单体型组装 MEC 问题的参数化算法研究

谢民主[1,2],王建新[2],陈建二[2]
XIE Min-zhu[1,2],WANG Jian-xin[2],CHEN Jian-er[2]

1.湖南师范大学 物理与信息科学学院,长沙 410081
2.中南大学 信息科学与工程学院,长沙 410083
1.College of Physics and Information Science,Hunan Normal University,Changsha 410081,China
2.School of Information Science and Engineering,Central South University,Changsha 410083,China
E-mail:xieminzhu@sina.com

XIE Min-zhu,WANG Jian-xin,CHEN Jian-er.Research on parameterized algorithm of Haplotypes assembly MEC problem.Computer Engineering and Applications,2007,43(35):57-60.

Abstract: The haplotype assembly MEC problem is the computational problem of inducing a pair of haplotypes from an individual's DNA fragments sequencing data by correcting minimum SNPs.Based on the characters of DNA fragments,the paper introduces a parameterized algorithm of time complexity $O(nk_2 2^{k_2}+m\log m+mk_1)$ with m fragments,n SNPs,the maximum number of SNP sites that a fragment covers k_1(usually smaller than 10) and the maximum number of the fragments covering a SNP site k_2(usually no more than 10).For the practical fragment data,the algorithm can solve the MEC problem efficiently even if m and n are larger and it is scalable and applicable in practice.

Key words: bioinformatics;haplotyping;parameterized algorithm;SNPs(Single-Nucleotide Polymorphisms)

摘 要:单体型组装 MEC 问题指如何利用个体的 DNA 测序片断数据,翻转最少的 SNP 位点值以确定该个体单体型的计算问题。根据片段数据的特点提出了一个时间复杂度为 $O(nk_2 2^{k_2}+m\log m+mk_1)$的参数化算法,其中 m 为片段数,n 为单体型的 SNP 位点数,k_1为一个片断覆盖的最大 SNP 位点数 (通常小于 10),k_2为覆盖同一 SNP 位点的片段的最大数 (通常不大于 10)。对于实际 DNA 测序中的片段数据,即使 m 和 n 都相当大,该算法也可以在较短的时间得到 MEC 问题的精确解,具有良好的可扩展性和较高的实用价值。

关键词:生物信息学;单体型检测;参数化算法;单核苷酸多态性
文章编号:1002-8331(2007)35-0057-04 **文献标识码**:A **中图分类号**:TP301

1 引言

不同的人的外貌和体格各不相同,对相同的疾病具有不同的免疫能力,对药物也具有不同的敏感性。从遗传上说,这些现象是因为不同个体的基因组不完全相同。两个人之间的 DNA 差异约占基因组的 0.1%。在人群中 1%上的个体中出现的染色体某个位点上的碱基变异称为单核苷酸多态性 SNPs (Single-Nucleotide Polymorphisms)。SNPs 广泛分布在人类基因组中,在整个人类基因组中大约有 340 万个 SNPs[1]。

SNPs 可用于个体识别、亲子鉴定。分析和识别 SNPs 对基因的精确定位、了解基因功能很有帮助,对遗传病等疾病的诊断和药物研究有重要作用。人类的染色体是成对存在的,在一条染色体上的某一区域的 SNP 位点上的碱基序列叫做单体型(haplotype)。单体型在 SNPs 的上述应用中扮演着重要的角色,不幸的是在当前的实验技术下,直接测定个体的单体型既费钱又费时间,因此利用计算机技术来确定个体的单体型有极其重要的现实意义。

2 单体型组装问题

由于在生物技术上很难把一对染色体分开, 对 DNA 进行测序时, 实验室只能对来自一对染色体的不同单体的很短的 DNA 片断进行直接测序。测序过程中也不可避免地会发生一些错误。个体单体型问题就是给定某个个体一组已经测序的 DNA 片断数据, 如何根据某些特定的优化原则得出该个体一对单体型。

基金项目:国家自然科学基金重点项目(the Key Project of National Science Foundation of China under Grant No.60433020);湖南省教育厅资助科研课题(the Research Project of Department of Education of Hunan Province,China under Grant No.06C526)。

作者简介:谢民主(1969-),男,博士研究生,主要研究领域为生物信息学;王建新(1969-),男,博士,教授,博士生导师,主要研究领域为网络优化、生物信息学;陈建二(1954-),男,博士,教授,博士生导师,主要研究领域为生物信息学、计算复杂性及优化。

对于任意一个 SNP 位点来说，一对染色体上的碱基可以是相同的，这种现象叫纯合(homozygous)；也可以是不同的，这种现象叫做杂合(heterozygous)。这样一条染色体在 SNP 位点的投影序列即单体型就可以用 2 个字母的字符集$\{A,B\}$上的字符序列来表示，而不必用真正的碱基字符 A、C、G、T 来表示。

n 个 SNP 位点按在染色体上的次序从左到右记作 $S:\{1,2,\cdots,n\}$，m 个片断记作 $F:\{1,2,\cdots,m\}$，任意 SNP 位点应该被某些DNA 片断覆盖，任意片断在它所覆盖的 SNP 位点的取值为“A”、“B”或“-”，其中“-”表示片断在该位点的取值为空(其值未能确定)，或者说该片断在这个 SNP 位点上有一个洞(hole)。如在图 1 中第 2 个片断覆盖的 SNP 位点是 2~6。DNA 片断的数据集就可以表示为在$\{A,B,-\}$上的一个 $m\times n$ 的矩阵，叫做 SNP 矩阵 $\boldsymbol{M}$[2]。其中 $\boldsymbol{M}$ 的行表示片断，列表示 SNP 位点，$\boldsymbol{M}_{i,j}$ 的值表示第 i 个片断在第 j 个 SNP 位点上的取值。

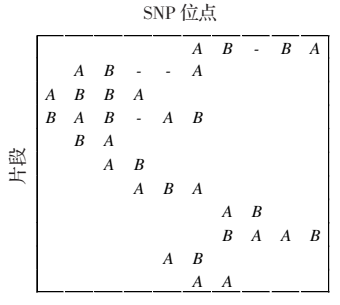

图 1 SNP 矩阵

对于一个 SNP 矩阵 $\boldsymbol{M}$ 有如下概念或定义：

对于 $\boldsymbol{M}$ 的两行，如果它们在列 j 上的值不相等，且都不为“-”，则这两行在列 j 上冲突；如果这两行在所有的列上均不冲突，则这两行互相兼容。

如果 $\boldsymbol{M}$ 的所有行可以分成 2 个不相交的子集，每个子集中的所有行都相互兼容则 $\boldsymbol{M}$ 是可行的。

显然，一个 SNP 矩阵 $\boldsymbol{M}$ 是可行的当且仅当可以找到一对单体型，使得 $\boldsymbol{M}$ 中的任意行总是可以与其中的一个单体型兼容(单体型表示为$\{A,B\}$上的长为 n 的字符序列)。

Lippert 等人最初引入了 MEC 问题[3]：

MEC(Minimum Error Correction)问题：给定一个 SNP 矩阵 $\boldsymbol{M}$，把其中的某些 A 翻转成 B，某些 B 翻转成 A，在翻转次数最少的条件下使 $\boldsymbol{M}$ 可行。

$\boldsymbol{M}$ 的 MEC 解就是使 $\boldsymbol{M}$ 可行所需翻转的最少的 SNP 值的个数，记作 $MEC(\boldsymbol{M})$。MEC 是 NP-hard，至今还没有实际有效的算法求得其精确解，Wang 设计了一个分支限界算法[4]，该算法的时间复杂度是 $O(2^m)$，只适合于片断数不多的场合。本文下面根据 DNA 片断数据的特点，特提出参数化算法来有效地求解实际的 MEC 问题。

3 参数化 MEC 算法

当前 DNA 测序的主导方法是 Sanger 双脱氧链终止法[5]。采用 Sanger 双脱氧链终止法测序，一次能测定的 DNA 序列的长度仅为 800~1 200 个碱基。各大测序中心使用的第三代测序仪如 ABI 3730、MageBACE 等可直接测定的片断的最大长度约为 1 000 个碱基。SNP 位点的平均分布密度约为 1/1 000[6]。虽然 SNP 在整个染色体上的分布很不均匀，从已有的数据来看，长度为 1 000 b 的片断上的 SNP 位点是极其有限的，为 10 个以内[7,8]。

出于时间和成本的考虑，实际测序得到的 DNA 片断对 SNP 位点的覆盖度是有限的，在目前的全基因组测序实验中，片段覆盖率约为 5 左右[6,9]。当测定某个体一条染色体上的单体型时，覆盖某一个 SNP 位点的 DNA 片断数远小于片断总数。在通常情况下，与 SNP 位点数(n)及 DNA 片断总数(m)相比，一个片断覆盖的 SNP 位点数(通常小于 $10^{[7,8]}$)和覆盖一个 SNP 位点的片断数(通常不大于 $10^{[10]}$)均是一个很小的数。

根据以上事实，特提出以下参数化条件：

定义 1 (k_1,k_2)参数化条件：k_1、k_2 是正整数，(k_1,k_2)参数化条件定义为片段的长度不超过 k_1，覆盖任意 SNP 位点的片段数不超过 k_2。

对 SNP 矩阵 $\boldsymbol{M}$ 而言，(k_1,k_2)参数化条件等价为矩阵 $\boldsymbol{M}$ 每一行第一个非空字符和最后一个非空字符间最多相隔 k_1-2 列；矩阵 $\boldsymbol{M}$ 每一列最多有 k_2 个行覆盖。

对于 SNP 矩阵 $\boldsymbol{M}$ 的一个 MEC 的解来说，翻转对应的 SNP 位点后，所有的行应该可以划分成 2 个子集，在同一子集中的片断相互兼容，即来自同一个单体型。这样，$\boldsymbol{M}$ 中的任意一行，它不是被划分在子集 0 中，就是被划分在子集 1 中。

定义 2 划分函数：划分函数 P 是定义在某个行集 R 上的一个映射：$P:R\rightarrow\{0,1\}$，即 P 把 R 中的行映射到 0 或 1。

令覆盖列 j 的行的有序集为 $RowSet(j)$，为了叙述的简便，$RowSet(j)$上的划分函数被称为列 j 上的划分函数。

显然对于一个满足(k_1,k_2)参数化条件的 SNP 矩阵，覆盖列 j 的行数不会大于 k_2，那么在列 j 上的可能的划分函数的个数不超过 2^{k_2}。

定义 3 投影和扩展：令 P 是定义在行集 R 上的一个划分函数，R'是 R 的子集，P'是定义在 R'上的一个划分函数，如果对于任意行 $i\in R'$，有 $P(i)=P'(i)$，则 P'是 P 在 R'上的投影，而 P 是 P'在 R 上的扩展。

定义 4 $E[P,j]$，$ES[P,j]$：P 为列 j 上的划分函数，$E[P,j]$定义为在满足下述条件下 $\boldsymbol{M}[:,j]$中必须翻转的最少 SNP 位点数，$ES[P,j]$则是对应的必须翻转的 SNP 位点的集合：

(1)$ES[P,j]$中的位点所在的列属于 1~j；

(2)对 $ES[P,j]$中的位点进行翻转后，存在着一个划分把 $\boldsymbol{M}[:,j]$中的行划分成两个子集，使得同一个子集中的行在列 1~j 上不冲突，且对于覆盖列 j 的任意行 i，如果它被划分在子集 0中当且仅当 $P(i)=0$；如果它被划分在子集 1 中当且仅当 $P(i)=1$。

根据定义 4，下式显然成立：

$$MEC(\boldsymbol{M})=\min_{P:P\text{是列}n\text{上的划分函数}}(E[P,n]) \quad (1)$$

对于列 j 上的划分函数 P，覆盖列 j 的 P 值等于 k 的行在列 j 上取“A”值的行数计作 $NofAs(P,j,k)$，取“B”值的行数计作 $NofBs(P,j,k)$(k 的值为 0 或 1)。如果 $NofAs(P,j,k)>NofBs(P,j,k)$，则 $Minor(P,j,k)$=“B”，否则 $Minor(P,j,k)$=“A”。

在覆盖列 j 的行按照 P 进行划分的情况下，为了使行不冲突，必须把被划分在同一子集中的所有行在同一列上的值统一起来。为了使翻转的位点最少，如果 $M_{i,j}=Minor(P,j,P(i))$，则必须对 $M_{i,j}$ 的值进行翻转。由此可知 $ES[P,1]$和 $E[P,1]$的值为：

$$ES[P,1]=\{(i,1)|i\in RowSet(1)\wedge M_{i,1}=Minor(P,1,P(i))\} \quad (2)$$

$$E[P,1]=|ES[P,1]| \quad (3)$$

即 $ES[P,1]$中要翻转的位点数，令既覆盖列 j_1 又覆盖列 j_2 的行的集合为 $ComRow(j_1,j_2)$。

定义 5 $B[P',j]$,$BS[P',j]$:P'为 $ComRow(j,j+1)$上的一个划分函数,$B[P',j]$定义为在满足下述条件下 $M[:,j]$中必须翻转的最少 SNP 位点数,$BS[P',j]$就是对应的翻转的 SNP 位点的集合:

(1)$BS[P',j]$ 位点所在的列属于 1~j;

(2)对 $\boldsymbol{M}[:,j]$中的所有行,在对 $BS[P',j]$的位点进行翻转后,存在着一个划分把这些行划分成两个子集,使得同一个子集中的行在列 1~j 上不冲突,且对于 $ComRow(j,j+1)$中任意行 i,如果它被划分在子集 0 中当且仅当 $P'(i)=0$;如果它被划分在子集 1 中当且仅当 $P'(i)=1$。

根据定义 5,要使 SNP 矩阵 $\boldsymbol{M}[:,j]$可行,在 $ComRow(j,j+1)$中的片断按照划分函数 P'进行划分的情况下,最少翻转的 SNP 位点数为 $B[P',j]$。

一旦对于 P'在 $RowSet(j)$上的所有可能的扩展 P, $E[P,j]$都已求出,那么 $B[P',j]$就是其中的最小值,即下列等式成立:

$$B[P',j]=\min_{P:P\text{是}P'\text{在}RowSet(j)\text{上的扩展}}(E[P,j]) \quad (4)$$

$$BS[P',j]=ES[P,j]|P=\arg\min_{P:P\text{是}P'\text{在}RowSet(j)\text{上的扩展}}(E[P',j]) \quad (5)$$

反过来,对于列 $j(j>1)$上的某一划分函数 P,其在 $ComRow(j-1,j)$的投影 P'是唯一的,如果知道了 $B[P',j-1]$和 $B[P',j-1]$,容易证明可由以下等式得出 $E[P,j]$和 $ES[P,j]$的值:

$$ES[P,j]=BS[P',j-1]\cup\{(i,j)|M_{i,j}=Minor(P,j,P(i))\} \quad (6)$$

$$E[P,j]=|ES[P,j]| \quad (7)$$

根据等式(2)、(3)可得出 E 和 ES 的初始值,然后由式(4)~(7)进行递推最终可求出 $\boldsymbol{M}$ 的 MEC 解。具体算法见图 2。

```
P_MEC 算法
输入:m×n SNP 矩阵 M
输出:M 的 MEC 解
Step1 初始化:对 M 中的行按其第一个非空字符所在的列的序号进行非降序排列,扫描 M 得到覆盖列 j 的行号的有序集 RowSet[j]及行数 H[j],j 从列 1 到 n
Step2  j=1;
    for (P=0;P<2^H[1];P++) //划分函数用一个二进制数编码
    { // E[P]存贮 E[P,1],ES[P]存贮 ES[P,1]
      E[P]=0;ES[P]=ϕ;
      // 根据公式(2)、(3)计算 E[P,1]和 ES[P,1],见图 3
      CompFlips(1,P,E[P],ES[P]);}
Step3  while(j<n)    //根据公式(4)~(7)递推,MAX 为最大整数
{ //计算列 j 与 j+1 共有的行数 CR 和表示覆盖列 j 的行是否覆盖列 j+1 的向量 Bits:Bits[i]=1 denotes the ith//row of RowSet[j] covers column j+1,见图 4
Step3.1 Common(j,Bits,CR);
Step3.2 for (P'=0;P'<2^CR;P'++) B[P']=MAX;
Step3.3 for (P=0;P<2^H[j];P++)
    {  //得到 P 在 ComRow(j,j+1)上的投影 P',见图 5
      Project(P,P',Bits,H[j]);
if (B[P']>E[P]) {B[P']=E[P];BS[P']=ES[P]; } //Eq.(4)(5)
Step3.4 j++; //下一列
Step3.5 for (P=0;P<2^H[j];P++){E[P]=MAX; ES[P]=Ø;}
Step3.6 for (P'=0; P'<2^CR; P'++)
Step3.6.1 for(p=0;p<2^(H[j]-CR);p++) //得到 P'在 RowSet(j)上的扩展
    { P=P'|(p<<CR); //位或,M 中的行有序,这样覆盖列 j
//而不覆盖列 j-1 的行序比 ComRow (j-1,j)中的行序大
      deltEp=0; deltESp=Ø;
      CompFlips(j,P,deltEp,deltESp);
      E[P]=deltEp+B[P']; ES[P]=BS[P']∪deltESp; //Eq.(6)(7)
    }    }
Step4 输出最小的 E[P]及相应的 ES[P]  (P=0..2^H[n]-1). // Eq.(1)
```

图 2　P_MEC 算法

定理 1 P_MEC 算法能正确地求出 SNP 矩阵 $\boldsymbol{M}$ 的 MEC 解。如果 $\boldsymbol{M}$ 满足(k_1,k_2)参数化条件,则其时间复杂度为 $O(nk_2 2^{k_2}+m\log m+mk_1)$,空间复杂度为 $O(mk_1 2^{k_2}+nk_2)$。

证明 该算法的正确性由等式(1)~(7)的正确性来保证。

$\boldsymbol{M}$ 满足 (k_1,k_2) 参数化条件,则意味着任意一行覆盖的 SNP 位点数不会超过 k_1, 覆盖任意一个 SNP 位点的行数不超过 k_2,因此 $\boldsymbol{M}$ 可以采用如下的存储结构:每一行存储其第一个和最后一个非空列的序号,再保存该行从第一个非空列到最后一个非空列的值, 这样 $\boldsymbol{M}$ 所需的存储空间为 $O(mk_1)$,$RowSet$ 所需的空间为 $O(nk_2)$,H 所需的空间为 $O(n)$,E 和 B 为 $O(2^{k_2})$,ES 和 BS 为 $O(mk_1 2^{k_2})$,所以算法的空间复杂度为 $O(mk_1 2^{k_2}+nk_2)$。

下面分析其时间复杂度:Step1 需时间 $O(m\log m+mk_1)$;CompFlips 函数需时间 $O(k_2)$,所以 Step2 需时间 $O(k_2 2^{k_2})$;Step3.1 调用 Common 函数需时间 $O(k_2)$,Step3.2 需时间 $O(2^{k_2})$,Project 函数需时间 $O(k_2)$,故 Step3.3 需时间 $O(k_2 2^{k_2})$,Step3.5 需时间 $O(2^{k_2})$,Step3.6 需时间 $O(k_2 2^{k_2})$,而 Step3 循环 n 次,所以所需时间为 $O(nk_2 2^{k_2})$;Step4 所需时间为 $O(2^{k_2})$。整个算法的时间复杂度为 $O(nk_2 2^{k_2}+m\log m+mk_1)$。

4　实验结果与结论

原始的 DNA 测序片段数据很难得到, 很多文献都是利用计算机模拟真实生物数据的特征生成测试数据集进行单体型组装问题的各种算法的实验测试[4,11,12],本文也采用与上述文献相同的方法和参数来生成测试数据。模拟数据生成的方法如下:首先随机生成指定长度的单体型,根据指定的两个单体型的差异率来随机生成另一个单体型。然后根据指定片段的覆盖率、片段的最小长度和最大长度来随机生成片段数据,最后根据指定的测序误差 e 和空隙率 p 植入错误和空值的 SNP 位点。实验室的 DNA 测序误差为 3%~5%[12], 片段的覆盖率为 5 左右[6,9]。为了使模拟生成的片段数据能很好地反映真实情况,根据文献[12]的测试方法,先采用著名的 shotgun 测序模拟片断生成器 Celsim[13]生成一系列的片断数据,生成参数设置为 2 个

```
CompFlips(j,P,Ep,ESp)  //初始值:Ep=0,ESp[P]=Ø
{ // As[k],Bs[k]分别表示 NofAs(P,j,k),NofBs(P,j,k),k=0,1
  As[0]=Bs[0]=As[1]=Bs[1]=0;
    bit=0;shift=P;
     按序取 RowSet(j)中的行号赋值给 i  do
     {   bit=shift&1; //位与,bit=P(i)
         shift=shift>>1;//右移 1 位
         if M_{i,1}=='A' then As[bit] ++;
         if M_{i,1}=='B' then Bs[bit] ++;}
     for bit=0..1 do
     {  Minor[bit]=As[bit]>Bs[bit]? 'B':'A';}
     shift=P;
     按序取 RowSet(j)中的行号赋值给 i  do
     {  bit=shift&1; shift=shift>>1;
        if (M_{i,j}==Minor[bit])
        {  Ep++;ESp=ESp∪(i,j); }
}  //计算列 j 在 P 的划分下要翻转的最少位点
```

图 3　CompFlips 函数

单体型的差异率为10%，生成的普通片段的最小长度为3，最大长度为6，片段覆盖率采用10，单体型长度 n、片断数 m 和测序误差 e 则按照需要进行变化以比较本文的 P_MEC 和 Wang[4] 的 BNB_MEC 算法的性能。模拟片断生成器的详细情况请参照文献[12,13]。

本文用 C++语言实现了 P_MEC，BNB_MEC 的源代码来自 Wang[4]，算法运行在一台 Linux 服务器上(4 个 Intel Xeon3.6 G CPU，4 G RAM)。实验对算法的运行时间和单体型重建率(Reconstruction rate)进行了比较。单体型重建率指的是算法得出的单体型中正确的 SNP 位点数与总的 SNP 位点数的比值[4]。

```
Common(j,Bits,CR)
{  CR=0; //初始化:覆盖列j的行不覆盖列j+1
   for(i=0;i<H[j];i++)Bits[i]=0;
   i=0; i1为RowSet(j)的第一行;//RowSet为有序集
   i2为RowSet(j+1)的第一行;
   while(i1>-1 && i2>-1 )
   { CASE1  i1>i2:
     {i2=RowSet(j+1)中下一行(如果没有则为-1,下同);}
     CASE2  i1<i2:{ i1=RowSet(j)中下一行;i++;}
     CASE3  i1==i2:
     { Bits[i]=1;// RowSet(j)中的第i行覆盖列j+1
   i1=RowSet(j)中下一行;
   i2=RowSet(j+1)中下一行;CR++;i++;
}
     }      }
```

图4 Common 函数

表1 算法性能比较

Parameters			Average running time[1]/s		Maximum running time[1]/s		Reconstruction rate[1]	
n	m	e	BNB_MEC	P_MEC	BNB_MEC	P_MEC	BNB_MEC	P_MEC
		0.01	0.038	0.002	0.123 6	0.011	1.000	1.000
16	32	0.03	0.412	0.002	2.531 5	0.011	0.991	0.991
		0.05	5.613	0.002	56.793	0.013	0.980	0.980
		0.01	185.882	0.002	738.325 0	0.018	0.971	0.971
22	44	0.03	200.312	0.003	1 501.560	0.018	0.961	0.961
		0.05	531.235	0.003	3 210.135	0.019	0.953	0.953
		0.01	–	0.012	>96 h	0.029	–	0.957
50	100	0.03	–	0.017	>96 h	0.031	–	0.932
		0.05	–	0.024	>96 h	0.031	–	0.908

表1显示当 n 和 m 增加时，BNB_MEC 的运行时间急剧上升。当 n=50、m=100 时，经过4天 BNB_MEC 仍无法得出结果，而 P_MEC 的运行时间却不到1 s。由于 BNB_MEC 和 P_MEC 都是精确算法，它们的单体型重建率没有区别，要比其它近似算法(如遗传算法)好[4]。当测序误差 e 较小时，两算法的单体型重建率都很高，在90%以上。图5和图6显示当单体型长达300个 SNP 位点时，P_MEC 仍有较好的性能。从实验结果可以看出，P_MEC 具有良好的扩展性和较高的实用价值。对于其它扩展的 MEC 模型如 MEC/GI[4]，进行相应的修改后，P_MEC 算法同样可以应用。(收稿日期：2007年6月)

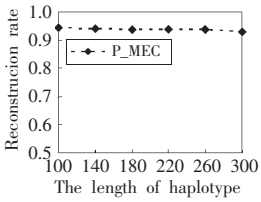

图5 P_MEC 的重建率

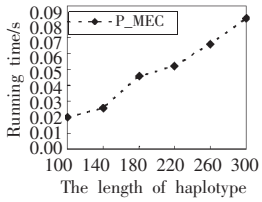

图6 P_MEC 的运行时间

参考文献：

[1] The International SNP Map Working Group.A map of human genome sequence variation containing 1.42 million single nucleotide polymorphisms[J].Nature,2001,409(6822):928-933.

[2] Lancia G,Bafna V,Istrail S,et al.SNPs problems,complexity and algorithms[C]//Heide F M.LNCS 2161:Proc of the 9th Ann European Symp on Algorithms.Heidelberg:Springer,2001:182-193.

[3] Lippert R,Schwartz R,Lancia G,et al.Algorithmic strategies for the single nucleotide polymorphism haplotype assembly problem[J].Brief Bioinform,2002,3(1):1-9.

[4] Wang R S,Wu L Y,Li Z P,et al.Haplotype reconstruction from SNP fragments by minimum error correction[J].Bioinformatics,2005,21(10):2456-2462.

[5] Sanger F,Nicklen S,Coulson A R.DNA sequencing with chain-terminating inhibitors[J].PNAS,1977,74(12):5463-5467.

[6] International Human Genome Sequencing Consortium.Initial sequencing and analysis of the human genome[J].Nature,2001,409(6822):860-921.

[7] Hinds D A.Stuve L L,Nilsen G B,et al.Whole-genome patterns of common DNA variation in three human populations[J].Science,2005,307(5712):1072-1079.

[8] Gabriel S B,Schaffner S F,Nguyen H,et al.The structure of haplotype blocks in the human genome[J].Science,2002,296(5576):2225-2229.

[9] Venter J C.The sequence of the human genome[J].Science,2001,291(5507):1304-51.

[10] Huson D H.Comparing assemblies using fragments and mate-pairs [C]//Gascuel O,Moret B M E.LNCS 2149:Proc of the 1st Int'l Workshop on Algorithms in Bioinformatics.Heidelberg:Springer,2001:294-306.

[11] Wernicke S.On the algorithmic tractability of Single Nucleotide Polymorphism (SNP)analysis and related problems[D].Germany:University of Tubingen,2003.

[12] Panconesi A,Sozio M.Fast hare:a fast heuristic for single individual SNP haplotype reconstruction[C]//Jonassen I,Kim J.LNCS 3240:Proc of the 4th Int'l Workshop on Algorithms in Bioinformatics.Heidelberg:Springer,2004:266-277.

[13] Myers G.A dataset generator for whole genome shotgun sequencing[C]//Lengauer T.Proc of the 7th Int'l Conf Intelligent Systems for Molecular Biology.California:AAAI Press,1999:202-210.

1.All experiments are repeated 100 times except for BNB_MEC with n=50 and m=100.The average time,the maximum running time and the reconstruction rate are over the repeated experiments with the same parameters.