Source Coding vs. Channel Coding

The goal of source coding (data compression) is to represent the source with a **minimum of symbols.**

The goal of channel coding (error correction) is to represent the source with a minimum of error probability in decoding.

These goals are obviously in conflict!

Usually we require additional symbols to be transmitted when performing error correction.

Lonardi and Szpankowski (DCC 2003) suggested a way to perform joint data compression and error correction without degrading the compression performance and without adding extra symbols for error correction.

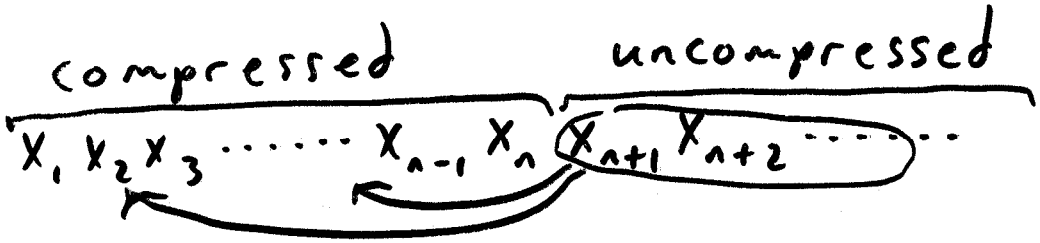
They suggest embedding a Reed-Solomon error-correcting code into the Lempel-Ziv '77 data compression algorithm.

They use the fact that LZ'77 is unable to remove all redundancy from the source.

How many redundant bits are available?
They asked me to find a precise description of how many redundant bits their scheme has available to use.

Recall how the LZ'77 data compression scheme wor

When n bits of the source have already been compressed:



LZ'77 finds the longest prefix of the uncompresse data that appears in the database (the compressed data) and it performs the compression by storing a pointer into the database.

Often, this longest prefix appears more than once in the database.
Any of these database entries will do,
so we can choose any of these pointers.

For example, we can embed an error correction b (say, a parity check bit) here by choosing the fir pointer for "0" and the second pointer for "1".

We let M_n denote the numbers of pointers at Stage. Note that M_n is what we are interested throughout this talk.

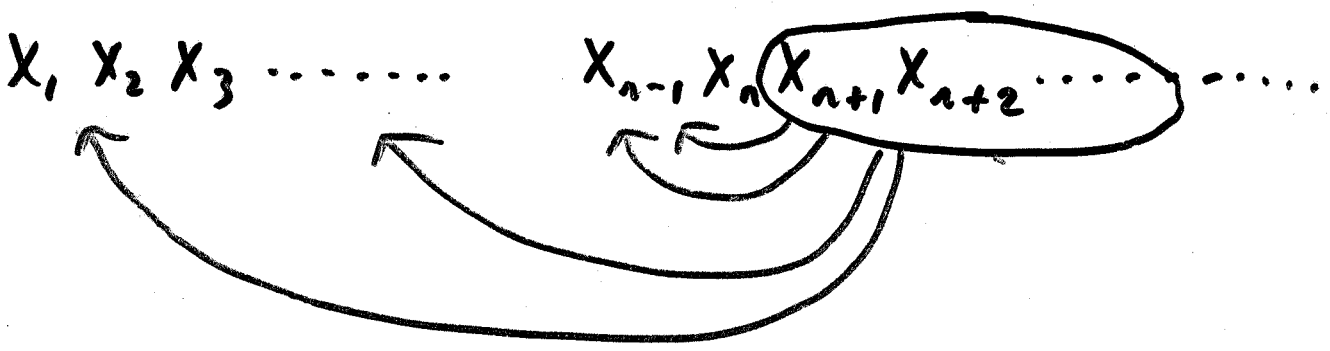
If we can determine the distribution of M_n, then we will have a good idea of how much error correction can be performed by this scheme. Note that $\lfloor \log_2 M_n \rfloor$ bits are available to be used for correcting errors.

In fact, if we modified LZ'77 by not insisting on using the longest prefix of the uncompressed data, then lots more pointers into the database would be available, so lots more error correction can be performed....

but that will have to wait until my postdoctoral years!

My analysis only considers sources $X_1 X_2 X_3 \ldots$ where the X_i's are i.i.d., say with $Pr(X_i = 0) = p$ and $Pr(X_i = 1) = q$. It would be more practical to study Markov sources, (but again this is more complicated and will have to wait until my postdoctoral years.)

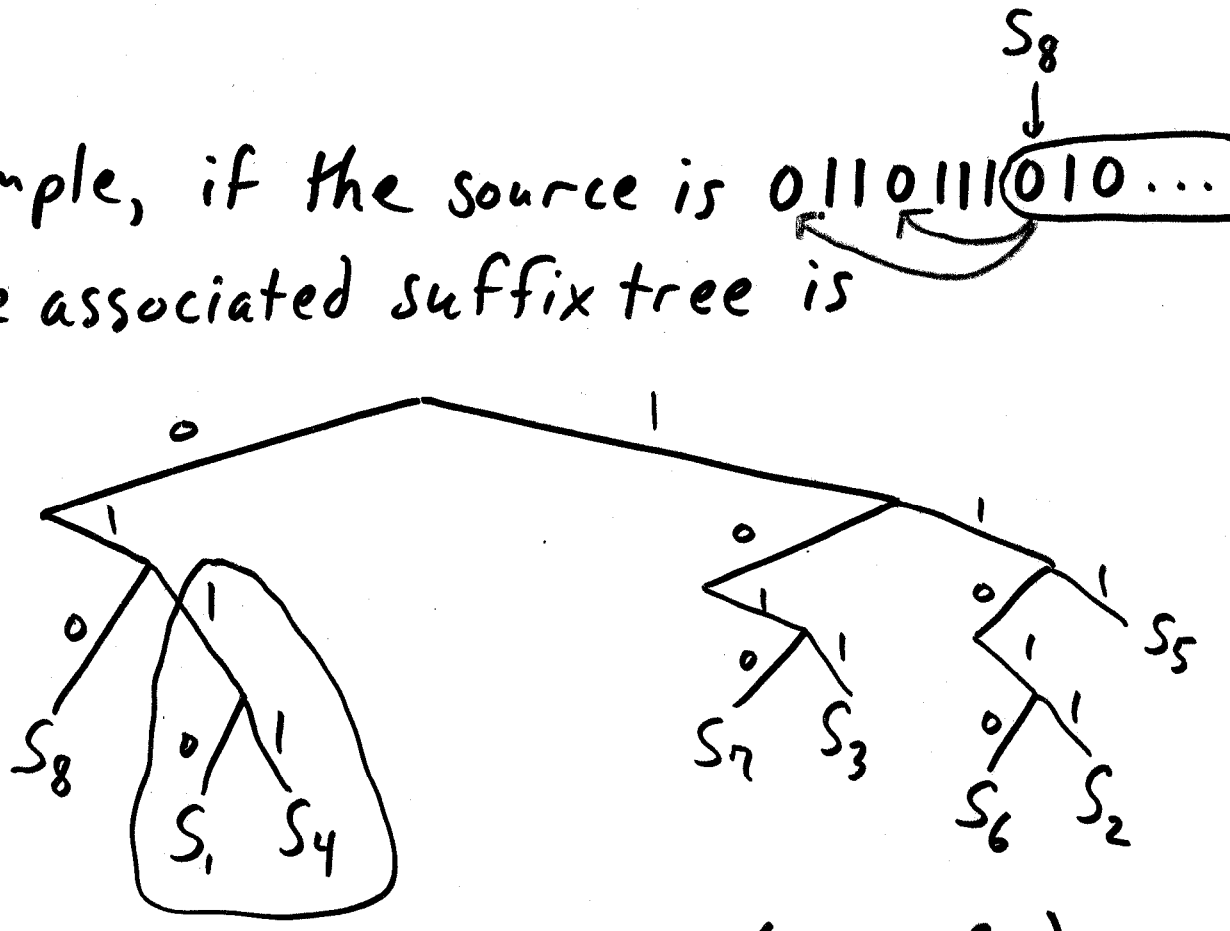
Once again, M_n is the number of copies of the longest prefix of $X_{n+1} X_{n+2} \ldots$ found in the database $X_1 X_2 \ldots X_n$. Here $\boxed{M_n = 4}$:



How do we find M_n in a suffix tree?

If we build a suffix tree from the first n+1 suffixes of the source, then M_n is the size of the subtree that starts at the insertion point of the (n+1)-st string (i.e. the number of strings in this subtree).

For example, if the source is 0110111010... then the associated suffix tree is



So $M_{78} = 2$ because two strings (S_1 and S_4) are in the indicated subtree.

We analyzed M_n for independent tries (call this parameter M_n^I) by establishing a recurrence relation: With probability $\binom{n}{k} p^k q^{n-k} p$ (for instance) the $(n+1)$st string begins with "0" and also exactly k of the first n strings begin with "0". In this situation, we are basically starting afresh with $k+1$ strings! This easily yields a recurrence relation.

If each of the strings begins with, say, "1" but the last string begins with "0", then we know that M_n is exactly the number of strings in the database:

here $M_n = n$

So for the recurrence, we just consider a subtree, and then a subtree of it, etc., until this situation finally occurs.

This problem was made easier to solve when first considering n to be itself a Poisson random variable (and then later using Jacquet & Szpankowski's Depoissonization Lemma

In the end, we obtained

$$E[u^{M_n}] = -\frac{q\ln(1-pu) + p\ln(1-qu)}{h} + \delta(\log_{\frac{1}{p}} n, u) + O(n$$

where δ is a fluctuation function with small amplitude.

So M_n follows the logarithmic series distribution asymptotically plus some fluctuations.

Also, the jth factorial moment of M_n is

$$E[M_n^{\underline{j}}] = \Gamma(j)\frac{q(p/q)^j + p(q/p)^j}{h} + \text{small fluctuations}$$

In particular $E[M_n] \sim \frac{1}{h} + \text{small fluctuations}$

so this verifies Wyner's assertion that $E[M_n] = O(1)$

(Incidentally, $h = -p\ln p - q\ln q$ is just the entropy of the source.)

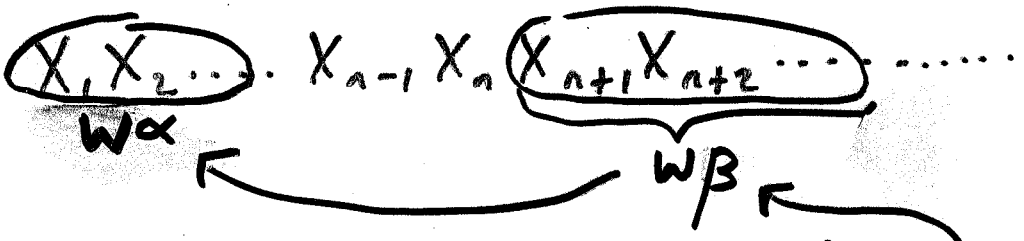
So we have very precise information about M_n defined for independent tries (now we'll be precise again and call this M_n^I).

What about the actual M_n, namely the parameter of suffix trees, that we are interested in for Lempel-Ziv '77?

Even determining the generating function for M was difficult (and has consumed much of my time since the MSRI "A of A" workshop this summer

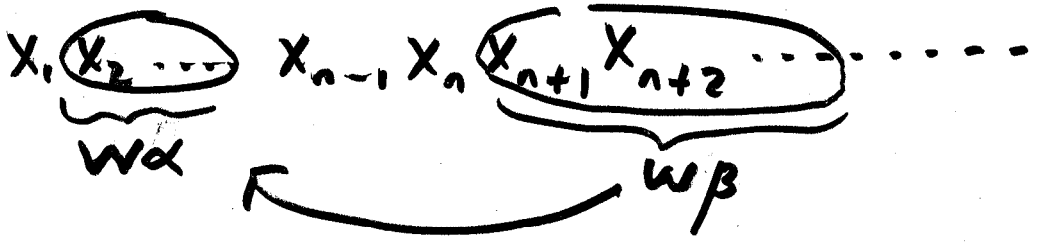
The trouble, of course, is all the overlaps.

We usually let w denote the longest prefix of the uncompressed data that appears in the databa



So we really need, for instance, $w\alpha$ here and then no copies of $w\beta$ in the database and then we have M_n copies of $w\alpha$ in the data

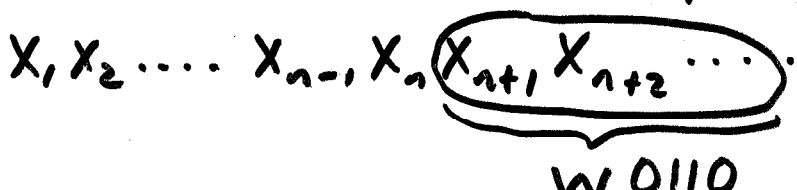
It is no trouble to look for the number of copies of word in the database. We can enumerate this somewhat easily. But we are also requiring that we have no copies of another word! This is tricky.



One proposed solution was the following:

If w is the prefix of the uncompressed portion of the data, then $1 \leq M_n \leq k$ if there are k copies of w is the database.

The trouble is this, for example:



w appears 10 times
w0 appears 10 times
w01 appears 8 times
w011 appears 2 times
w0110 does not appear so here $M_n = 2$

We have no idea what length w to consider.

$M_n = k$ exactly when:

if L is the largest nonnegative integer such that $w_{n+1} w_{n+2} \cdots w_{n+L} = w_{i+1} w_{i+2} \cdots w_{i+L}$ for some $0 \le i < n$

(i.e. $w_{n+1} w_{n+2} \cdots w_{n+L}$ appears in the database starting at position $i+1$)

then there are exactly k such values of i

(i.e. $w_{n+1} w_{n+2} \cdots w_{n+L}$ appears exactly k times in the database)

So $M_n = k$ if and only if

(1) there are exactly k (possibly overlapping) copies of $w\alpha$ in the database

(2) there are no copies of $w\beta$ in the database

(3) $X_{n+1} \cdots X_{n+|w|} = w\beta$

---

i.e. w is the longest prefix of $X_{n+1} X_{n+2} \cdots$ in the database, and it appears exactly k times in the database.

We define M_n^{I} by assuming that the suffixes of $X_1, X_2, \ldots$ are independent.

Thus, if we have $n+1$ independent strings, then $M_n = k$ if and only if

(1) exactly k of the first n strings begin with $w\alpha$

(2) none of the first n strings begin with $w\beta$

(3) the $(n+1)$st string begins with $w\beta$.

So the probability that $M_n = k$ is exactly

$$\binom{n}{k} \Pr(w\alpha)^k (1 - \Pr(w))^{n-k} \Pr(w\beta)$$

(1) (2) (3)

follows immediately that

$$u^{m_n^I}\Big] = \sum_k \sum_{\substack{\omega \in A^* \\ \alpha \in A}} \binom{n}{k} Pr(\omega\alpha)^k (1 - Pr(\omega))^{n-k} Pr(\omega\beta) u^k$$

ltiplying by z^n and summing over n, it follows immediately that

$$E\left[u^{m_n^I}\right] z^n = \sum_{\omega \in A^*} \frac{Pr(\omega)}{1 - z(1 - Pr(\omega))} \sum_{\alpha \in A} Pr(\beta) \frac{uz\, Pr(\omega) Pr(\alpha)}{1 - z(1 + u Pr(\omega) Pr(\alpha) - Pr(\omega))}$$

Now we discuss how to compute $\sum_n E[u^{M_n}] z^n$

Define

R_w = set of words containing only one copy of w, located at the right end

(as in Jacquet & Szpankowski, Ch. 7 of new Lothaire

$\tau_w^{(\alpha)}$ = Set of words such that any word in $w\alpha\tau_w^{(\alpha)}$ contains exactly two occurrences of w, one at the left end and one at the right end

---

When examining $X_1 X_2 \ldots$ we have $M_n = k$ when

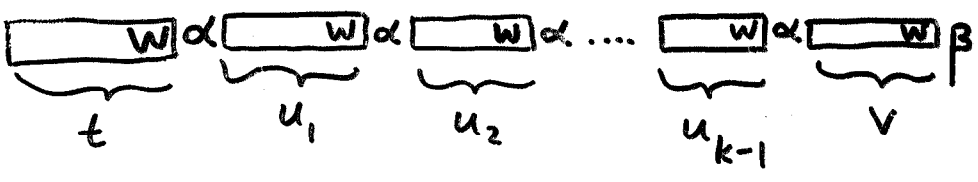
(1) k copies of $w\alpha$ appear (with no $w\beta$'s)

(2) then one copy of $w\beta$ appears, i.e.:

$$X_1 X_2 \ldots = t\alpha u_1 \alpha u_2 \alpha \ldots u_{k-1} \alpha v \beta$$

with $t \in R_w$

$u_1, u_2, \ldots, u_{k-1}, v \in \tau_w^{(\alpha)}$



Thus

$$\sum_{n} E[u^{M_n}] z^n = \sum_{k} \sum_{\substack{\omega \in A^* \\ \alpha \in A}} \sum_{t \in R_\omega} Pr(t\alpha) z^{|t|+1} u \left(\prod_{i=1}^{k-1} \sum_{u_i \in T_\omega^{(\alpha)}} Pr(u_i \alpha) z^{|u_i|+1} u \right) \sum_{v \in T_\omega^{(\alpha)}} Pr(v\beta) z^{|v|-|\omega|}$$

$$= \sum_{\alpha \in A} Pr(\beta) \sum_{\omega \in A^*} \frac{R_\omega(z)}{z^{|\omega|}} \sum_{k \geq 1} \left(Pr(\alpha) z u T_\omega^{(\alpha)}(z) \right)^k$$

$$= \sum_{\alpha \in A} Pr(\beta) \sum_{\omega \in A^*} \frac{R_\omega(z)}{z^{|\omega|}} \frac{Pr(\alpha) z u T_\omega^{(\alpha)}(z)}{1 - Pr(\alpha) z u T_\omega^{(\alpha)}(z)}$$

Again, as in Jacquet & Szpankowski's "Lothaire Ch." define

$$M_w = \{v \mid wv \text{ has exactly two occurrences of } w, \text{ one at the left and one at the right}\}$$

A crucial way to understand $\tau_w^{(\alpha)}$ is to note that

$$\alpha\tau_w^{(\alpha)} + \beta\tau_w^{(\beta)} = M_w$$

In other words, $\alpha\tau_w^{(\alpha)}$ is exactly the set of words in M_w that begin with α.

For ease of notation, write $H_w^{(\alpha)} = \alpha\tau_w^{(\alpha)}$

and $H_w^{(\beta)} = \beta\tau_w^{(\beta)}$

(I think of the $H_w^{(\alpha)}$ as "half" of M_w, even though of course they do not split M_w in half.)

$$\text{So } M_w = H_w^{(\alpha)} + H_w^{(\beta)}$$

Finally, use Jacquet & Szpankowski's notation

$$U_w = \{u \mid wu \text{ has exactly one occurrence of } w \text{ (at the left)}\}$$

Recall that, if we can determine $T_w^{(\alpha)}(z)$ then we know $\sum_n E[u^{M_n}] z^n$. It suffices to determine $H_w^{(\alpha)}$

To do this, we find two equivalent ways to describe the set of all words with no occurrences of $w\beta$.

(1.) $A^* - R_{(w\beta)}\left(M_{(w\beta)}\right)^* U_{(w\beta)}$

which has generating function

$$\frac{1}{1-z} - \frac{R_{(w\beta)}(z)\, U_{(w\beta)}(z)}{1 - M_{(w\beta)}(z)}$$

Method #2 for words with no occurrences of $w\beta$

2a. Words with no occurrences of $w\beta$ or $w\alpha$ at all but do end with w:

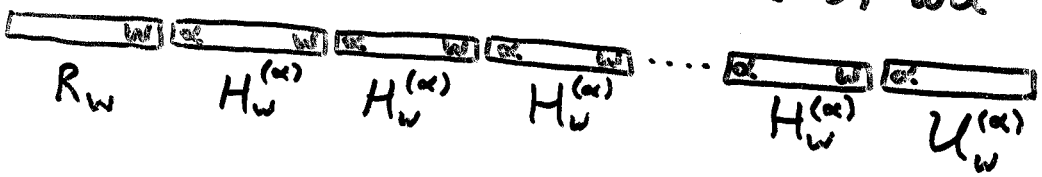
R_w which has generating function $R_w(z)$

2b. Words with no occurrences of $w\beta$ or $w\alpha$ at all and do not end with w
(i.e. words with no occurrences of w)

$A^* - R_w(M_w)^* U_w$

which has generating function $\frac{1}{1-z} - \frac{R_w(z)U_w(z)}{1-M_w(z)}$

2c. Words with no occurrences of $w\beta$ but at least one occurrence of $w\alpha$



First we define

$U_w^{(\alpha)} = \{ v \mid$ v starts with α
wv has exactly one occurrence of $w\alpha$ but no occurrences of w

NOTE: It is OK if words from $U_w^{(\alpha)}$ end with w.

So the words in 2c. are

$R_w(H_w^{(\alpha)})^* U_w^{(\alpha)}$

So now we need the generating function associated with

$$U_w^{(\alpha)} = \{v \mid \begin{array}{l} v \text{ starts with } \alpha \\ wv \text{ has exactly one occurrence of } w\alpha \\ \qquad \text{but no occurrences of } w\beta\} \end{array}$$

$U_w^{(\alpha)}$ can be broken into two disjoint subsets — the v's which end in w and the v's which do not end in w.

---

1. $\{v \mid v \in U_w^{(\alpha)} \text{ and } v \text{ ends in } w\} = H_w^{(\alpha)}$

in this case, v starts with α and wv has exactly two occurrences of w, one at the left and one at the right

---

2. $\{v \mid v \in U_w^{(\alpha)} \text{ and } v \text{ does not end in } w\}$

$= \{v \mid v \text{ starts with } \alpha \text{ and } wv \text{ has exactly one occurrence of } w\}$

We write $V_w^{(\alpha)}$ to denote the set of these v's.

We claim $V_w^{(\alpha)} \cdot A = H_w^{(\alpha)} + V_w^{(\alpha)} - \{\alpha\}$

Proof that $V_w^{(\alpha)} \cdot A = H_w^{(\alpha)} + V_w^{(\alpha)} - \{\alpha\}$

First we prove $\subseteq$.

If $v \in V_w^{(\alpha)}$ and $\gamma \in A$ then since v starts with α we have $v\gamma \neq \alpha$.

If $v\gamma$ ends in w, then $v\gamma \in H_w^{(\alpha)}$

If $v\gamma$ does not end in w then $v\gamma \in V_w^{(\alpha)}$

So $\subseteq$ holds.

Now we prove $\supseteq$.

If $h \in H_w^{(\alpha)}$ with $h \neq \alpha$ then pull off the last character of h and call the result h^- (so $h = h^- \cdot \gamma$ for some

Then wh^- has exactly one occurrence of w.

Also $h \neq \alpha$ but h starts with α so h^- also starts with

Thus $h^- \in V_w^{(\alpha)}$.

If $v \in V_w^{(\alpha)}$ with $v \neq \alpha$ then pull off the last character of v and call the result v^- (so $v = v^- \cdot \gamma$ for some

Then wv^- has exactly one occurrence of w.

Also $v \neq \alpha$ but v starts with α so v^- also starts with

Thus $v^- \in V_w^{(\alpha)}$.

□

So we proved $V_w^{(\alpha)} \cdot A = H_w^{(\alpha)} + V_w^{(\alpha)} - \{\alpha\}$.

Thus $(z-1)V_w^{(\alpha)}(z) = H_w^{(\alpha)}(z) - Pr(\alpha)z$

so $V_w^{(\alpha)}(z) = \dfrac{H_w^{(\alpha)}(z) - Pr(\alpha)z}{z-1}$

Before that, we proved $U_w^{(\alpha)} = H_w^{(\alpha)} + V_w^{(\alpha)}$ so

$$U_w^{(\alpha)}(z) = H_w^{(\alpha)}(z) + \frac{H_w^{(\alpha)}(z) - Pr(\alpha)z}{z-1}$$

$$= \frac{z H_w^{(\alpha)}(z) - Pr(\alpha)z}{z-1}$$

From our discussion about words with no occurrences of $w\beta$, we discovered

$$A^* - R_{(w\beta)}(M_{(w\beta)})^* U_{(w\beta)}$$

$$= R_w + (A^* - R_w M_w^* U_w) + R_w (H_w^{(\alpha)})^* U_w^{(\alpha)}$$

Now we know <u>all</u> of the generating functions for these languages except $H_w^{(\alpha)}$ so we can solve for it.

We simplify and simplify, using Jacquet & Szpankowski helpful observations that

$$\frac{1}{1-M_w(z)} = S_w(z) + \frac{P_r(w)\, z^{|w|}}{1-z}$$

$$U_w(z) = \frac{M_w(z) - 1}{z-1}$$

$$R_w(z) = P_r(w)\, z^{|w|} \cdot U_w(z)$$

and we finally get

$$H_w^{(\alpha)}(z) = \frac{R_{(w\beta)}(z) - z P_r(\beta) R_w(z)}{R_{(w\beta)}(z)}$$

$$= \frac{U_{(w\beta)}(z) - U_w(z)}{U_{(w\beta)}(z)}$$

$$= \frac{M_{(w\beta)}(z) - M_w(z)}{M_{(w\beta)}(z) - 1}$$

$$\vdots$$

$$= 1 - \frac{(1-z) S_{(w\beta)}(z) + P_r(w\beta)\, z^{|w|+1}}{(1-z) S_w(z) + P_r(w) z^{|w|}}$$

Now recall $H_w^{(\alpha)} = \alpha \tau_w^{(\alpha)}$

and

$$\sum_n E[u^{M_n}] z^n = \sum_{\alpha \in A} Pr(\beta) \sum_{w \in A^*} z^{|w|} \cdot \frac{R_w(z)}{1 - Pr(\alpha) z u T_w^{(\alpha)}(z)}$$

So, since we know the generating function for $H_w^{(\alpha)}$, we can substitute it in here and then we obtain the bivariate generating function for M_n (the multiplicity matching parameter).

We get...

$$M(z,u) = \sum_n E[u^{M_n}] z^n$$

$$= \sum_{w \in A^*} \frac{Pr(w)}{(1-z)S_w(z) + z^m P(w)}$$

$$\times \sum_{\alpha \in A} Pr(\beta) \frac{u\left[(1-z)(S_w(z) - S_{(w\beta)}(z)) + z^m Pr(w)(1 - Pr(\beta)z)\right]}{(1-z)S_w(z) - u\left[(1-z)(S_w(z) - S_{w\beta}(z)) + z^m Pr(w)(1-Pr(\beta)z)\right] + z^m Pr(w)}$$

Recall that M_n for independent strings in a trie (not a suffix tree) has bivariate generating function

$$M^I(z,u) = \sum_n E[u^{M_n^I}] z^n$$

$$= \sum_{w \in A^*} \frac{Pr(w)}{1 - z(1-Pr(w))} \sum_{\alpha \in A} Pr(\beta) \frac{uz\, Pr(w) Pr(\alpha)}{1 - z(1 + u\, Pr(w) Pr(\alpha) - Pr(w))}$$

Comparison of $M(z,u)$ and $M^{I}(z,u)$ looks hopeful for a variety of reasons.

For example, recall Jacquet and Szpankowski's comparison of the typical depth in suffix trees and in tries built over independent strings.

$$D(z,u) = \sum_n n E[u^{D_n}] z^n = (1-u) \sum_w u^{|w|} \frac{P_r(w) z}{[(1-z) S_w(z) + P_r(w) z^{|w|}]^2}$$

$$D^{I}(z,u) = \sum_n n E[u^{D_n^I}] z^n = (1-u) \sum_w u^{|w|} \frac{P_r(w) z}{[1 - z(1 - P_r(w))]^2}$$

Currently I am working on comparing $M(z,u)$ and $M^I(z,u)$

For $M(z,u)$, I am trying to prove that for $|z|<1$
the first pole occurs for $|u|>1$.

Then I can compare the behavior of $M(z,u)$ and $M^I(z,u)$
at the poles.

Since we have already a very precise description of $M^I(z,u)$,
this will (hopefully) yield a nice analysis of $M(z,u)$.

Jacquet and Szpankowski proved already that $S_w(z)$ is simply 1 with high probability.

More precisely, $\exists \delta < 1$, $\theta > 0$, and $\rho > 1$ such that $\rho\delta <$ and

$$\sum_{w \in A^k} [\![|S_w(\rho) - 1| \le (\rho\delta)^k \theta]\!] \Pr(w) \ge 1 - \theta\delta^k$$

Applying this result twice (once to $S_w(z)$ and again to $S_{(w\beta)}(z)$) we can prove similarly that $S_w(z) - S_{(w\beta)}(z)$ is simply 0 with high probability,

but I prefer the following argument:

Except for the z^m term of $S_{(w\beta)}(z)$ we observe that each term of $S_{(w\beta)}(z)$ is the <u>same</u> as the analogous term of $S_w(z)$ <u>or</u> is <u>exactly</u> 0. Also $S_{(w\beta)}(z)$ always has a "1"

Thus $|S_w(z) - S_{(w\beta)}(z)| \le |S_w(z) - 1| + p^m z^m$

(if $q \le p$). This gives a result similar to the one fro

Jacquet & Szpankowski's Lemma

$$\sum_{w \in A^k} [\![|S_w(\rho) - S_{(w\beta)}(z)| \le (\rho\delta)^k \theta]\!] \Pr(w) \ge 1 - \theta\delta^k$$

but we are only applying their lemma once.