*

Greedy Algorithms for the Shortest Common Superstring That Are Asymptotically Optimal

ALAN FRIEZE[†]
Department of Mathematics
Carnegie Mellon University
Pittsburgh, PA 15213
U.S.A.

WOJCIECH SZPANKOWSKI [‡]
Department of Computer Science
Purdue University
W. Lafayette, IN 47907
U.S.A.

June 17, 1997

[*]To appear in *Algorithmica*

[†]This work was supported by NSF Grant CCR-9225008.

[‡]Research supported by NSF Grants NCR-9206315, NCR-9415491, and CCR-9201078, and in part by NATO Collaborative Grants 0057/89 and CGR.950060.

Outline of the Talk

1. Problem Formulation and Its Applications
2. Definitions
3. Generic Greedy Algorithm
4. Main Results
5. Remarks and Extensions
6. Sketch of the Proof:
   - Upper Bound
   - Lower Bound
7. Open Problems

Problem Formulation

Given a collection of strings, say $\mathbf{x}^1, \mathbf{x}^2, \ldots, \mathbf{x}^n$ over an alphabet Σ, find the shortest string $\mathbf{z}$ such that each of $\mathbf{x}^i$ appears as a substring (a consecutive block) of $\mathbf{z}$.

EXAMPLE. We show below a common superstring of built from the following strings: $\mathbf{a}^1 = abaaab$, $\mathbf{a}^2 = aabaaaa$, $\mathbf{a}^3 = aababb$, $\mathbf{a}^4 = bbaaba$, and $\mathbf{a}^5 = bbbb$.

Graph Representation

It is know to be NP-hard since it can be reduced to finding a Hamiltonian cycle in a complete digraph.

EXAMPLE: Let us consider $\mathbf{a}^1 = abaaab$, $\mathbf{a}^2 = aabaaaa$, $\mathbf{a}^3 = aababb$, $\mathbf{a}^4 = bbaaba$, and $\mathbf{a}^5 = bbbb$, and we build a matrix of the C_{ij} that represents the longest suffix of $\mathbf{a}^i$ that is equal to a prefix of $\mathbf{a}^j$.

$$\mathbf{C} = \begin{bmatrix} \star & 3 & 3 & 1 & 1 \\ 1 & \star & 2 & 0 & 0 \\ 0 & 0 & \star & 2 & 2 \\ 3 & 4 & 4 & \star & 0 \\ 0 & 0 & 0 & 2 & \star \end{bmatrix}$$

Probabilistic Model

Worst-case analysis has resulted in several heuristics giving constant worst-case performance guarantees:
Blum,Jiang,Li,Tromp and Yannakakis;
Kosaraju,Park and Stein;
Teng and Yao;
Armen and Stein.

Here we concentrate on the PROBABILISTIC ANALYSIS of simple GREEDY heuristics. We will assume that the input strings are independently generated.

In the **Bernoulli model** each $\mathbf{x} = \mathbf{x}^j = x_1 x_2 \ldots x_\ell$ is of the same length ℓ and x_i is generated independently of $x_1, x_2, \ldots, x_{i-1}$. Furthermore, $\mathbf{P}(x_i = \omega_j \in \Sigma) = p_j > 0$ for $1 \leq j \leq M$. Let

$$H = -\sum_{i=1}^{m} p_i \log p_i$$

be the associated entropy.

Mixing Model

(M) Mixing Model

Let $\mathcal{F}_i^j$ be a σ-field generated by $\mathbf{x}_{k=i}^j$ for $i \le j$. There exists a function $\alpha(\cdot)$ of g such that: (i) $\lim_{g\to\infty} \alpha(g) = 0$, (ii) $\alpha(1) < 1$, and (iii) for any m, and two events $A \in \mathcal{F}_{-\infty}^i$ and $B \in \mathcal{F}_{i+g}^{\infty}$ the following holds

$$(1-\alpha(g))\mathbf{P}(A)\mathbf{P}(B) \le \mathbf{P}(AB) \le (1+\alpha(g))\mathbf{P}(A)\mathbf{P}(B).$$

Then entropy H becomes

$$H = \lim_{k\to\infty} -\frac{\mathbf{E}\log \mathbf{P}(\mathbf{x}_1^k)}{k}.$$

Let also h_2 be defined as

$$h_2 = \lim_{k\to\infty} \frac{\log(\mathbf{E}\{\mathbf{P}(\mathbf{x}_1^k)\})^{-1}}{k} = -\lim_{k\to\infty} \frac{\log\left(\sum_{\mathbf{x}_1^k \in \Sigma^k} \mathbf{P}^2(\mathbf{x}_1^k)\right)}{k}$$

which can be proved to exists (h_2 is related to Rènyi second order entropy).

Definitions

Suppose $\mathbf{x} = x_1x_2 \ldots x_r$ and $\mathbf{y} = y_1y_2 \ldots y_s$ are strings over the same finite alphabet $\Sigma = \{\omega_1, \omega_2, \ldots, \omega_M\}$ where $M = |\Sigma|$ is the size of the alphabet.

We define their *overlap* $o(\mathbf{x}, \mathbf{y})$ by

$$o(\mathbf{x}, \mathbf{y}) = \max\{j : \ y_i = x_{r-i+1},\ 1 \leq i \leq j\}.$$

EXAMPLE: if $\mathbf{x} = ababb$ and $\mathbf{y} = abbbbbbbb$, then $\o(\mathbf{x}, \mathbf{y}) = 3$

If $\mathbf{x} \neq \mathbf{y}$ and $k = o(\mathbf{x}, \mathbf{y})$, then

$$\mathbf{x} \oplus \mathbf{y} = x_1x_2 \ldots x_r y_{k+1} y_{k+2} \ldots y_s.$$

EXAMPLE: If $\mathbf{x}$ and $\mathbf{y}$ are defined as above, then $\mathbf{x} \oplus \mathbf{y} = ababbbbbbb$.

Generic Greedy Algorithm

1. $I \leftarrow \{\mathbf{x}^1, \mathbf{x}^2, \mathbf{x}^3, \ldots, \mathbf{x}^n\}$; $O_n^{\mathrm{gr}} \leftarrow 0$;

2. **repeat**

3. choose $\mathbf{x}, \mathbf{y} \in I$; $\mathbf{z} = \mathbf{x} \oplus \mathbf{y}$;

4. $I \leftarrow (I \setminus \{\mathbf{x}, \mathbf{y}\}) \cup \{\mathbf{z}\}$;

5. $O_n^{\mathrm{gr}} \leftarrow O_n^{\mathrm{gr}} + o(\mathbf{x}, \mathbf{y})$;

6. **until** $|I| = 1$

Greedy Algorithms – RGREEDY

RGREEDY: In Step 3, $\mathbf{x}$ is the string $\mathbf{z}$ produced in the previous iteration, while $\mathbf{y}$ is chosen in order to maximise $o(\mathbf{x}, \mathbf{y}) = o(\mathbf{z}, \mathbf{y})$. Our initial choice for $\mathbf{x}$ is $\mathbf{x}^1$. Thus, in RGREEDY we have one "long" string $\mathbf{z}$ which grows by addition of strings at the *right hand end*.

More precisely: Let D be the digraph $([n], A)$ with edge weights $w_{i,j} = o(\mathbf{b}^i, \mathbf{a}^j)$ for $i, j \in [n]$.

Algorithm:
1. Start with any vertex of D and move along an edge with largest weight as long as there is no cycle.

2. Finish when all nodes are visited.

Greedy Algorithms – GREEDY

GREEDY: In Step 3, choose $\mathbf{x} \neq \mathbf{y}$ in order to maximise $o(\mathbf{x}, \mathbf{y})$.

GREEDY: Sort the edges A into $e_1, e_2, \ldots, e_N$, $N = n^2$ so that $w(e_i) \geq w(e_{i+1})$; $S_G \leftarrow \emptyset$;

For $i = 1$ **to** N **do**:
if $S_G \cup \{e_i\}$ contains in D neither
(i) a vertex of outdegree or in–degree at least 2 in S_G,
(ii) a directed cycle,
then $S_G \leftarrow S_G \cup \{e_i\}$.

On termination S_G contains the $n-1$ edges of a Hamilton path of D and corresponds to a superstring of $\mathbf{x}^1, \mathbf{x}^2, \ldots, \mathbf{x}^n$. The selection of an edge weight $(\mathbf{b}^i, \mathbf{a}^j)$ corresponds to overlapping $\mathbf{x}^i$ to the left of $\mathbf{x}^j$.

Greedy Algorithm – MGREEDY

MGREEDY: In Step 3 choose $\mathbf{x}, \mathbf{y}$ in order to maximise $o(\mathbf{x}, \mathbf{y})$. If $\mathbf{x} \neq \mathbf{y}$ proceed as in GREEDY. If $\mathbf{x} = \mathbf{y}$, then $I \leftarrow I \setminus \{\mathbf{x}\}$, O_n^{gr} is not incremented, and $\mathcal{C} \leftarrow \mathcal{C} \cup \{\mathbf{x}\}$ where the set $\mathcal{C}$ is initially empty. On termination we add the final string left in I to $\mathcal{C}$.

MGREEDY: sort the edges A into $e_1, e_2, \ldots, e_N$, $N = n^2$ so that $w(e_i) \geq w(e_{i+1})$; $S_{MG}, C \leftarrow \emptyset$;

For $i = 1$ **to** N **do**:
if $S_{MG} \cup \{e_i\}$ contains no vertex of outdegree or indegree at least 2 in S_{MG},
then $S_{MG} \leftarrow S_{MG} \cup \{e_i\}$.
If e_i closes a cycle, **then** $C \leftarrow C \cup \{e_i\}$.

On termination the edges of S_{MG} form a collection of vertex disjoint cycles $C_1, C_2, \ldots, C_t$, $t = |C|$ which cover $[n]$. Each C_j contains one edge f_j which is a member of C and f_j is a lowest weight edge of C_j. Let $P_j = C_j - f_j$. The catenation of paths $P_1, P_2, \ldots, P_t$ define a superstring of the input.

Main Result: Bernoulli Model

Theorem 1. *Consider the Shortest Common Superstring problem under the Bernoulli model. Let* $P = \sum_{j=1}^{M} p_i^2$*. Then,* **whp**

$$\lim_{n\to\infty} \frac{O_n^{\mathrm{opt}}}{n \log n} = \frac{1}{H} \qquad\qquad \lim_{n\to\infty} \frac{O_n^{\mathrm{gr}}}{n \log n} = \frac{1}{H} \qquad (1)$$

provided

$$|\mathbf{x}^i| > -\frac{4}{\log P} \log n$$

for all $1 \leq i \leq n$.

Remark: The optimal part of the above theorem was proved by Alexander (1995) by a different method.

Main Result: Mixing Model

Theorem 2. *Consider the Shortest Common Superstring problem under the mixing model (M). Then, with high probability (***whp***)*

$$\lim_{n\to\infty} \frac{O_n^{\mathrm{opt}}}{n \log n} = \frac{1}{H} \qquad\qquad \lim_{n\to\infty} \frac{O_n^{\mathrm{gr}}}{n \log n} = \frac{1}{H}$$

provided

$$|\mathbf{x}^i| > \frac{4}{h_2} \log n$$

for all $1 \le i \le n$.

Remarks

We first proved Theorem 1 only for the algorithm RGREEDY. Subsequently, Yang and Zhang extended it to include MGREEDY. Then we gave a shorter proof of this along with a proof for GREEDY as well.

Not Equal Length Strings. The assumption regarding equal length strings is not relevant as long as there are enough long strings satisfying our condition. A precise formulation of the proportion of short and long strings such that Theorem 1 still holds can be found in Alexander (1995).

Markovian Model. The sequence $\mathbf{x} = \mathbf{x}^j$ $(1 \leq j \leq n)$ forms a stationary Markov chain with $p_{i,j} = \mathbf{P}\{x_{k+1} = j \in \Sigma | x_k = i \in \Sigma\}$. The entropy $H = -\sum_{i,j=1}^{M} \pi_i p_{i,j} \log p_{i,j}$ where π_i is the stationary distribution of the Markov chain. The quantity h_2 is a little harder to compute but it can be shown that

$$h_2 = -\log \theta$$

where θ is the largest eigenvalue of the Schur product of the transition matrix of the underlying Markov chain with itself (that is, element-wise product).

Remarks

SCS Does Not Compress Optimally. The SCS can be used to compress strings by storing n pointers indicating the beginning of the original strings plus lengths of all strings. But, this does NOT provide optimal compression.

The compression ratio C_n is

$$C_n = \frac{n\ell - \frac{1}{H} n \log n + n \log_2 (n\ell - \frac{1}{H} n \log n)}{n\ell}$$

where the first term of the numerator represents the length of the shortest superstring and the second term corresponds to the number of bits needed to encode the pointers. Observe that when the length of a string ℓ grows faster than $logn$, then $C_n \to 1$ (i.e., no compression). When $\ell = O(\log n)$ some compression might take place.

The fact that SCS does not compress well is hardly surprising: In the construction of SCS we do not use all available redundancy of all strings but only that contained in suffixes/prefixes of the original strings.

Extensions

Approximate SCS. Let us define a distance between two strings, say $\mathbf{x}$ and $\mathbf{y}$ as the relative Hamming distance. For a given $D < 1$, we introduce an APPROXIMATE SCS as follows:

Construct the shortest common superstring of strings $\mathbf{x}^1, \mathbf{x}^2, \ldots, \mathbf{x}^n$ *such that every string* $\mathbf{x}^i$ *is within Hamming distance* D *of a substring of the superstring.*
(Of course, a restriction on D is necessary since for too large D any two randomly chosen strings are within distance D.)

We CONJECTURE that also for the Approximate SCS the optimal and greedy overlaps are asymptotically equivalent. However, the constant in front of $n \log n$ is not any longer the entropy H. Recently, Yang and Zhang (1996) proved that this constant is the reverse of the so called LOWER MUTUAL INFORMATION, provided the lengths of the strings are not too short (i.e., $\ell > \frac{4}{r_1(D)} \log n$, where $r_1(D)$ the so called second generalized Rényi's entropy introduced recently by Łuczak and Szpankowski (IT, 1997).

Limiting Distribution?

Limiting Distribution ?. Theorem 1 presents only a convergence in probability, and might be insufficient for some applications. We, therefore, conjecture that a stronger result is also true, namely, the central limit theorem. We claim that

$$\mathbf{Var}\ O_n^{\text{opt}} \sim \mathbf{Var}\ O_n^{\text{gr}} \sim \frac{h_2 - H^2}{H^3} n \log n + O(n)$$

where $h_2 = \sum_{i=1}^{M} p_i \log^2 p_i$, and more importantly

$$\frac{O_n^{\text{opt}} - \mathbf{E} O_n^{\text{opt}}}{\sqrt{\mathbf{Var}\ O_n^{\text{opt}}}} \sim \frac{O_n^{\text{gr}} - \mathbf{E} O_n^{\text{gr}}}{\sqrt{\mathbf{Var}\ O_n^{\text{gr}}}} \to N(0,1)$$

where $N(0,1)$ is the standard normal distribution.

Proof: Upper Bound

Define
C_{ij} – the length of the longest suffix of $\mathbf{x}^i$ that is equal to the prefix of $\mathbf{x}^j$. Let

$$\begin{aligned} M_n(i) &= \max_{1\le j\le n, j\ne i}\{C_{ij}\} , \\ H_n &= \max_{1\le i\le n}\{M_n(i)\} = \max_{1\le i<j\le n}\{C_{ij}\} . \end{aligned}$$

UPPER BOUND:

$$O_n^{\mathrm{opt}} \le \sum_{i=1}^{n} M_n(i) .$$

RESTRICTION OF LENGTHS:
Let $\mathcal{E}$ denote the event that there is no such a pair, say i, j, that $o(\mathbf{x}^i, \mathbf{x}^j) \ge \ell/2$ (so no two strings overlap more than half). Observe that $\mathcal{E}$ is equivalent to postulate that

$$H_n \le \ell/2$$

.

Upper Bound

Lemma 1. (i) *In the mixing model, for any* $\varepsilon > 0$

$$\lim_{n\to\infty} \mathbf{P}\left((1-\varepsilon)\frac{1}{H}\log n \le M_n \le (1+\varepsilon)\frac{1}{H}\log n\right) = 1 - O(1/n^{\varepsilon})$$

provided $\alpha(g) \to 0$ *as* $g \to \infty$. *Furthermore, for almost all strings that are sufficiently long all but* εn *of the numbers* $M_n/\log n$ *are within* ε *of* $1/H$.

(ii) *In the mixing model, for any* $\varepsilon > 0$

$$\lim_{n\to\infty} \mathbf{P}\left((1-\varepsilon)\frac{2}{h_2}\log n \le H_n \le (1+\varepsilon)\frac{2}{h_2}\log n\right) = 1 - O(1/n^{\varepsilon})$$

provided $\alpha(g) \to 0$ *as* $g \to \infty$. *If, in addition, the mixing coefficients are summable, that is,* $\sum_g \alpha(g) < \infty$, *then*

$$\lim_{n\to\infty} \frac{H_n}{\log n} = \frac{2}{h_2} \qquad (a.s.)\ .$$

(iii) *In the Bernoulli model (also in the Markovian model), for large n we have*

$$\begin{aligned}\mathbf{E}M_n &= \frac{1}{H}\log n + \frac{\gamma}{H} + \frac{h_2}{2H^2} - P_1(\log n) + O(1/n) \\ \mathbf{Var}\ M_n &= \frac{h_2 - H^2}{H^3}\log n + C + P_2(\log n) + O(1/n)\end{aligned}$$

where C is a constant, $h_2 = \sum_{i=1}^{M} p_i \log^2 p_i$, $\gamma = 0.577\ldots$ is the Euler constant, $P_1(x)$ and $P_2(x)$ are fluctuating function with small amplitude. Furthermore, the following is true for an asymmetric Bernoulli model (i.e., probabilities of symbol generations are not the same)

$$\frac{M_n - \mathbf{E}M_n}{\sqrt{\mathbf{Var}\ M_n}} \xrightarrow{d} N(0,1)$$

where $N(0,1)$ is the standard normal distribution. The rate of convergence is $O(1/\sqrt{\log n})$, and the convergence also holds in moments.

Sketch of Proof for the Lemma

PROOF FOR M_n. Let w_k be a typical string (so that $e^{-kH(1+\varepsilon)} \leq \mathbf{P}(w_k) \leq e^{-kH(1-\varepsilon)}$). The:

$$\mathbf{P}(M_n \geq k) \leq n\mathbf{P}(w_k) + \mathbf{P}(\mathcal{B}_k)$$

where $\mathbf{P}(\mathcal{B}_k)$ is the probability of the set $\mathcal{B}_k$ of all non-typical sequences. The result follows immediately after substituting $k = (1+\varepsilon)H^{-1}\log n$.

For a LOWER BOUND, let w_k be any fixed typical sequence with $k = \frac{1}{H}(1-\varepsilon)\log n$. Define Z_k as the number of strings $j \neq i$ such that a prefix of length k is equal to w_k *and* a suffix of length k of the ith string is equal to the typical sequence w_k. Since w_k is fixed, the random variables C_{ij} are independent, and hence by the *second moment method* or Chebyshev's inequality we have

$$\mathbf{P}(M_n < k) = \mathbf{P}(Z_k = 0) \leq \frac{\mathbf{Var}Z_k}{(\mathbf{E}Z_k)^2} \leq \frac{1}{n\mathbf{P}(w_k)} = O(n^{-\varepsilon^2}),$$

since $\mathbf{Var}Z_k \leq nP(w_k)$.

Proof of Lemma

PROOF FOR H_n. For an upper bound, we only need

$$\mathbf{P}(H_n > k) \le n^2 \sum_{w_k \in \Sigma^k} \mathbf{P}^2(w_k)$$

where $w_k \in \Sigma^k$ denotes a fixed string of length k. An upper bound follows immediately from the definition of h_2 after substituting $k = (1+\varepsilon)\frac{2}{h_2}\log n$.

For a LOWER BOUND, we again apply the second moment method (however, expressed slightly differently). Let $A_{ij} = \{C_{ij} > k\}$ for some $k = (1-\varepsilon)\frac{2}{h_2}\log n$.

$$\begin{aligned}\mathbf{P}(H_n > k) &= \mathbf{P}\left(\bigcup_{i,j=1}^{n} A_{ij}\right) \\ &\ge \frac{\left(\sum_{i,j}\mathbf{P}(A_{ij})\right)^2}{\sum_{i,j}\mathbf{P}(A_{ij}) + \sum_{i,j\ne l,m}\mathbf{P}(A_{ij}\cap A_{lm})}\end{aligned}$$

where the last inequality follows from the SECOND MOMENT INEQUALITY.

Limiting Distribution for M_n

Consider the Bernoulli model over a binary alphabet with probability of symbol generation p and $q = 1 - p$, respectively. From the inclusion-exclusion rule we have

$$\begin{aligned} \mathbf{P}(M_n \geq k) &= \mathbf{P}\left(\bigcup_{j=1}^{n}[C_j \geq k]\right) \\ &= \sum_{r=1}^{n}(-1)^{r+1}\binom{n}{r}\mathbf{P}(C_1 \geq k, \ldots, C_r \geq k) \\ &= \sum_{r=1}^{n}(-1)^{r+1}\binom{n}{r}(p^{r+1} + q^{r+1})^k \end{aligned}$$

where the last equality is a consequence of

$$\mathbf{P}(C_1 \geq k, \ldots, C_r \geq k) = (p^{r+1} + q^{r+1})^k \ .$$

Let now $G_n(z)$ be the probability generating function of M_n, and $\widehat{G}_n(z) = \sum_{k \geq 0} z^k \mathbf{P}\{M_n \geq k\}$ Observer that

$$\widehat{G}_n(z) = \frac{1 - G_n(z)}{1 - z}.$$

Limiting Distribution

Thus:

$$\widehat{G}_n(z) = -\sum_{r=1}^{n}(-1)^r\binom{n}{r}\frac{1}{1-z(p^{r+1}+q^{r+1})}\ .$$

Observe that $\mathbf{E}M_n = \widehat{G}_n(1)$ and $\mathbf{E}M_n(M_n-1) = 2\widehat{G}'_n(1)$. In both cases we have to deal with alternating sums shown below

$$\begin{aligned}
\mathbf{E}M_n &= -\sum_{r=1}^{n}(-1)^r\binom{n}{r}\frac{1}{1-(p^{r+1}+q^{r+1})} \\
\mathbf{E}M_n(M_n-1) &= -2\sum_{r=1}^{n}(-1)^r\binom{n}{r}\frac{p^{r+1}+q^{r+1}}{(1-(p^{r+1}+q^{r+1}))^2}\ .
\end{aligned}$$

Observe that all above sums have a form of an alternating sum, which was recently quite extensively studied in the analysis of algorithms.

Alternating Sums

To deal efficiently with such sums we use a Mellin-like approach In particular, for all sequences f_k that do not grow too fast at infinity we have

$$\sum_{r=1}^{n}(-1)^r\binom{n}{r}f_r = \left(1+O\left(\frac{1}{n}\right)\right)\frac{1}{2\pi i}\int_{1/2-i\infty}^{1/2+i\infty} n^{-s}\Gamma(s)f(-s)ds \ ,$$

where $\Gamma(s)$ is the Euler gamma function, and $f(s)$ is an analytical continuation of f_r, that is, $f(s)\mid_{s=r} -f_r$. Then, the limiting distribution of M_n follows from the above and Levy's continuity theorem, that is, M_n is normally distributed and for a complex θ

$$\lim_{n\to\infty} e^{-\theta\mu n/\sigma n}G_n(e^{\theta/\sigma n}) = e^{\frac{1}{2}\theta^2}$$

where $\mu_n = \mathbf{E}M_n$ and $\sigma_n = \sqrt{\mathbf{Var}M_n}$.

Lower Bound for Bernoulli Model

1. With high probability any pair of strings overlaps for less than $\ell/2$. Indeed, $H_n \leq \ell/2$ implies that with **whp**

$$\ell \geq -4 \log_P n \ .$$

2. Let us split every string $\mathbf{z}^i = \mathbf{a}^i \mathbf{b}^i$ where $\mathbf{a}^i$ and $\mathbf{b}^i$ are prefix and suffix of $\mathbf{z}^i$, respectively. We assume that $|\mathbf{b}^i| > \ell/2$ **whp**.

3. Let ρ_t denote the number of positions in a string $\mathbf{a}$ that are equal to the tth symbol $\omega_t \in \Sigma$. For a typical sequence $\rho_t \sim kp_t$.

4. Let $\Omega(k, \epsilon)$ denote the set of strings $\mathbf{a} \in \Sigma^k$ for which $\rho_t \leq (1+\varepsilon)kp_t$, that is,

$$\Omega(k,\epsilon) = \{\mathbf{a} \in \Sigma^k : \ \rho_t(\mathbf{a}, k) \leq (1+\epsilon)kp_t, \ 1 \leq t \leq M\}.$$

From Chernoff's bound we know that

$$\mathbf{P}(\mathbf{a}^{i,k} \notin \Omega(k,\epsilon)) \leq \sum_{t=1}^{M} e^{-\epsilon^2 kp_t/3} = \theta,$$

Lower Bound

5. Set now

$$\epsilon = (\log n)^{-1/3} \text{ and } k = \left\lfloor (1-2\epsilon)\frac{1}{H}\log n \right\rfloor .$$

Then, $\epsilon^2 k \to \infty$ with n and **whp** almost every $\mathbf{a}^{i,k} \in \Omega(k,\epsilon)$, so $\Omega(k,\epsilon)$ is a TYPICAL SET.
Also

$$M(k,\epsilon) = |\{i :\ \mathbf{a}^{i,k} \notin \Omega(k,\epsilon)\}|$$

. Then, $M(k,\epsilon)$ is stochastically dominated by $B(n,\theta)$. So **whp**

$$M(k,\epsilon) = o(\epsilon n).$$

6. Since for a typical $\mathbf{a} \in \Omega(k,\epsilon)$, we have

$$\xi(\mathbf{a}) := \mathbf{P}(\mathbf{a}^{i,k} = \mathbf{a}) \sim e^{-kH},$$

hence the set

$$N(\mathbf{a}) = |\{i : \mathbf{a}^{i,k} = \mathbf{a}\}|$$

has cardinality $n\xi(\mathbf{a}) \geq n^{\epsilon}$ **whp**.

Lower Bound

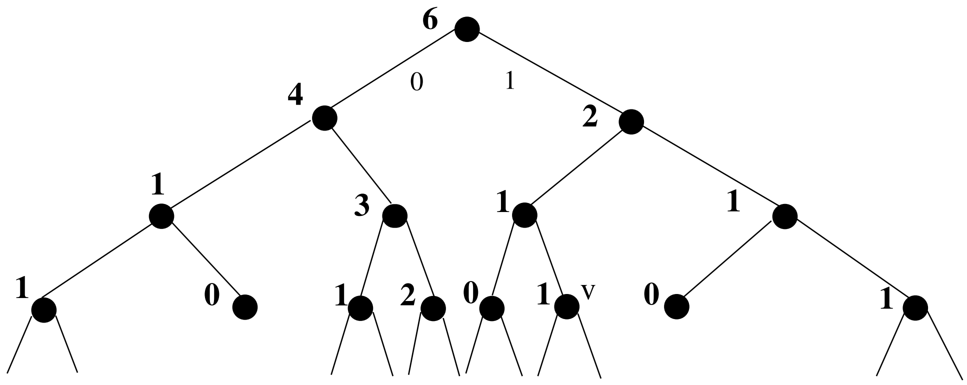

Figure 1: First few levels of T with $\nu(v)$ marked in bold for $\mathbf{a}^1 = 01111$, $\mathbf{a}^2 = 11110$, $\mathbf{a}^3 = 10101$, $\mathbf{a}^4 = 00000$, $\mathbf{a}^5 = 01011$, and $\mathbf{a}^6 = 011000$ (e.g. if $\mathbf{z}$ ends with $\ldots 101$, then the particle Z reaches the vertex v).

7. Let T denote an infinite rooted M-ary tree. The M edges leading down from each vertex are labelled with $\omega_1, \omega_2, \ldots, \omega_M$. The child w of vertex v for which edge (v, w) is called the ω_i child of v. A vertex v of T at depth d is identified with a string $s_d s_{d-1} \ldots s_1$ and is labelled with an integer $\nu(v)$. Here the edges of the path from the root of T to v have labels $s_1, s_2, \ldots, s_d$ and $\nu(v)$ is the number of i such that the d-prefix of $\mathbf{a}^i$ is $s_d s_{d-1} \ldots s_1$. Thus T is defined by the strings $\mathbf{a}^i$ and is independent of the strings $\mathbf{b}^i$.

Lower Bound

8. We model the progress of RGREEDY in the following way: A particle Z starts at the root. When at a vertex v it moves to v's ω_j descendent with probability p_j. The particle stops at depth $\ell/2$. Let $w = s_\kappa s_{\kappa-1}\ldots s_1$ be the lowest vertex on the path traversed that has a non-zero ν value. This process models the computation of the largest suffix $s_\kappa s_{\kappa-1}\ldots s_1$ of $\mathbf{z}$ which can be merged with a prefix of an $\mathbf{a}^i$ i.e. $\mathbf{a}^{i,k}$.

9. We then model the deletion of $\mathbf{a}^t = a_1a_2\ldots a_{\ell/2}$ which had the prefix $a_1a_2\ldots a_\kappa$. Let $w_i = a_1a_2\ldots a_i$. Put $\nu(w_i) = \max\{0, \nu(w_i) - 1\}$ for $1 \le i \le \ell/2$.

10. We repeat the above process $n-1$ times achieving values $\kappa_1, \kappa_2, \ldots, \kappa_n$ of κ. We will show that **whp**

$$\kappa_1 + \kappa_2 + \cdots + \kappa_n \ge (1-5\epsilon)\frac{1}{H} n \log n.$$

Lower Bound

11. We want to show that **whp** we will have $\kappa_t \geq k$ for $1 \leq t \leq n_0 = \lceil (1-3\epsilon)n \rceil$. Now, most of the time the k-suffix $\mathbf{z}^k$ of $\mathbf{z}$ lies in $\Omega(k,\epsilon)$. This follows from the above and because $s_1 s_2 \ldots$ is a random string.

12. If $\mathbf{z}^k \in \Omega(k,\epsilon)$ and

$$\nu(\mathbf{a}) \neq 0 \text{ for all } \mathbf{a} \in \Omega(k,\epsilon) \ ,$$

then $\kappa \geq k$, where $\nu(\mathbf{a})$ is defined for $\mathbf{a} = s_k \ldots s_1$.

We argue next that **whp** the above holds up to $n_0 = \lceil (1-3\epsilon)n \rceil$. If we consider a fixed $\mathbf{a} \in \Omega(k,\epsilon)$, then at this point the number of decrements $r(\mathbf{a})$ in $\nu(\mathbf{a})$ is distributed as $B(n_0, \xi(\mathbf{a}))$. Hence, using $n_0 \xi(\mathbf{a}) \geq (1-3\epsilon)n^{\epsilon}$,

$$\begin{aligned} \mathbf{P}(\exists \mathbf{a} \in \Omega(k,\epsilon) : \ r(\mathbf{a}) \geq (1+\epsilon) n_0 \xi(\mathbf{a})| &\leq 2M^k e^{-(1-3\epsilon)\epsilon^2 n^{\epsilon}/3} \\ &= o(1). \end{aligned}$$

So **whp** at this point $\nu(\mathbf{a}) \geq n(1-\epsilon)\xi(\mathbf{a}) - n_0(1+\epsilon)\xi(\mathbf{a}) > 0$ for every $\mathbf{a} \in \Omega(k,\epsilon)$.

Open Problem

In the DNA fragmentation the following problem arises:
Consider a long DNA string (a sequence over four-letter alphabet). To read it, you first must cut it into smaller pieces (strings). After you read the smaller pieces, you must resembly the original DNA. One way of doing it is to use Shortest Superstring Algorithm, however, in this case the strings are dependent!