On Compressibility of Protein Sequences

Donald Adjeroh and Fei Nan
Data Compression Conference, 2006. Proceedings. DCC
2006 pages 422-431

Presented by Chen LIN
Protein

- String of amino acids
- 20 types amino acids – symbols
  \( \log_2(20)=4.32 \text{ bps} \)
- Protein sequence – the entire collection of all the proteins in an organism
Content

- Why interest in compressing protein
- Why compression is difficult
- Long-Range Correlation
- Compressing Protein Sequences
Why interest?

- Exponential growth of biological sequences →
  Efficient storage and data transport

- Compression is a means for analyzing protein sequences
Why difficult?

- Difficult to model
  
  Due to the apparent randomness of symbols, model derived for proteins is different from models derived for text.

Classical compression methods – expansion.

“Protein is incompressible”

-- Nevill-Manning C.G & Witten I.H.
The IEEE Data Compression Conference, 257-299, 1999
In studying the SCP statistics of genomic sequences, the authors bumped unto an unusual observation:

an unprecedented redundancy in protein sequences – long range correlation
Sorted Common Prefix (SCP)

Using the relationship between the BWT and the suffix tree, we can obtain all the sorted suffixes of an input sequence in linear time.

A table of sorted suffixes

Example: Sequence $S = \text{“BRATATBAT…”}$

<table>
<thead>
<tr>
<th>Index</th>
<th>Sorted Suffixes</th>
<th>Sorted Common Prefix</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>SS1 = ATATBAT…</td>
<td>AT</td>
</tr>
<tr>
<td>2</td>
<td>SS2 = ATBAT…</td>
<td>AT</td>
</tr>
<tr>
<td>3</td>
<td>SS3 = AT…</td>
<td>AT</td>
</tr>
<tr>
<td>4</td>
<td>SS4 = BAT…</td>
<td>B</td>
</tr>
<tr>
<td>5</td>
<td>SS5 = BRATATBAT…</td>
<td>B</td>
</tr>
<tr>
<td>…</td>
<td>…</td>
<td>…</td>
</tr>
<tr>
<td>ith</td>
<td>SSi</td>
<td></td>
</tr>
<tr>
<td>jth</td>
<td>SSj</td>
<td></td>
</tr>
</tbody>
</table>
## Result of Sorted Common Prefix (SCP)

Kmax – the maximum common prefix for a given sequence

### Table 3: SCP Statistics for Concatenated Protein Sequences

<table>
<thead>
<tr>
<th>Seq</th>
<th>size, u</th>
<th># of genes</th>
<th>Kmax</th>
<th>Kmax/u</th>
<th>Start Index1</th>
<th>Start Index2</th>
<th>Diff</th>
</tr>
</thead>
<tbody>
<tr>
<td>HI</td>
<td>448770</td>
<td>1740</td>
<td>220685</td>
<td>0.492</td>
<td>53200</td>
<td>8</td>
<td>53192</td>
</tr>
<tr>
<td>MJ</td>
<td>509508</td>
<td>1680</td>
<td>343105</td>
<td>0.673</td>
<td>34899</td>
<td>3</td>
<td>34896</td>
</tr>
<tr>
<td>SC</td>
<td>2900346</td>
<td>8220</td>
<td>886531</td>
<td>0.306</td>
<td>480296</td>
<td>29</td>
<td>480267</td>
</tr>
<tr>
<td>HS</td>
<td>3295749</td>
<td>5733</td>
<td>392004</td>
<td>0.119</td>
<td>358676</td>
<td>24</td>
<td>358652</td>
</tr>
</tbody>
</table>

HI: *H. influenzae*; MJ: *M. jannaschii*; HS: *H. sapiens*; SC: *S. cerevisiae*

Diff = |Index1 - Index2|
Long-Range Correlation

- Unusually high values of $K_{max}$
- Long range of separation between the repeats (common prefix)
  
  \[ \text{separation} > 350,000 \text{ protein symbols} \]

- $K_{max} > |\text{Index2} - \text{Index1}|$

![Protein Sequence Diagram]

Start Index 2

Start Index 1

Repeat 1

Repeat 2

Protein Sequence
Compressing Protein Sequences

A dictionary-based compression:

1) remove the repeated substring from the input sequence, and move it to an external dictionary.
   D + R

2) In the dictionary, record the positions in the sequence where each repetition occurred, along with the repetition type.

3) Pass D and R to the core algorithm again until no compression could be achieved
Flowchart of proposed compression algorithm
Feed dictionary entries $D$ & remaining sequence $R$ for further decomposition and Parsing

Stops when the compression gain $G(S)$ is negative or less than a threshold.

Each leaf in the tree is part of the final output

Fig.2: Multilevel hierarchical decomposition
Example of Parsing and Encoding

Sample sequence $S$:

\[
\begin{array}{cccccccc}
\downarrow & & & & & & & \\
P_1 & P_2 & P_3 & P_4 & P_5 & P_6 & P_7 \\
\hline
x_1 MMCTGTGTCMM & x_2 MM & x_3 GTCMM & x_4 TG & x_5 MMCTG & x_6 TTGMCMMGT & x_7 MM
\end{array}
\]

Remaining parsed sequence:

\[\text{Parse}(S) : x_1 x_2 x_3 x_4 x_5 x_6 x_7\]
# Example -- Dictionary

<table>
<thead>
<tr>
<th>index</th>
<th>Repeat Pattern</th>
<th>$l(r)$</th>
<th>$t(r)$</th>
<th>$\eta(r)$</th>
<th>positions</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>MMCTGTCMM</td>
<td>9</td>
<td>1</td>
<td>1</td>
<td>$p_1$, $p_6$</td>
</tr>
<tr>
<td>2</td>
<td>GTCMM</td>
<td>5</td>
<td>1</td>
<td>1</td>
<td>$p_3$, $p_5$</td>
</tr>
<tr>
<td>3</td>
<td>MM</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>$p_2$, $p_7$</td>
</tr>
</tbody>
</table>

**Notation:**

- $r$ = a repetition pattern;
- $l(r) = |r| = \text{length of } r$;
- $\eta(r) = \text{total number of occurrences of } r$;
- $t(r) = \text{repetition type for current occurrence of } r$. (1: direct repeat; 2: reverse repeat; 3: complementary palindrome)
### Results

#### Table 4: Statistics of maximal repeats and compression results using the observed long-range correlations.

<table>
<thead>
<tr>
<th>Size</th>
<th>Kmax</th>
<th>repeat length ( l(r) )</th>
<th>number of occurrence ( \eta(r) )</th>
<th>Compression Results (bps)</th>
</tr>
</thead>
<tbody>
<tr>
<td>HI</td>
<td>509508</td>
<td>34896</td>
<td>7</td>
<td>2.546</td>
</tr>
<tr>
<td>MJ</td>
<td>448770</td>
<td>53192</td>
<td>5</td>
<td>2.273</td>
</tr>
<tr>
<td>SC</td>
<td>2900346</td>
<td>406239</td>
<td>3</td>
<td>3.111</td>
</tr>
<tr>
<td>HS</td>
<td>3295749</td>
<td>338359</td>
<td>3</td>
<td>3.435</td>
</tr>
</tbody>
</table>

20 symbols \( \Rightarrow \) \( \log_2(20) = 4.32 \text{ bps} \)

#### Table 5: Comparative compression performance using the observed long-range correlations. Compression results in bits/symbol (smaller values imply better performance).

<table>
<thead>
<tr>
<th></th>
<th>Gen Compress</th>
<th>CP(0)</th>
<th>CP(1)</th>
<th>CP(2)</th>
<th>CP(3)</th>
<th>IzaCTW (8)</th>
<th>Block Code</th>
<th>Proposed Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>HI</td>
<td>4.156</td>
<td>4.156</td>
<td>4.149</td>
<td>4.146</td>
<td>4.143</td>
<td>4.118</td>
<td>3.665</td>
<td>2.546</td>
</tr>
</tbody>
</table>
Conclusion

- Identify the correlated protein sequences based on the sorted common prefix
- Using a dictionary-based parsing and encoding scheme to provide compression
- Can provide consistent compression, at times down to less than 2.3 bps