文章编号:1001 - 2486(2002)06 - 0067 - 04

DNA 片段拼接中基于定长特征子串的重复序列信息屏蔽方法*

张博锋,王正华
(国防科技大学并行与分布处理国家重点实验室,湖南 长沙　410073)

摘　要:包含重复序列(repeats)的 DNA 序列的重构是大规模 DNA 片段拼接所面临的实际困难之一。在考虑片段数据所隐含的位置信息的基础上,提出了一种基于定长特征子串的屏蔽片段数据中重复序列信息的方法,即在进行序列相互比对前利用独特子串标识大多数片段,从而减少可能的错误重叠,讨论了方法中几个参数的确定问题并用计算结果说明了方法的有效性。

关键词:生物信息学;片段拼接;重复片段;屏蔽;定长特征子串
中图分类号:Q811.4　　文献标识码:A

Definite-sized Characteristic Substrings Based Method for the Masking-off of Repeats in DNA Fragment Assembly

ZHANG Bo-feng, WANG Zheng-hua
(National Laboratory for Parallel & Distributed Processing, National Univ. of Defense Technology, Changsha 410073, China)

Abstract: One of the practical difficulties that remains in large-scale DNA fragment assembly is the correct reconstruction of DNA sequences including repeats. An approach based on the definite-sized characteristic substring for the masking-off of repeats is proposed after considering the relative position information contained in fragment data. Before pair-wise alignment the approach chose unique substrings to mark fragments for the sake of decrease in possible incorrect overlaps. We also concretely describes the determination of some parameters and finally presents the computational result to prove the effectiveness of the method.

Key words: bioinformatics; fragment assembly; repeats; masking-off; definite-sized characteristic substring

在 DNA 片段拼接中,包含重复序列信息的片段(简称重复片段)的正确匹配是拼接 shotgun 序列的实质性困难之一。在人类基因组中有 50%以上是重复序列[1],由这样的基因组产生的片段数据若不经过重复序列信息的屏蔽的话,将会产生大量错误的重叠,最终导致结果的严重偏差。目前已经出现的很多用于 shotgun 片段拼接的工具(比较流行的有 Phrap、CAP3、TIGER、Celera assembler 等[3, 5, 6]),在处理重复片段时,都是采用对大量的片段数据进行反复叠代的方法,此间还需要加入很多人工的经验分析和干预[3, 7],一定程度上增加了拼接所花费的时间,降低了机器的使用效率。

本文针对上述问题,提出了一种基于定长特征子串的屏蔽重复序列信息的方法,在进行拼接前利用片段数据中隐含的位置标识性信息一次性屏蔽大多数重复片段间可能的错误的重叠,以减少拼接时可能出现的错误和反复处理的次数。

1　基于特征子串的重复片段屏蔽方法

DNA 序列和每一个片段序列都可以看做是字符集 {A, C, T, G} 上的字符串,每个长为 k 的字符串称为 k - 串;若它是某个片段(或序列)的一部分,则称它为此片段(或序列)的 k - 子串;当指定它为某个片段的标识性信息时,就是文中所说的特征子串。为了便于讨论,文中定义下列符号:k 为特征子串的定长,n 为要拼接片段的总数,T 为认定某一 k - 子串是 repeat 的阈值,L 为 DNA 原序列的长。另外我

* 收稿日期:2002 - 07 - 12
基金项目:国家自然科学基金资助重点项目(69933030)
作者简介:张博锋(1978—),男,硕士生。

们称 $\lg L$ 为测序规模。

1.1 基本思想和算法概要

实际上在很长的 DNA 原始序列中包含了许多分布较均匀的子串,它们往往只出现过一次或出现的频率很小,这种独特的子串是包含它的片段的标志性信息,往往可以在某种程度上暗示片段在整个 DNA 序列中的相互位置关系[3],只要考虑这种信息,对所有片段数据所含的所有长度为 k 的定长子串的出现次数进行统计,再利用统计结果,为每个片段指定它的若干子串(子串出现次数很小)作为其特征信息。那么两片段可能相邻的条件就是它们含有至少一个公共的特征子串,称之为可能相邻(PL)条件。只有满足了 PL 条件的片段才可能考虑去对它们进行比对,以根据它们的重叠结果确定它们是否真的相邻,描述如图 1。

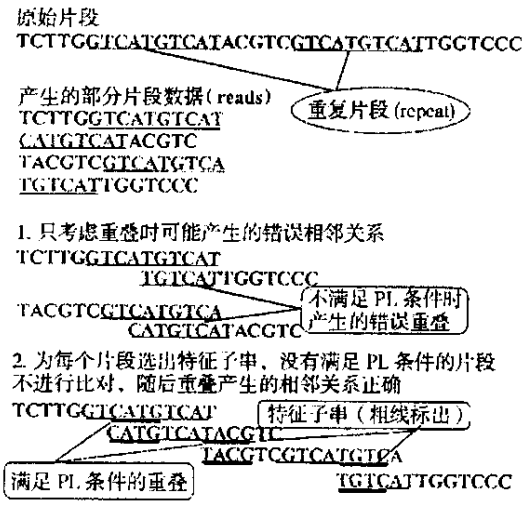

图 1 基于特征子串的 PL 条件示意图

Fig.1 PL condition based on characteristic substrings

在上述方法下,即使两个本不相邻的片段因为重复片段的原因存在很长的重叠,但只要它们的特征子串均不相同,处理时就不会对它们进行比对,也就不会认为它们是相邻的。这样就达到了'屏蔽'重复片段干扰的目的,也为后续的拼接产生了有用的依据。

基于特征子串的复片段屏蔽方法的算法概要如下:

step1 对所有的片段数据进行扫描,计算每一 k – 串 i 在所有片段中出现的次数 t_i,如果 t_i 大于某一阈值 T,则此子串标记为 repeat;

step2 对每一片段,根据其包含的每个 k – 子串 i 的出现总数和 repeat 信息为此片段挑选特征子串,选取原则是要求 i 未被记为 repeat, $t_i \geqslant 2$ 且在一定范围最小;

step3 对每一片段,如果其某个 k – 子串被挑选为其它片段的特征子串,则此 k – 子串也被挑选为自己的特征子串。

step4:根据 PL 条件执行拼接算法。

对算法有几点需要说明:

(1) 对于在原始序列中只出现一次的 k – 子串,它在片段数据中最多应出现 d 次,考虑到那些原序列中出现次数大于 1 但出现频率也很低的 k – 子串,我们取 T 为 d 的 1.5 倍。此参数可以根据实际情况进行调整。

(2) 实际测定出的片段数据总是有一定的差错,这样 m 过小可能会导致我们错过某些重叠信息,过大又会增加计算负担,为了平衡这一矛盾,适当增加 m 以保证一定的冗余度,取 $m = 3$。根据 m 的值,在 step2 中我们将片段平均分为三段,选取特征子串时要使 k – 子串尽量接近首段的首部、中段的中部、尾段的尾部。例如对首部的特征子串选取,它是首段中第一个出现次数最少且未被标定为 repeat 的 k – 子串。对其它段可以类似选取。

(3) 为了使 PL 条件最大限度保留正确的相邻信息,我们在 step3 中对片段数据进行了第三次扫描,以使更多的重叠关系反映在特征子串中。

1.2 对特征子串定长 k 的讨论

很明显,k 值越大,那么每个 k – 子串就越独特,但 k 也受到片段长度以及相邻片段间重叠区域的约束。由于 DNA 的序列中只由 A、C、G、T 四个字母组成,则长度为 k 的所有不同串就有 4^k 种,考虑到选取的特征 k – 子串的总数 s 不会超过这个数目,经过数学推导认为 k 要满足:

$$k \geqslant \log_4 n \tag{1}$$

从(1)式可以看出,特征子串的长度 k 是片段数目 n 的函数,片段数目越多,特征子串应取得越长。

另一方面如果考虑到一个 k – 子串在一个 DNA 序列中最多出现一两次的概率 P,我们就可以用 Poisson 分布估计出实际工程中 k 的一个上界。任意一个 k – 子串在长度为 L 的序列中出现 u 次的概率可以用下式估计[4] :

$$P(u) = \frac{\lambda^u e^{-\lambda}}{u!} \quad \left(\text{这里 } \lambda = \frac{L}{4^k}\right)$$

则

$$P = P(1) + P(2)$$

在实际的测序中 DNA 序列的全长总是有限的,取 L 的长度为现在人类基因组全长的 10 倍(3×10^{10}),我们计算了不同 k 值下 P 的值,见图 2。

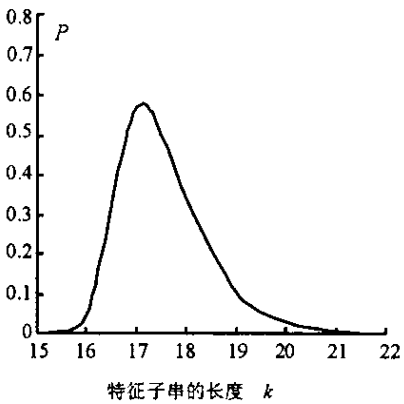

图 2 $L = 3 \times 10^{10}$时 P 与 k 的关系

Fig.2 Relation between P and k when $L = 3 \times 10^{10}$

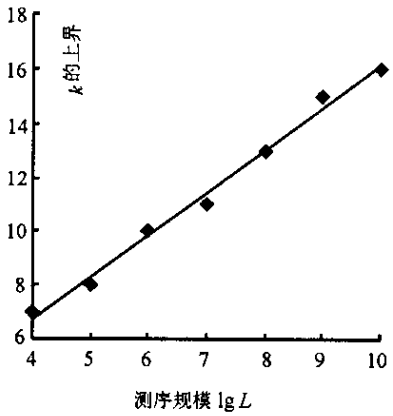

图 3 不同测序规模下 k 的上界

Fig.3 The upper bounds of k in different sequencing scale

从图 2 中可以看出,即使测序规模很大,当 $k = 17$ 时子串低次数出现的概率达到峰值,此时 k – 子串的特征最明显,区分能力最强。我们认为 17 就是此测序规模下 k 的上界。另外我们计算了在不同测序规模下 k 值的上界,发现它和测序规模有近似的线性关系,如图 3。这样,k 的选取只要满足(1)式且不大于图 3 中相应测序规模下相应的上界。

1.3 方法的计算复杂性

方法的主要过程是对所有片段和 k – 子串进行扫描或查找,经过估算可以得到其复杂性为 $O(n \cdot l \cdot 4^k)$,但考虑到最终 l 和 k 均有上界,故复杂性进一步简化为 $O(n)$,我们在后续的实验中也验证了这个结果。而引言中提到的工具的总复杂性均在 $O(n \log n)$以上[1,5~7],大部分是 $O(n^2)$,因而可考虑将此方法作为拼接程序对数据进行预处理和提供拼接依据的辅助手段,这里不具体展开讨论。

2 实验结果及分析

首先,对三条长度均在 200kbps 之上的人类基因序列(来自 EMBL 序列数据库,索取号分别为 :EMBL :AL390202、EMBL :AL359456、EMBL :AB019439),利用模拟的方法产生片段数据,然后用我们的方法进行处理(没有进行拼接),处理结果的统计见表 1。可以看到,有大量的子串被标记为 repeat,重复序列信息也就随着这些子串在 step2 中的‘落选’而被屏蔽,从而避免了包含它们的片段间的比对。另外在最坏的情形下,只有 8 条片段的信息被丢失,我们认为这些片段是重复序列的可能性很大,对拼接的结果可能会没有影响 ;只有 9.42% 的特征子串(无效特征子串)标识了不同位置上的片段,可能会产生错误重叠,但我们要求拼接时对具有相同标识的片段还要进行再比对以确认它们的相邻关系,所以,这部分错误在拼接时顺便可以排除。表 1 还表明每个片段上平均都有 5 个以上特征子串,这说明特征子串的标识能力很强。

表1 处理结果的统计

Tab.1 Statistic of the processing result

序列长	片段数	子串定长	特征子串数	标记为 repeat 的子串数	丢失片段	无效特征子串数(百分比)	每个片段的平均特征子串数
988 176	9512	10	19 269	224 124	1	1343 (6.97%)	5.6
593 964	5737	9	10 357	117 776	1	809 (7.80%)	5.4
200 000	1925	8	3024	34 959	8	285 (9.42%)	5.6

其次,我们统计了不同片段数目下我们的方法处理数据所用的时间,结果见图4。处理时间与片段数目有很好的线性关系,这正说明了1.3节的结论,即我们的方法的时间复杂性为 $O(n)$。

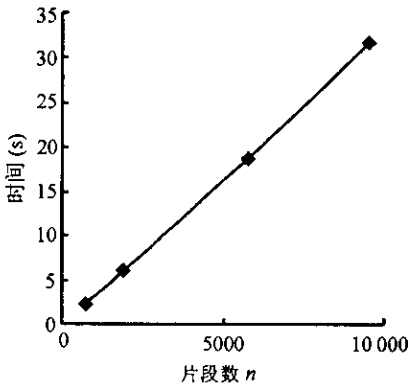

图4 处理时间与片段数目的关系

Fig.4 Relation between processing time and fragments number

3 结论

利用DNA片段的定长特征子串,可以以较少的处理代价很好地屏蔽片段数据中的重复序列信息,减少拼接时无谓的序列比对和重叠,提拼接算法的效率和准确性。

本文进一步的工作是研究基于特征子串的片段拼接算法及海量片段数据的大规模并行处理技术。

参考文献:

[1] International Human Genome Sequencing Consortium. Initial Sequencing and Analysis of the Human Genome [J]. Nature, 2001, 409: 860 – 864.

[2] Jain M, Myers E W. Algorithms for Computing and Integrating Physical Maps Using Unique Probes [J]. Journal of Computational Biology, 1997, 4(4): 449 – 466.

[3] Setuball J C, Werneck R F. A Program for Building Contig Scaffolds in Double-barrelled Shotgun Genome Sequencing [R]. Institute of Computing Technical Report IC-01-05, Unicamp, 2001.

[4] Lander E S, Waterman M S. Genomic Mapping by Fingerprinting Random Clones a Mathematical Analysis [J]. Genomics, 1998, 2: 231 – 239.

[5] Kececioglu J D, Meyers E W. Combinatorial Algorithms for DNA Sequence Assembly [J]. Algorithmica, 1995, 13: 7 – 15.

[6] Allex C F. Computational Methods for Fast and Accurate DNA Fragment Assembly[D]. A Dissertation for the Degree of Doctor of Philosophy (Computer Science) at the University of Wisconsin-Madison, 1999: 83 – 142.

[7] Pevzner P A, Tang Haixu, Waterman M S. An Eulerian Path Approach to DNA Fragment Assembly [J]. Proceedings of National Academy of Sciences, 2001, 98: 9487 – 9753.

(上接66页)

参考文献:

[1] ITU-T Recommendation H.324: Terminal for Low Bitrate Multimedia Communication [S]. 1996.

[2] ITU-T Recommendation H.263: Video Coding for Low Bitrate Communication [S]. March 1996.

[3] TMS320VC5509 Fixed-point Digital Signal Processor [R]. Texas Instruments Inc., April 2001.

[4] Liang J, Tran T D. Fast Multiplierless Approximation of the DCT with the Lifting Scheme [J]. IEEE Transaction on Signal Processing, 2001, 49(12): 3032 – 3044.