ISSN 1000-9825, CODEN RUXUEW
Journal of Software, Vol.17, No.2, February 2006, pp.193−199
DOI: 10.1360/jos170193

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel/Fax: +86-10-62562563

一种 DNA 测序纠错算法*

郑纬民+, 张 华, 王小川

(清华大学 计算机科学与技术系,北京 100084)

An Approach to Correcting DNA Sequencing Error

ZHENG Wei-Min+, ZHANG Hua, WANG Xiao-Chuan

(Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China)

+ Corresponding author: Phn: +86-10-62785592, E-mail: zwm-dcs@tsinghua.edu.cn, http://www.tsinghua.edu.cn

Zheng WM, Zhang H, Wang XC. An approach to correcting DNA sequencing error. *Journal of Software*, 2006,17(2):193−199. http://www.jos.org.cn/1000-9825/17/193.htm

Abstract: An error correcting algorithm is presented for detecting and correcting errors in the sequencing data before assembly process. The approach maps the sequencing data to an Euler superpath, and simplifies it dynamically by an equivalent transformation named Merging Transformation. In such a process, the algorithm isolates the right edges and error ones so that error paths are substituted and the corresponding errors in the sequencing data are corrected. In two test sets T.tengcongensis and T.whipplei, the algorithm has detected and corrected 86% and 83% errors on the "corrected" sequences respectively, compared with 71% and 53% errors using the original error correcting algorithm in the Eulerian path approach.

Key words: error correction; fragment assembly; DNA sequencing; Eulerian superpath; merging transformation

摘 要: 提出了一种新的测序纠错算法.该算法在对测序数据拼接之前对其进行检查,找出并修正测序序列中的错误.该算法将测序数据映射成欧拉超路,并通过一种称为合并变换的等价变换,通过一系列规则的限制和引导,动态地对欧拉超路进行简化.在此过程中,该算法将错误的边和正确的边对应起来,再通过替换纠错过程消除错误.在对 T.tengcongensis(TT)和 T.whipplei(TW)两个数据集的测试过程中,这种方法分别找出并修正了 86%和 83%的错误,而原欧拉序列拼接中的纠错算法对这两组数据集的纠错结果只有 71%和 53%.

关键词: 纠错;序列拼接;DNA 测序;欧拉超路;合并变换
中图法分类号: TP391 文献标识码: A

在大规模的 DNA 测序过程中,所获得的 read 数据不可避免地含有大量错误.如何在拼接过程中发现并且消除这些错误的影响,直接影响着拼接算法的可行性和最终结果的质量[1].

在 CAP3 Phrap 等基于"重叠检查-排列-连接"模式的软件中,采用打分矩阵的方式,对多序列间的不匹配进行扣分,并从中查找错误[2,3].这些算法的复杂度较高,对于大规模的基因数据,计算时间相当长,难以解决 repeat 问题.

* Supported by the National Natural Science Foundation of China under Grant No.60273007 (国家自然科学基金); the ChinaGrid (中国教育科研网格计划)

Received 2004-12-30; Accepted 2005-05-18

1995 年,Waterman,Myers 等人将测序数据转化为图问题进行处理,将对应序列相同位置的 read 映射到相同的边,代替了重叠检查[4,5].在此基础上,2001 年将构造和求解欧拉道路的方法引入到基因拼接中,较好地解决了拼接中的 repeats 问题[6−8].本文介绍了一组纠错算法 Euler-Correction,能够在进行构建 de Bruijn 图之前对测序数据进行纠错,有较高的纠错率.

Euler Correction 利用输入的一组 reads 数据之间的覆盖关系进行纠错.该方法输出也是一组 reads,比起输入的数据具有更低的错误率.2002 年和 2003 年推出的新的全基因组拼接软件 Arachne[9,10]也采用了类似的做法.但是,Euler Correction 方法存在以下缺陷,使得进一步提高纠错率存在一定的困难:

(a) 检测不到的错误:有时候,read 上的一个测序错误,使得包含错误后,前后相邻的局部序列,正好与其他片段上的子序列相同.这样,错误就隐藏在正确的序列中了.

(b) 无法纠正的错误:如果 read 中相邻位置连续出现多个错误,Euler Correction 方法虽然能够探测到这种错误,但在尝试修改一个碱基错误后,可能无法匹配到正确的碱基局部序列,就不能够进行纠错.

(c) 混合型错误:即一个 read 中出现多个错误(无法纠正),且落在其他正确的 read 上(无法检测).这种原始错误相对较少,但在 Euler Correction 中反而可能制造出这种错误来.

在本文中,我们将讨论并分析一种新的纠错算法——欧拉超路法.该方法能够识别和纠正以上几种错误类型,比 Euler-Correction 具有更高的纠错率.

1 欧拉超路纠错

1.1 问题定义

read:通过基因测序得到的小片断,每一个小片断可以看作一个字符串,由代表 DNA 这 4 种碱基的字母 A,T,G,C 构成.

l-tuple:将长度为 n 的 read 表达为 $n-l+1$ 个相互重叠的 *l-tuple* 的集合.*l-tuple* 为 read 上的 l 长的连续字符串,相邻的两个 *l-tuple* 所包含的字符只相差一个.例如:对 n=5 的 read“AGCCT”,取 l=3,那么,对应的 *l-tuple* 是 {AGC,GCC,CCT}.

de Bruijn 图 $G(S_l)$:设给定的一组测序结果的 read 集合为 S,设 $S=\{R_1,R_2,\ldots,R_n\}$,定义 S_l 为集合 S 的 read 对应的所有 *l-tuple* 的集合.定义图 $G(S_l)=(V,E)$.V 是图 $G(S_l)$的顶点集合,$V=S_{l-1}$.E 是图 $G(S_l)$的边集合,E 的定义如下:

$$E=\{e \mid e=(u,v), e\in S_l, u,v\in S_{l-1}\text{且}u\text{是}e\text{的前}l-1\text{个碱基组成}(l-1)\text{-}tuple,\ v\text{是}e\text{的后}l-1\text{个碱基组成}(l-1)\text{-}tuple\},$$

我们称 $G(S_l)$为测序结果对应的 de Bruijn 图.可以看出,在上述定义下,每一条 read 变成了 $G(S_l)$中的一条路径.

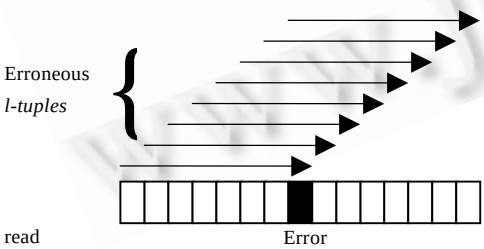


Fig.1 An error in a read affects l *l-tuple*

图 1 小片断中的一个错误将影响 l 个 *l-tuple*

欧拉超路(G,P):$G=G(S_l)$,P 是所有的 read 对应的路径集合,$P=\{P_1,P_2,\ldots,P_n\}$,P_i 是 *read* R_i 在 $G(S_l)$中对应的路径.

正确边/错误边:测序中会出现错误,例如碱基识别错误,或者插入/删除了一个碱基.那么,包含这种错误的 *l-tuple* 称为错误的 *l-tuple*,对应的边 e 称为错误的边;反之,不包含错误信息的边称为正确边.

一般来说,测序中的一个错误,通常会影响到 l 个 *l-tuple*,如图 1 所示.这 l 个 *l-tuple* 在图 $G(S_l)$中对应了一条都是由错误的边组成的路径.我们称这种路径中,起始的边为错误起始边,结束的边为错误结束边.如图 2 所示,e'为错误开始边,f'为错误结束边,e 和 f 分别是错误边 e'和 f'所对应的正确边.一般情况下,错误起始边前面的边为正确边,错误结束边的后续边为正确边.

测序纠错问题:给定欧拉超路(G,P),令 $E(P_i)=\{e|e$ 为错误边且 $e\in P_i\}$.测序纠错问题是找出所有的错误边 e,并将错误边用正确边 e'进行替换,得到一个新的欧拉超路(G',P'),使得 $\sum|E(P_i')|$ 最小.

1.2 动态分支构造

动态分支构造方法是对欧拉超路进行简化的过程,通过该过程不断进行边合并变换,最终的目标是把在同一条路径上的相连的正确边和错误边分别合并为一条路径.为保证这个过程的正确性,需要在合并过程中做一定的规则限制.在讨论规则限制之前,我们先引入几个定义:

定义 $P_{x,y}$ 为 P 中包含子路径(x,y)的路径集合;定义 $P_{a,x,y}$ 为 P 中包含子路径(a,x,y)的路径集合,以此类推.

定义 P_{x-}为 P 中以边 x 开头的路径集合;定义 P_{-y} 为 P 中以边 y 结束的路径集合.

定义 x-y 合并变换操作:新建一条边 z,按照如下情况对 P 中的路径进行处理(如图 3 所示):

(1) 用 z 代替 $P_{x,y}$ 中对应的子路径(x,y).

(2) 用 z 代替 P_{x-}中在路径开始的边 x.

(3) 用 z 代替 P_{-y} 中在路径结束的边 y.

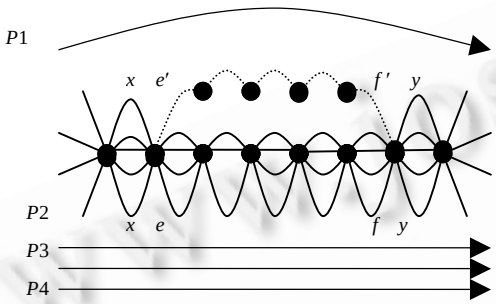


Fig.2 A path contains l error edges

图 2 路径中包含 l 条错误边

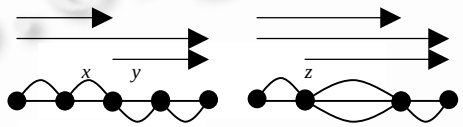


Fig.3 The combination of x-y is an equivalent transformation.

图 3 x-y 的合并是一种等价变换

在不考虑边 x 和边 y 是否为错误边的前提下,我们认为,只要不存在以下两种情况中的任何一种,那么 x-y 合并变换就是一种等价变换.

情况 1:存在边 a 和边 y',满足 $P_{a,x,y}$,$P_{a,x,y'}$和 P_{x-}这 3 个集合都不为空;

情况 2:存在边 x'和边 b,满足 $P_{x,y'b}$,$P_{x',y,b}$ 和 P_{-y} 这 3 个集合都不为空.

要注意这种变换和 detach[6,7]变换的区别,在 detach 变换中,为了保持变换为等价变换,需要对整条路径的一致性进行检查;而我们采用的变换只需要对相邻的边进行检查.

值得注意的是,如果 x-y 不满足合并限制,并不意味着边 x-y 之后不可能进行合并了,因为边 x-y 相邻的边被变换后,可能使得 x-y 满足合并限制并能够进行合并.如图 4 所示.

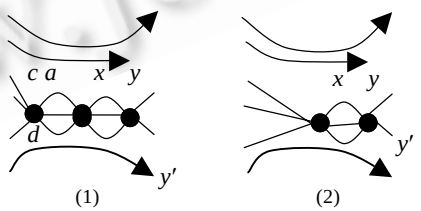


Fig.4 In graph (1), the edge x can not be combined with edge y, but after we combine (edge c, edge a) and (edge d , edge a), edge x can be combined with edge y in graph (2)

图 4 (1)中 x-y 为暂时性不可合并,但是在(1)中的 c-a 和 d-a 合并后,(2)中 x-y 可以进行合并变换

当考虑存在错误边的情况以后,问题变得复杂了,我们需要在合并过程中禁止路径中两类边合并变换:

(1) 正确边同错误边的合并.因为错误边同正确边相连,这条边是错误起始边或者错误结束边.这种合并将正确边合并进入了错误边,扩大了错误边的比重.

(2) 两条正确边之间合并,但其中一条正确边有对应的错误起始边或者错误结束边.当正确边被合并后,错误边失去了对应的正确边,使得无法完成之后的替换纠错过程.

在我们知道最终结果之前,并不能够准确地判断边的类型.这个时候,我们可以用一个近似的方法来识别边的类型.

定义边覆盖度 $m(e)$为边 e 在 P 中不同的路径上出现的次数.

测序中的一个错误将产生多条错误的边,这些边通常不出现在其他路径中,因此错误边的覆盖度一般比较低;同时,对于正确边来说,一般情况下会被多条路径包含,其覆盖度比较高.因此,我们可以设定一个适当的阈值 M,如果 $m(e)<M$,则认为 e 是错误边.若 $m(e)>=M$,则 e 是正确边.

对于(1)型禁止合并边的情况比较简单,假设要对边 x 和边 y 进行合并,那么直接检查这两条边的覆盖度 $m(x)$,$m(y)$和 M 的关系即可.

(2)型禁止合并的边的识别条件相对复杂一些,描述如下:

若 x,y 是正确边,且存在 $y'\in G$,满足$|P_{x,y}|\geqM$ 和$|P_{x,y'}|<M$,则 y'为错误结束边,y 为 y'对应的正确边,应禁止 x,y 合并;

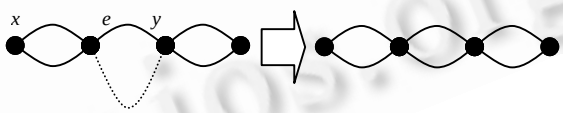
若 x,y 是正确边,且存在 $x'\in G$,满足$|P_{x,y}|\geq M$ 和$|P_{x',y}|<M$,则 x'为错误起始边,x 为 x'对应的正确边,应禁止 x,y 合并.

在判断两类不得合并的边时,都利用了边覆盖度 $m(e)$和阈值 M 的大小比较,可能造成误判.一种情况是,正确的边 e 因为测序数据的覆盖度不够,使得 $m(e)<M$,此时,只要适当调整 M,即可将此种判断错误的可能性降低;另一种情况是:有的错误边正好落在其他路径上,与正确边重合,此时 $m(e)>M$,并不能够通过调整 M 有效解决.此种情况称为错误边隐藏,采用 Euler Correction 完全无法解决这个问题,我们将在替换纠错过程中讨论这一问题的求解.

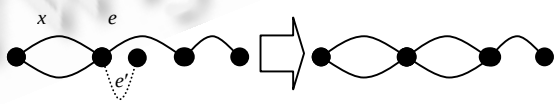
1.3 替换纠错过程

在理想情况下,按照边合并变换的限制规则进行动态分支构造,路径中相邻的正确边和错误边,都分别合成了一条边.对于变换后的欧拉超路中的错误边 e',有两种可能,我们分别对这两种情况进行纠错:

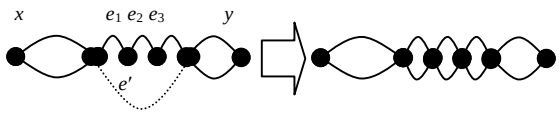
e'有前继和后续的边,且均为正确边.可以通过其前继和后续边,找到 e'对应的正确边,并进行替换.如图 5(a)所示,路径中包含 x-e'-y 子路径,x,y 均为正确边.这时找到一条边 e,使得$|P_{x,e,y}|$最大,如果$|P_{x,e,y}|>M$,那么我们认为 e 是 e'对应的正确边,并将 P 中的所有子路径(x,e',y)替换为(x,e,y).



(a) Ideal status 1
(a) 理想情况 1



(b) Ideal status 2
(b) 理想情况 1



(c) The correct edges haven't been combined together
(c) 纠错的边无法被合并

Fig.5 Three error correcting methods in different status

图 5 不同情况下的 3 种纠错方法

e'是路径的开始或者结束边,可以通过其前继或后续边,找到 e'对应的正确边,并进行替换.如图 5(b)所示,若 e'所在的路径以 x-e'结束,x 是正确边,则找到一条边 e 使得$|P_{x,e}|$最大,若$|P_{x,e}|>M$,则将 P 中的所有子路径(x,e')替换为(x,e);如果 e'所在的路径以 e'-y 开始,用类似的方法纠错.

实际情况下,因为相容性限制,或边类型识别错误,可能导致边的合并不充分或者过度合并.而且,因为一个 read 上邻近的测序错误,或者对应最终序列的 reads 的相似位置均有测序错误,将使得系统变得更加错综复杂.因而,对一条错误边,还有可能存在其他情况.

如图 5(c)所示,e'对应的正确边 e 并未合并成,而是以一条子路径$(e_i,e_{i+1},\ldots,e_{j-1},e_j)$的形式存在.此时不满足上面的两种情况,但是存在子路径$(e_i,e_{i+1},\ldots,e_{j-1},e_j)$,使得$|P_{x,e_i,e_{i+1},\ldots,e_{j-1},e_j}|>M$,并满足$(e_i,e_{i+1},\ldots,e_{j-1},e_j)$对应的 *l-tuple* 个数与 e'代表的 *l-tuple* 个数差别小于Δ(Δ表示 e'中包含的插入/删除性错误的限制),此时,可以将 P 中所有的子路径(x,e',y)替换为$(x,e_i,e_{i+1},\ldots,e_{j-1},e_j,y)$.

如图 6 所示,e'所在的路径上,前继边 x 或者后续边 y 应为错误边(设 y 边为错误边).但 y 正好与另一组不相关的路径上的边相同,系统认为是正确边.该问题为错误边隐藏问题,此时通常 $m(y)>M$.如果一个错误边不能够被上面 3 种纠错方法纠正,即可确定为边隐藏问题,此时无法知道是边 x 还是边 y 为隐藏错误边.处理的方法是:对 x-e'-y 做合并变换,此变换后得到的边 z 从这组不相关的路径上分离开来,从而使得错误边隐藏问题获解.

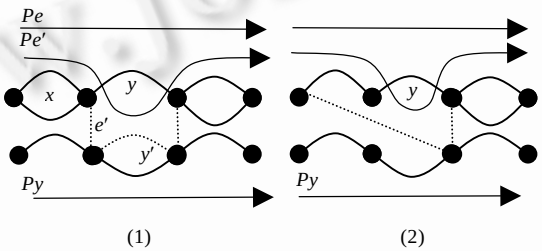


Fig.6 Solving error edge hidden problem through an combination transformation

图 6 通过合并变换解决错误边隐藏问题

1.4 欧拉超路法纠错的流程

(a) 对欧拉超路按照边类型限制进行合并变换.获得更加简化的欧拉超路,进行动态分支构造.

(b) 进行替换纠错.对欧拉超路中覆盖率低的边 e,若 $m(e)<M$,则视为错误边,尝试用上述纠错规则进行纠错.

经过上述两个步骤之后,图中的错误边减少,正确边增加.重复上述两个步骤,直到找不到更多错误或者无法再合并边的时候,算法终止.

2 结果分析

2.1 实验结果

为了测试本算法的正确性,不仅需要有测序获得的 read 数据,同时也需要有正确的 read 数据作对照,这种数据在互联网上是不公开的.我们采用的测试样本是 T. tengcongensis(TT)和 T.whipplei(TW)两种测序数据,前者是从华大基因中心获得的中国耐热菌的 DNA 测序数据;后者是欧拉算法的合作机构 Sanger 基因研究中心提供的真实数据.

取 *l-tuple* 的长度为 28,对数据用欧拉超路法纠错(下表中简称 EPC),得到的结果与 Euler Correction(表 1 中简称 EC)对比,如图 7 和表 1 所示.

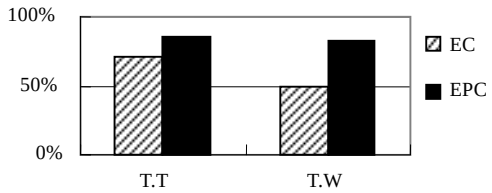


Fig.7 EPC error correcting rate and EC error correcting rate

图 7 欧拉超路法纠错率与 EC 方法纠错率对比

Table 1 Comparison of EPC and EC

表 **1** 欧拉超路法纠错和 Euler Correction 法纠错对比

	Total read length	Original error count	Error count after EC	Error count after EPC	Read count	EC error correcting rate (%)	EPC error correcting rate (%)
TT	1 610 878	5 509	1 599	760	2 942	71	86
TW	7 724 433	33 017	15 516	5 508	16 324	53	83

2.2 复杂度分析

我们假设有 n 条 read,其长度依次为 $L_1,L_2,\ldots,L_n$,设 L 为总长度,*l-tuple* 的总数为 T.根据前面讨论,我们可以得到结果:

$$T=\sum_{i=1}^{n}(L_i-l+1)=L-n*(l-1).$$

一般情况下,l 的取值为 20~30,可以认为 $T\approx L$.

对于每一次合并操作的边 x 和边 y,首先检查 x 和 y 是否是错误边,由于我们可以保存每一条边的覆盖度,因此这一步不需要时间.

假设 x 和 y 都是正确边,那么进行合并的条件检查,由于每一条路径对应一个 read,该检查最多涉及到 n 条路径,因此这一步判断的时间复杂度为 $O(n)$.

假设 x 和 y 包含错误边,那么按照有错误边的情况进行处理,由于纠错过程每一次判断最多涉及到 n 条路径,其时间复杂度也为 $O(n)$.

从算法流程上看,每进行一次合并操作可以减少一条边,因此,该算法最多进行 T 次合并操作.由于 $T\approx L$,因此算法的时间复杂度为 $O(n\times L)$.

程序采用十字链表方式保存图的结构,每一条边对应一个记录,该记录的首尾指针指向这一条边所在的路径上的前序和后继边,记录的上下指针指向和这条边对应的 *l-tuple* 相同的其他边.占用的空间和图中边的数目成正比,可以得到空间复杂度为 $O(L)$.

在全基因组序列拼接中,L 的数目比较大,很难把所有数据都放在一台机器内存中,可以考虑按照 read 分组,将不同组的 read 数据分布到不同机器上,保证有足够的内存并加快计算速度.

3 总 结

数据纠错是测序数据拼接中的一个重要环节.数据纠错的方式和效果直接影响着拼接算法的实现和复杂性.对于 Euler Correction 方法,因为将 reads 拆成 *l-tuple*,丢弃了 *l-tuple* 的前后关联信息,因而存在一些测序错误不能被发现或者无法纠正.

我们在欧拉超路上定义了测序纠错问题,保持了 *l-tuple* 前后相关信息的完备性.同时,将拼接工作中的欧拉超路 detach 变换的思路引入到纠错中,定义了称为合并变换的等价变换,能够不断对含有错误的欧拉超路进行简化.在边合并变换的限制规则引导下,动态分支构造过程将含有错误的边变换成几种可以识别的类型,并通过替换纠错过程,用正确的边代替错误的边,完成纠错.

欧拉超路法纠错克服了 Euler Correction 的不足.在欧拉超路的变换过程中,能够有效地分离和识别隐藏的错误,也能够较好地处理连续的多个错误.实验表明,这种方法在对真实数据的测试中优于 Euler Correction.由于欧拉超路法输入数据是含有测序错误的一组 read,输出是一组经过纠错的 read,因而可以作为独立的降低错误率的工具,与其他拼接软件一起使用.

致谢 我们感谢南加州大学助理项目科学家 Haixu Tang 博士后给我们的工作提出了许多建议,进行了反馈,并且提供给我们一些宝贵的数据处理工具.我们也感谢华大基因研究中心的生物信息部副主管李蔚博士后,和我们进行有价值的讨论,并提供了一手的测序数据.

References:

[1] Weber J, Myers G. Whole genome shotgun sequencing. Genome Research, 1997,5(7):401−409.

[2] Huang X, Madan A. CAP3: A DNA sequence assembly program. Genome Research, 1999,9(9):868−877.

[3] Green P. PHRAP documentation: ALGORITHMS. 1994. http://www.phrap.org

[4] Idury R, Waterman M. A new algorithm for DNA sequence assembly. Journal of Computational Biology, 1995,2(2):291−306.

[5] Boneld JK, Smith KF, Staden R. A new DNA sequence assembly program. Nucleic Acids Research, 1995,23(24):4992−4999.

[6] Pevzner PA, Tang H, Waterman MS. A new approach to fragment assembly in DNA sequencing. In: Proc. of the 5th Annual Int'l Conf. on Computational Molecular Biology. Montreal, 2001. 256−267.

[7] Pevzner PA, Tang H, Waterman MS. An Eulerian path approach to DNA fragment assembly. Proc. of the National Academy of Sciences of the USA, 2001,98(17):9748−9753.

[8] Pevzner PA, Tang H. Fragment assembly with double-barreled data. Bioinformatics, 2001,6(17):225−233.

[9] Jaffe DB, Butler J, Gnerre S, Mauceli E, Lindblad-Toh K, Mesirov JP, Zody MC, Lander ES. Whole-Genome sequence assembly for mammalian genomes: Arachne 2. Genome Research, 2003,1(13):91−96.

[10] Batzoglou S, Jaffe DB, Stanley K, Butler J, Gnerre S, Mauceli E, Berger B, Mesirov JP, Lander ES. ARACHNE: A whole-genome shotgun assembler. Genome Research, 2002,1(12):177−189.

郑纬民(1941－),男,浙江宁波人,教授,博士生导师,CCF 高级会员,主要研究领域为并行处理,分布式系统.

王小川(1978－),男,硕士,主要研究领域为生物信息学.

张华(1979－),男,硕士,主要研究领域为生物信息学.