哈尔滨工业大学

<<数据压缩>>

结课报告

(2020 年度春季学期)

姓名:	彭钰驯
学号:	1170300916
学院:	计算机科学与技术学院
教师:	刘岩

《基于自索引结构的高通量基因组重测序数据压缩算法》论文报告

一、研究背景

测序已经成为在生物研究中广泛应用的基本技术，获取不同生物体的基因遗传信息意义重大。高通量测序技术一次可以对几十万到几百万条核苷酸分子进行序列测定，是当前最为广泛应用的测序技术，其发展使经典分子生物科学家对基因组学的认识提升到一个新的水平，对临床和基因组学的研究产生了深远的影响。

由于测序技术的发展和测序成本的降低，海量的测序数据呈现爆炸式增长，已经远远超过了硬件性能的增长速度，因而对基因组数据的压缩有重大意义。

二、解决的问题

一方面，数据的采集是高度分散的，这需要很大的带宽来通过网络传输和访问这些大量的信息。这种情况下需要先进高效的大规模生物数据集的压缩方法，这不仅可以缓解存储需求，还可以促进这些数据的交换和传播。另一方面，压缩可用长期存储数据，无损压缩可以保证不会丢失实际信息，满足日后对该数据的使用需求。数据压缩这项工作至关重要，因为数据的存储和获取正在成为主要的瓶颈。测序数据有自身的特点和规律，分子信息之间有密切的联系，存在大量的信息冗余。这些数据特征也是进行数据压缩的基础。

在高通量测序技术中，DNA 测序是针对全基因组进行大规模测序，产生的数据量极其大，使用目前已有压缩算法的压缩效率有限。论文分析了索引技术在数据压缩中的应用，然后在此基础上，提出了一种基于自索引技术的压缩 SRVZip 压缩算法，并对应用 SRVZip 算法进行局部解压缩进行了详细说明。最后通过实验，实现自索引结构的局部随机解压缩算法。

三、详细描述

基于自索引结构的数据压缩算法 SRVZip。BWT（Burrows-Wheeler transfer）变换，被广泛应用于压缩技术中，当对一个字符串进行 BWT 变换的时候，可以对字符进行一次有规律的重排序，可以使字符中相同的字符串变成一些连续重复的字符，有利于提升压缩比。并且 BWT 操作数据变换的过程是可逆的，本身不会减少数据量，但是变换后的数据更容易压缩。FM-index 是构建在 BWT 的索引算法，是一种全文索引结构。它占用空间的大小依赖于建立索引的文本的可压缩性，易压缩的索引结构占用的空间小，不易压缩的索引结构占用的空间大。采用后缀数组数据结构，可促进压缩序列上搜索效率，同时 FM-index 对后缀数组也进行了压缩，使压缩后的空间占用降低，并且压缩以后的索引结构没有带来查询性能的明显下降。基于此类思想，文章采用 PBWT 算法。使用 PBWT 数据结构处理数据，一方面可以完整保存数据自身信息，同时也可以用于保存后一列数据的索引信息。采用类似 FMindex 的思路，引入 checkpoint，间隔一定区域存

储索引信息，以此达到对指定区域进行解压缩。

编码：

使用 BWT 算法进行编码的流程如下：

(1) 首先，BWT 先对需要转换的文本块，进行循环右移，每次循环一位。对长度为 n 的文本块，进行 n 次循环重复，这样就得到 n 个长度为 n 的字符串。

(2) 对循环移位后的 n 个字符串按照字典序排序。

(3) 记录下排序结果列中每个字符串的最后一个字符，组成了“L”列。

编码算法伪代码：

Algorithm1 PBWT BuildPrefixArray
算法 1：PBWT 构造前缀数组

输入：原始数组信息 y[k]
输出：构造完成的前缀数组 a[k]

1: $u = 0, v = 0$, *create empty arrys* $a[], b[]$
2: ***for*** $i = 0$ *to* $M - 1$ ***do***
3: ***if*** $y_i^k[k] = 0$ ***then***
4: $a[u] = a_k[i], u = u + 1$
5: ***else***
6: $b[v] = a_k[i], v = v + 1$
7: a_{k+1} = *the concatenation of a followed by b*
8: ***return*** a_{k+1}

Algorithm2　Dynamic PBWT BuildPrefixArray

算法 2：动态 PBWT 构造前缀数组

输入：原始数组信息$y[M]$，序列起始、终止位置信息$start[M]$ $end[M]$

输出：构造完成的前缀数组$a[M]$

1: *create arrys* $start[M], end[M]$ *to record N reads's start alignme*
   $u = 0, v = 0$ *, create empty arrys* $a[], b[]$
2: ***for*** $i = 0$ *to* $M - 1$ ***do***
3: ***if*** $y_i^k[k] = 0$ *and* $start[i] \le k \le end[i]$ ***then***
4: $a[u] = a_k[i], u = u + 1$
5: ***else***
6: $b[v] = a_k[i], v = v + 1$
7: a_{k+1} *= the concatenation of a followed by b*
7: **return** a_{k+1}

解码：

使用 BWT 算法进行解码的流程如下：

(1) 通过"F"列中的元素，找到他前面的字符，就是对应的同一行“L”列；

(2) 通过“L”列中的元素，找到他在“F”列中的对应字符位置。但是“L”中有 3 个字符 a，如何对应 F 中的 3 个 a 呢？因为 L 是 F 的前一个元素，多个具有相同前缀的字符串排序，去掉共同前缀后相对次序没有变化。所有遇到多个相同的字符，相对位置不变。

(3) 转到(1)，直到结束。

因为 F 列是已经排序的，可以从 L 列获得，所有只需要保存 L 列就可以。其中从 L 列中的字符获取在 F 列中的位置时需要：

(1) 前缀和数组，记录小于当前字符的字符数个数。

(2) count 计数，计算 L 中从开始位置到当前字符位置等于该字符的字符数。(保证多个相同字符下"L"到“F”的相对位置不变)。

解码算法伪代码：

Algorithm 3 LF-Mapping(r)

算法 3: LF-Mapping

输入：BWT 压缩后的字符串数组 BWT[]

输出：还原的字符串 T

```
1:  r = 1
2:  T = ""
3:  while BWT[r] ≠ $ do
4:      T = prepend BWT[r] to T
5:          c = BWT[r]
6:          r = C[c] + Occ[c, r] + 1
7:  end while
7:  return T
```

Algorithm4 Dynamic PBWT Decoding

算法 4： 动态 PBWT 解压缩

输入： BWT 压缩后的字符串数组 BWT[]

输出： 还原的字符串 T

1: *get* $start[M], end[M]$
2: $u = 0, v = 0, p = k + 1, q = k + 1$
3: *get create empty arrays* $a[], b[], d[], e[]$
4: ***for*** $i = 0$ *to* $M - 1$ ***do***
5: ***if*** *reads's* start array and end array between *process section then*
6: ***if*** $d_k[i] > p$ ***then*** $p = d_k[i]$
7: ***if*** $d_k[i] > q$ ***then*** $q = d_k[i]$
8: ***if*** $y_i[k] = 0$ ***then***
9: $a[u] = a_k[i], d[u] = p, u = u + 1, p = 0$
10: ***else***
11: $b[v] = a_k[i], e[v] = q, v = v + 1, q = 0$
12: $a_{k+1} =$ *the concatenation of a followed by b*
13: $d_{k+1} =$ *the concatenation of d followed by e*
14: **return** *the concatenation of* a_{k+1} *followed by* d_{k+1}

四、讨论，评价与创新点

高通量测序技术的发展，产生了急剧膨胀的数据，带来的巨大的存储和传输压力。DNA 数据压缩技术是缓解这一压力的有效途径。论文讨论了不同的测序数据以及对应的压缩方法，并给出了相应的理论分析，并提出了基于参考基因组的自索引压缩策略。

论文主要创新点如下：

（1）调研了高通量数据集的存储格式，以及现有的压缩算法。分析了测序数据的生物特性，同时通过分析表明，对质量分数的有损压缩，在提高压缩性能的同时，在下游分析中还能保持较好（有时甚至更优）的性能。

（2）在基于参考基因组进行差异化压缩编码的方案基础上，提出了 RVZip 压缩算法。该算法采用垂直方向的编码方式，同时对质量数采用稀疏化处理和均值处理相结合的方式，获得较好的有损压缩性能，并对压缩数据终端其他信息流采用统计分析的处理策略，有针对性地进行处理，实验表明压缩效果更优。

（3）针对数据需要随机解压缩和快速检索的需求，在分析自索引压缩技术原理的基础上，借鉴了 Bowite 中 FM-index 的设计思路，利用 PBWT 的自索引数据结构特性，提出 SRVZip 压缩算法，该算法是改进的动态 PBWT 算法，实现数据压缩与局部解压缩。实验表明，自索引技术的引入，在随机解压缩上有较好的性能。