control chromosomes in  $S$ 
12:    depthFirst( $p, i, i, 0, 0, \pi_{AP}, \pi_{CP}, S$ )
13:     $p[i] = *$ 
14:   end for
15: end for

16: Procedure depthFirst( $p, start, i, nrOfGaps, gapLength, \pi_{AP}, \pi_{CP}, S$ )
17: if  $\pm\chi^2(\pi_{AP}, \pi_{CP}, \pi_A - \pi_{AP}, \pi_C - \pi_{CP}) \geq x$  and  $p[i] \neq *$  then
18:    $\mathcal{P} = \mathcal{P} \cup \{p\}$ 
19: end if
20: if  $i = k$  or  $i + 1 - start > l$  then
21:   return
22: end if
23: if  $\pi_{AP} < lb$  then
24:   return
25: end if
26: for all  $a \in \text{alleles}(i + 1)$  do
27:    $p[i] = a$ 
28:    $S' = \{s \in S \mid M[s][i] == a\}$ 
29:    $\pi'_{AP}$  = number of case chromosomes in  $S'$ 
30:    $\pi'_{CP}$  = number of control chromosomes in  $S'$ 
31:   depthFirst( $p, start, i + 1, nrOfGaps, 0, \pi'_{AP}, \pi'_{CP}, S'$ )
32: end for
33: if  $p[i] \neq *$  and  $nrOfGaps < g$  and  $s \geq 1$  then
34:    $p[i + 1] = *$ 
35:   depthFirst( $p, start, i + 1, nrOfGaps + 1, 1, \pi_{AP}, \pi_{CP}, S$ )
36: end if
37: if  $p[i] = *$  and  $gapLength < s$  then
38:    $p[i + 1] = *$ 
39:   depthFirst( $p, start, i + 1, nrOfGaps, gapLength + 1, \pi_{AP}, \pi_{CP}, S$ )
40: end if
41:  $p[i + 1] = *$ 
42: return

```

Figure 2

Original HPM algorithm. Pseudo code for the original HPM algorithm with some improvements.

Input: $\pi \times k$ matrix M where $M[i][j]$ = value of chromosome i at marker j , phenotype vector Φ , x , l , g

Output: \mathcal{P}

```

1:  $\mathcal{P} = \emptyset$ 
2:  $\pi_A$  = number of case chromosomes
3:  $\pi_C$  = number of control chromosomes
4: for  $i = 1$  to  $k$  do
5:   for all  $a_1 \in \text{alleles}(i)$  do
6:     for  $j = i$  to  $\min(i + l, k)$  do
7:       for all  $a_2 \in \text{alleles}(j)$  do
8:          $\text{hap}[1] \dots \text{hap}[m]$  = all haplotypes from pos  $i$  to  $j$  where  $\text{hap}[y][i] == a_1$  and  $\text{hap}[y][j] == a_2$ 
9:         for  $z = 1$  to  $m$  do
10:           $I[z]$  = set of individuals having  $\text{hap}[z]$ 
11:           $na[z]$  = number of case chromosomes in  $I[z]$ 
12:           $nu[z]$  = number of control chromosomes in  $I[z]$ 
13:         end for
14:         for  $z = 1$  to  $m$  do
15:           $\text{depthFirst}(z, na[z], nu[z], \text{hap}[z], \{z\})$ 
16:         end for
17:       end for
18:     end for
19:   end for
20: end for

21: Procedure  $\text{depthFirst}(maxNo, \pi_{AP}, \pi_{CP}, P, S)$ 
22: if  $P$  cannot be legal then
23:   return
24: end if
25: if  $P$  is valid wrt.  $S$  and  $\pm\chi^2(\pi_{AP}, \pi_{CP}, \pi_A - \pi_{AP}, \pi_C - \pi_{CP}) \geq x$  then
26:   if  $P$  is legal then
27:      $\mathcal{P} = \mathcal{P} \cup \{P\}$ 
28:   end if
29:   for all legal patterns  $P'$  equivalent to  $P$  that can be created by inserting * symbols in  $P$  do
30:      $\mathcal{P} = \mathcal{P} \cup \{P'\}$ 
31:   end for
32: end if
33: for  $z = maxNo + 1$  to  $m$  do
34:    $P' = \text{consensus pattern of } P \text{ and } \text{hap}[z]$ 
35:    $S = S \cup \{z\}$ 
36:    $\text{depthFirst}(z, \pi_{AP} + na[z], \pi_{CP} + nu[z], P', S)$ 
37:    $S = S \setminus \{z\}$ 
38: end for

```

Figure 3
New HPM algorithm. Pseudocode for the new HPM algorithm.

the Wellcome Trust Case-Control Consortium (WTCCC) [4].

Old algorithm

The algorithm presented in [11] recursively generates haplotype patterns using a depth-first-search strategy. To

avoid looking at all possible patterns the algorithm prunes away parts of the search tree based on a lower bound on the number of disease-associated chromosomes that match a pattern.

Some simple improvements can be made to this algorithm. As presented in the paper counting the number of affected and unaffected individuals that match the pattern in each call to *depthFirst* will take time $O(n \cdot l)$ where n is the number of individuals and l is the length of the pattern. If we remember which individuals match the pattern at a given time then we only need to look through these when a new non- "*" symbol is inserted in the pattern. Pseudo code for the algorithm with this improvement is shown in fig. 2. The improvement greatly speeds up the algorithm.

New algorithm

The idea of the new algorithm is to exploit that LD structure means that you usually only see a handful of the 2^n possible haplotypes if you look at n neighboring SNPs. Instead of looking at all different haplotype patterns spanning a region we look at all combinations of haplotypes over the region. We search these haplotype sets in a depth-first-search but stop examining a branch if there is no legal haplotype pattern that could occur in all of the haplotypes in the current set.

Induced pattern

Given a set of haplotypes $h_1 \dots h_k$ the induced pattern of the set is the haplotype pattern that occurs in all of the haplotypes and contains fewest possible "*" ("don't care") symbols.

An induced pattern over a set of haplotypes that is not legal can sometimes be made legal by inserting extra "*" symbols if $s > 2$. This happens if a pattern is illegal because it contains too many gaps but would become legal if two gaps were joined into one. If for example $l = 5, g = 1$ and $s = 3$ then "0 * 1 * 0" is an illegal pattern because it contains more than g gaps. The pattern can however be made legal by substituting the "1" for a "*" yielding the pattern "0 * * 0".

Valid pattern

An induced pattern over a set S of haplotypes is said to be valid with regard to S , if the pattern occurs in all of the haplotypes in S but not in any of the other haplotypes found in the input data.

Equivalent pattern

A haplotype pattern will split the set of individuals into those that match the patterns and those that do not. We say that two patterns are equivalent if they result in the same bipartitions of the set of individuals.

The algorithm

The new algorithm (Fig. 3) looks at sets of haplotypes. It traverses all possible combinations of haplotypes by gradually expanding a set one haplotype at the time. If at any

point the induced pattern of the current set of haplotypes cannot be made into a legal pattern by adding extra "*" characters the current set is not expanded further. If a pattern is valid and strongly associated it is added to the output set along with all its equivalent patterns.

Results and discussion

We have implemented both the old and the new algorithm in Python using the *SNPfile* library [21] to read and store the input data. To evaluate the algorithms, we have used the Crohn's disease data set from the Wellcome Trust Case-Control Consortium (WTCCC) [4]. This data set contains 491208 markers in 2005 disease affected individuals and 3004 unaffected control individuals. We used the *Beagle* [22] program to phase the haplotypes and infer missing genotypes.

Time vs. number of individuals

First we tested the running time as a function of the number of individuals. From the WTCCC data we created test data by picking subsets of individuals, keeping the affected/unaffected ratio constant, and we then ran both algorithms on chromosome 22. Figure 4 shows the "wall time" of both algorithms for varying data sizes. Both algorithms show a linear increase but with the original algorithm having the highest increment.

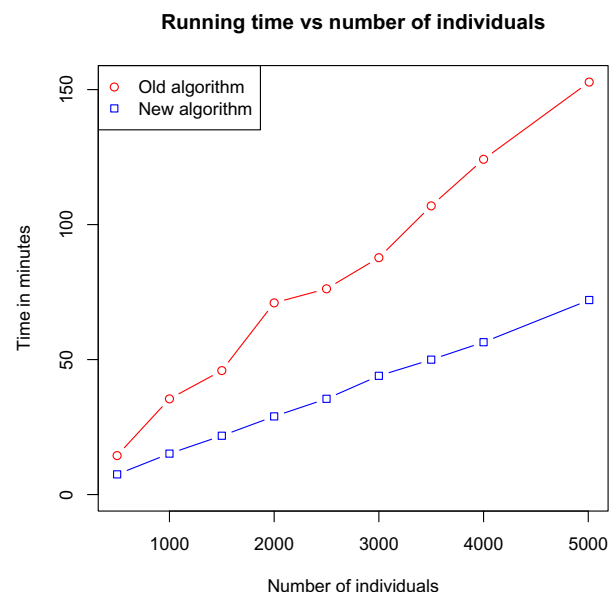


Figure 4
Time vs. number of individuals. The time consumption of the two algorithms as a function of the number of individuals in the data sets. The parameters were the default parameters specified in [10] ($l = 7, g = 2, s = 2, x = 9$).

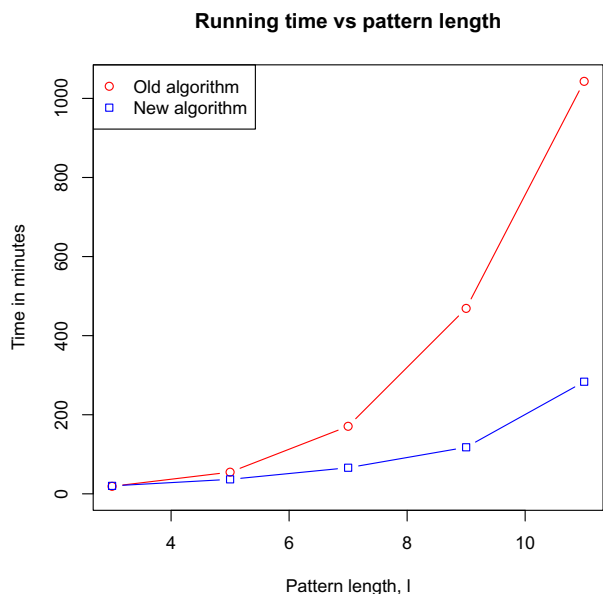


Figure 5
Time vs. maximal allowed pattern length. The time consumption of the two algorithms as a function of the maximal allowed pattern length (l). The rest of the parameters were fixed at the default settings ($g = 2, s = 2, x = 9$).

Time vs. pattern length

An important parameter for the running time is the maximal allowed pattern length, l . Figure 5 show the running time of the two algorithms as a function l , when analysing the full chromosome 22 from the WTCCC data set. The running time of both algorithms clearly grows super-linear, but with the time for the new algorithm clearly growing slower.

Time vs. haplotype diversity

Since the time usage of the new algorithm depends on the number of different haplotypes over a region we expect it to use less time in regions with few distinct haplotypes and more time in regions with many distinct haplotypes. Figure 6 shows the running time (with pattern length $l = 11$) and the number of haplotypes along chromosome 22 of the WTCCC data: The plot on the left shows both the running time per marker (the time to test all patterns beginning in a given marker) together with the number of distinct haplotypes starting in a given marker. Figure 7 shows the running time for scoring a marker as a function of the number of unique haplotypes overlapping the marker.

The same dependency on haplotype diversity is not seen for the old algorithm (results not shown), nor is it expected to be as the old algorithm does not depend on

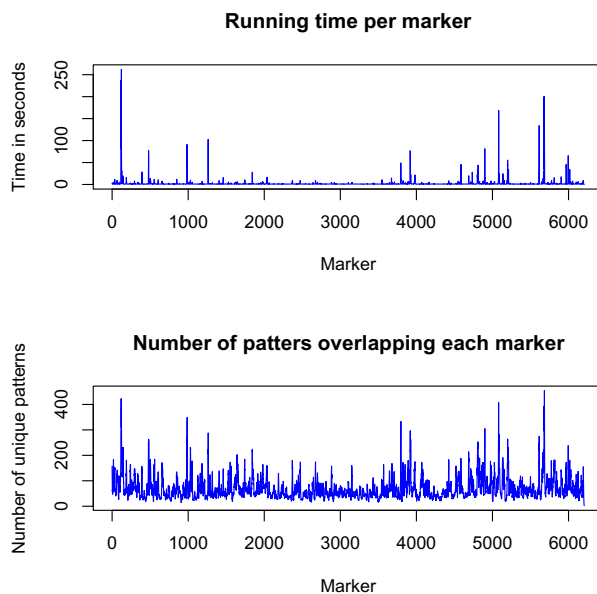


Figure 6
Running time and number of unique haplotypes per marker. The time usage per marker on chromosome 22 of the WTCCC Crohn's disease data (top) and the number of unique haplotypes overlapping each marker (bottom).

the number of distinct haplotypes seen in the data. Instead, the running time could depend on the maximal score we see when scoring a marker, since this is the threshold used in the branch and bounds heuristic. From the data, however, we do not see a significant effect here.

Genome-wide analysis

As the final comparison of the old and the new HPM algorithm, we compare the running time on the full Crohn's disease data set. Table 1 shows the time consumption of the two algorithms on each chromosome.

Conclusion

We have developed a new algorithm for the haplotype pattern mining method and shown that it outperforms the original algorithm on genome wide association data. As a function of the number of individuals or the maximal pattern length, both the new and old algorithm appears to have the same asymptotic running time, with the new algorithm having a significantly smaller time overhead.

The new algorithm is very sensitive to the haplotype diversity. The same is not the case for the old algorithm, but here the mean running time per marker is 8.8 ± 0.57 sec

onds (with pattern length $l = 11$) where for the new algorithm the mean running time per marker is 2.5 ± 9.6 sec

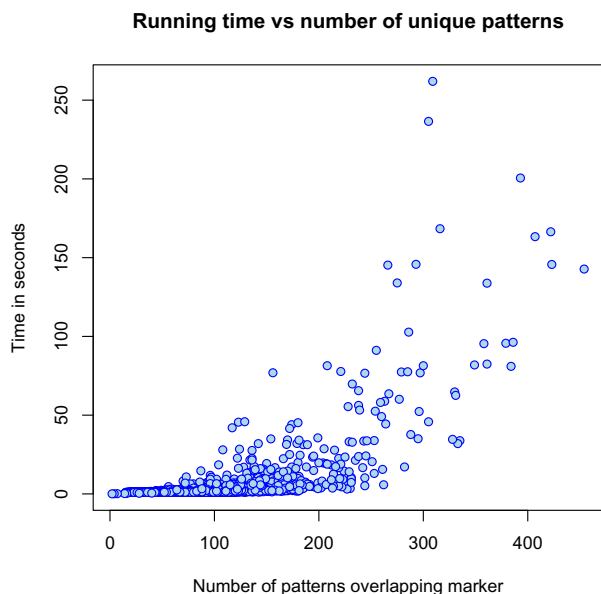


Figure 7
The relationship between time usage and haplotype diversity. Time usage per marker vs. number of haplotypes overlapping the marker.

onds. It might therefore be worthwhile to use a hybrid

Table 1: Time per chromosome. Table showing the time it took to analyze the different chromosomes of the WTCCC Crohn's disease data set. With the following parameters: $l = 7, g = 2, s = 2, x = 20$.

Crohn's disease running times			
Chromosome	# markers	Original HPM (with optimisation)	New HPM
1	40220	9 hours, 16 minutes	4 hours, 51 minutes
2	41400	9 hours, 37 minutes	4 hours, 54 minutes
3	33799	7 hours, 48 minutes	4 hours, 1 minutes
4	32334	7 hours, 31 minutes	3 hours, 55 minutes
5	32056	7 hours, 20 minutes	3 hours, 59 minutes
6	31470	7 hours, 16 minutes	3 hours, 48 minutes
7	25835	5 hours, 52 minutes	3 hours, 09 minutes
8	27457	6 hours, 11 minutes	3 hours, 21 minutes
9	22864	5 hours, 13 minutes	2 hours, 50 minutes
10	28501	6 hours, 35 minutes	3 hours, 25 minutes
11	26273	6 hours, 8 minutes	3 hours, 11 minutes
12	24954	5 hours, 38 minutes	3 hours, 5 minutes
13	19188	4 hours, 19 minutes	2 hours, 19 minutes
14	15721	3 hours, 37 minutes	1 hour, 54 minutes
15	14355	3 hours, 16 minutes	1 hour, 47 minutes
16	15308	3 hours, 35 minutes	1 hour, 53 minutes
17	11281	2 hours, 37 minutes	1 hour, 23 minutes
18	14881	3 hours, 28 minutes	1 hour, 44 minutes
19	6399	1 hours, 28 minutes	43 minutes
20	12400	2 hours, 53 minutes	1 hour, 23 minutes
21	7125	1 hour, 38 minutes	51 minutes
22	6207	1 hour, 27 minutes	44 minutes

algorithm where the new algorithm is used in areas with lower haplotype diversity and the old algorithm is used in areas with high haplotype diversity. If this would reduce the time usage on markers now taking more than 4 seconds to only 3, the hybrid algorithm would spend 1.4 ± 0.63 seconds per marker.

Competing interests

The authors declare that they have no competing interests.

Authors' contributions

SB developed the algorithm and implemented the software. All authors designed the experiments. TM and SB drafted the manuscript. All authors read and approved the manuscript.

Acknowledgements

TM is funded by the Danish Research Council, grant FNU-272-07-0380.

This article has been published as part of BMC Bioinformatics Volume 10 Supplement 1, 2009: Proceedings of The Seventh Asia Pacific Bioinformatics Conference (APBC) 2009. The full contents of the supplement are available online at <http://www.biomedcentral.com/1471-2105/10?issue=S1>

References

1. Arking DE, Pfeufer A, Post W, Kao WH, Newton-Cheh C, Ikeda M, West K, Kashuk C, Akyol M, Perz S, Jalilzadeh S, Illig T, Gieger C, Guo CY, Larson MG, Wichmann HE, Marban E, O'donnell CJ, Hirschhorn JN, Kaab S, Spooner PM, Meitinger T, Chakravarti A: **A common**

- genetic variant in the NOS1 regulator NOSIAP modulates cardiac repolarization.** *Nat Genet* 2006, **38(6)**:644-651.
2. Smyth DJ, Cooper JD, Bailey R, Field S, Burren O, Smink LJ, Guja C, Ionescu-Tirgoviste C, Widmer B, Dunger DB, Savage DA, Walker NM, Clayton DG, Todd JA: **A genome-wide association study of nonsynonymous SNPs identifies a type 1 diabetes locus in the interferon-induced helicase (IFIH1) region.** *Nat Genet* 2006, **38(6)**:617-619.
 3. Amundadottir LT, Sulem P, Gudmundsson J, Helgason A, Baker A, Agnarsson BA, Sigurdsson A, Benediktsson KR, Cazier JB, Sainz J, Jakobsdottir M, Kostic J, Magnusdottir DN, Ghosh S, Agnarsson K, Birgisdottir B, Le Roux L, Olafsdottir A, Blondal T, Andresdottir M, Gretarsdottir OS, Bergthorsson JT, Gudbjartsson D, Gylfason A, Thorleifsson G, Manolescu A, Kristjansson K, Geirsson G, Isaksson H, Douglas J, Johansson JE, Balter K, Wiklund F, Montie JE, Yu X, Suarez BK, Ober C, Cooney KA, Gronberg H, Catalona WJ, Einarsson GV, Barkardottir RB, Gulcher JR, Kong A, Thorsteinsdottir U, Stefansson K: **A common variant associated with prostate cancer in European and African populations.** *Nat Genet* 2006, **38(6)**:652-658.
 4. **Genome-wide association study of 14,000 cases of seven common diseases and 3,000 shared controls.** *Nature* 2007, **447(7145)**:661-78.
 5. Gudmundsson J, Sulem P, Manolescu A, Amundadottir LT, Gudbjartsson D, Helgason A, Rafnar T, Bergthorsson JT, Agnarsson BA, Baker A, Sigurdsson A, Benediktsson KR, Jakobsdottir M, Xu J, Blondal T, Kostic J, Sun J, Ghosh S, Stacey SN, Mouy M, Saemundsdottir J, Backman VM, Kristjansson K, Tres A, Partin AW, Albers-Akkers MT, Godino-Ivan Marcos J, Walsh PC, Swinkels DW, Navarrete S, Isaacs SD, Aben KK, Graif T, Cashy J, Ruiz-Echarri M, Wiley KE, Suarez BK, Witjes JA, Frigge M, Ober C, Jonsson E, Einarsson GV, Mayordomo JL, Kiemene LA, Isaacs WB, Catalona WJ, Barkardottir RB, Gulcher JR, Thorsteinsdottir U, Kong A, Stefansson K: **Genome-wide association study identifies a second prostate cancer susceptibility variant at 8q24.** *Nat Genet* 2007, **39(5)**:631-7.
 6. Gudmundsson J, Sulem P, Steinthorsdottir V, Bergthorsson JT, Thorleifsson G, Manolescu A, Rafnar T, Gudbjartsson D, Agnarsson BA, Baker A, Sigurdsson A, Benediktsson KR, Jakobsdottir M, Blondal T, Stacey SN, Helgason A, Gunnarsdottir S, Olafsdottir A, Kristinsson KT, Birgisdottir B, Ghosh S, Thorlacius S, Magnusdottir D, Stefansson G, Kristjansson K, Bagger Y, Wilensky RL, Reilly MP, Morris AD, Kimber CH, Adeyemo A, Chen Y, Zhou J, So WY, Tong PC, Ng MC, Hansen T, Andersen G, Borch-Johnsen K, Jorgensen T, Tres A, Fuertes F, Ruiz-Echarri M, Asin L, Saez B, van Boven E, Klaver S, Swinkels DW, Aben KK, Graif T, Cashy J, Suarez BK, van Vierssen Trip O, Frigge ML, Ober C, Hofker MH, Wijmenga C, Christiansen C, Rader DJ, Palmer CN, Rotimi C, Chan JC, Pedersen O, Sigurdsson G, Benediktsson R, Jonsson E, Einarsson GV, Mayordomo JL, Catalona WJ, Kiemene LA, Barkardottir RB, Gulcher JR, Thorsteinsdottir U, Kong A, Stefansson K: **Two variants on chromosome 17 confer prostate cancer risk, and the one in TCF2 protects against type 2 diabetes.** *Nat Genet* 2007, **39(8)**:977-83.
 7. Saxena R, Voight BF, Lyssenko V, Burtt NP, de Bakker PI, Chen H, Roix JJ, Kathiresan S, Hirschhorn JN, Daly MJ, Hughes TE, Groop L, Altshuler D, Almgren P, Florez JC, Meyer J, Ardlie K, Bengtsson Bostrom K, Isomaa B, Lettre G, Lindblad U, Lyon HN, Melander O, Newton-Cheh C, Nilsson P, Orho-Melander M, Rastam L, Speliotes EK, Taskinen MR, Tuomi T, Guiducci C, Berglund A, Carlson J, Gianniny L, Hackett R, Hall L, Holmkvist J, Laurila E, Sjogren M, Sterner M, Surti A, Svensson M, Tewhey R, Blumenstiel B, Parkin M, Defelice M, Barry R, Brodeur W, Camarata J, Chia N, Fava M, Gibbons J, Handsaker B, Healy C, Nguyen K, Gates C, Sougnez C, Gage D, Nizzari M, Gabriel SB, Chirn GW, Ma Q, Parikh H, Richardson D, Ricke D, Purcell S: **Genome-wide association analysis identifies loci for type 2 diabetes and triglyceride levels.** *Science* 2007, **316(5829)**:1331-6.
 8. Pe'er I, de Bakker PI, Maller J, Yelensky R, Altshuler D, Daly MJ: **Evaluating and improving power in whole-genome association studies using fixed marker sets.** *Nat Genet* 2006, **38(6)**:663-667.
 9. de Bakker PIW, Yelensky R, Pe'er I, Gabriel SB, Daly MJ, Altshuler D: **Efficiency and power in genetic association studies.** *Nat Genet* 2005, **37(11)**:1217-1223.
 10. Toivonen HT, Onkamo P, Vasko K, Ollikainen V, Sevon P, Mannila H, Herr M, Kere J: **Data mining applied to linkage disequilibrium mapping.** *Am J Hum Genet* 2000, **67**:133-145.
 11. Toivonen HT, Onkamo P, Vasko K, Ollikainen V, Sevon P, Mannila H, Kere J: **Gene Mapping by Haplotype Pattern Mining.** In *IEEE International Symposium on Bio-Informatics and Biomedical Engineering (BIBE'00) Los Alamitos, CA, USA: IEEE Computer Society; 2000:99.*
 12. Liu JS, Sabatti C, Teng J, Keats BJ, Risch N: **Bayesian analysis of haplotypes for linkage disequilibrium mapping.** *Genome Res* 2001, **11(10)**:1716-1724.
 13. Morris AP, Whittaker JC, Balding DJ: **Fine-scale mapping of disease loci via shattered coalescent modeling of genealogies.** *Am J Hum Genet* 2002, **70(3)**:686-707.
 14. Larribe F, Lessard S, Schork NJ: **Gene mapping via the ancestral recombination graph.** *Theor Popul Biol* 2002, **62(2)**:215-229.
 15. Molitor J, Marjoram P, Thomas D: **Fine-scale mapping of disease genes with multiple mutations via spatial clustering techniques.** *Am J Hum Genet* 2003, **73(6)**:1368-1384.
 16. Zöllner S, Pritchard JK: **Coalescent-based association mapping and fine mapping of complex trait loci.** *Genetics* 2005, **169(2)**:1071-1092.
 17. Minichiello MJ, Durbin R: **Mapping Trait Loci by Use of Inferred Ancestral Recombination Graphs.** *American Journal of Human Genetics* 2006, **79(5)**:910-922.
 18. Mailund T, Besenbacher S, Schierup MH: **Whole genome association mapping by incompatibilities and local perfect phylogenies.** *BMC Bioinformatics* 2006, **7**:454.
 19. Li J, Jiang T: **Haplotype-based linkage disequilibrium mapping via direct data mining.** *Bioinformatics* 2005, **21(24)**:4384-4393.
 20. Browning SR: **Multilocus association mapping using variable-length Markov chains.** *Am J Hum Gen* 2006, **78(6)**:903-913.
 21. Nielsen J, Mailund T: **The SNPFile library.** [<http://www.birc.au.dk/~mailund/SNPFile/>].
 22. Browning SR, Browning BL: **Rapid and Accurate Haplotype Phasing and Missing-Data Inference for Whole-Genome Association Studies By Use of Localized Haplotype Clustering.** *Am J Hum Genet* 2007, **81(5)**:1084-1097.

Publish with **BioMed Central** and every scientist can read your work free of charge

"BioMed Central will be the most significant development for disseminating the results of biomedical research in our lifetime."

Sir Paul Nurse, Cancer Research UK

Your research papers will be:

- available free of charge to the entire biomedical community
- peer reviewed and published immediately upon acceptance
- cited in PubMed and archived on PubMed Central
- yours — you keep the copyright

Submit your manuscript here:
http://www.biomedcentral.com/info/publishing_adv.asp

