Capacity of Binary Deletion Channel

Joshua Brakensiek and Samuel Zbarsky

Mitzenmacher (2008), ”A survey of results for deletion channels and related synchronization channels”

Dec 3 2014
A *binary deletion channel with deletion probability* $p$ takes a binary string and deletes each bit independently with probability $p$.

\[ \begin{align*}
1011011001010000 \\
1011110100
\end{align*} \]

- We care about this channel’s capacity
- One instance of deletion-insertion codes (where each bit gets replaced with some distribution of run lengths)
Upper bound

- less information than binary erasure channel
- Thus, capacity at most $1 - p$.
- Slightly better bounds are known.
- Substantially better bounds exist only for low $p$. 

Joshua Brakensiek and Samuel Zbarsky
Deletion Codes
Idea: pick codewords randomly. When receiving, if message could come from just one codeword, decode to that; else, fail.

- Works poorly on the deletion channel.
- Gives positive rate only if $p < 1/2$.
- Main idea: if $X$ is a string of length $< n/2$, then w.h.p. $X$ is a substring of a random codeword.
Better idea: generate codewords using a Markov chain.

- There is a fixed probability $\gamma$ that the $i$th bit is the same as the $(i+1)$th bit.
- Decode same way as before.
- Received codeword satisfies an analogous Markov chain, making analysis easier.
- Gives positive rate.
- More generally, can have length of runs of 0s and 1s follow some distribution.
Better idea: generate codeword randomly, having run lengths follow some distribution

- split received word into blocks, see what each block came from:
  
<table>
<thead>
<tr>
<th>sent</th>
<th>received</th>
</tr>
</thead>
<tbody>
<tr>
<td>001100</td>
<td>000</td>
</tr>
<tr>
<td>01000</td>
<td>1</td>
</tr>
<tr>
<td>1011001</td>
<td>111</td>
</tr>
</tbody>
</table>

- will get some distribution of block lengths, for each length $k$, some distribution of sequences it could have come from

- for each pair $(k, s)$ of a length and sequence it comes from (e.g. $(3, 001100)$), number of times it happens is close to expected value.
Look at all possibilities for how many times each pair happens
For each, look at all possible ways the word you receive could be made
if rate of original code below some value, unlikely to get any collisions
Poisson Repeat Channel

Different channel: independently repeat each bit of codeword based on Poisson distribution

- High-rate codes exist over this channel using jigsaw idea
- Take such a code $C$ of length $n(1 - p)/\lambda$ for the Poisson repeat channel with mean $\lambda$.
- Send each $c \in C$ through Poisson repeat channel with mean $\lambda/(1 - p)$ to get a new codebook $C'$ with words of length $n$.
- Send through deletion channel. Each $c \in C$ is sent through Poisson channel with mean $\lambda$.
Decide using the Poisson decoder w.h.p. there are few collisions in $C'$.  
At $\lambda = 1.79$, yields capacity bound of $0.1185(1 - p)$ (about $(1 - p)/9$).  
Best known for large $p$.  

Poisson Repeat Channel
”Optimal” Idea: When receiving a codeword, pick the most likely codeword from the codebook.

- Would give better capacity, but difficult to analyze.
- Jigsaw gives better bounds because it is closer to maximum likelihood.
Open Questions

- Efficient algorithms for the deletion channel (especially $p$ large)
- Improve bounds using the methods discussed.
- Find better approximations of maximum likelihood decoding.
- Look at other deletion-insertion channels
  - sticky channels