Random Linear Coding in Distributed Storage Systems

Adam Visegradi, MTA SZTAKI
2013, BOINC Workshop

The presented work is supported by the FP7 Capacities Programme under contract nr RI-312297 (IDGF-SP)
Volunteer Storage

• Unreliable node $\rightarrow$ need **redundancy**

• Possibilities
  
  o Replication
    • $k$ blocks of data
    • $r$ factor of replication
    • $kr$ copies of each block are stored
  
  o Erasure coding
    • $k$ blocks of data
    • $n>k$ distinct coded blocks are generated
    • Any $k' \geq k$ of the coded blocks will be sufficient to reconstruct data

• [http://boinc.berkeley.edu/trac/wiki/VolunteerDataArchival](http://boinc.berkeley.edu/trac/wiki/VolunteerDataArchival)
  
  o 40 blocks, 20 node failures: 2000% vs. 50% redundancy
Volunteer Storage

• Need *redundancy*
• Possibilities
  - Replication
  - Erasure coding
  - **Random Linear Coding**

<table>
<thead>
<tr>
<th>Coeff. GF(2)</th>
<th>Packet</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>A</td>
</tr>
<tr>
<td>0</td>
<td>B</td>
</tr>
<tr>
<td>1</td>
<td>C</td>
</tr>
<tr>
<td>1</td>
<td>D</td>
</tr>
</tbody>
</table>

Coded block: $A \text{ xor } C \text{ xor } D$
Volunteer Storage

- Need **redundancy**
- Possibilities
  - Replication
  - Erasure coding
  - **Random Linear Coding**

<table>
<thead>
<tr>
<th>Coeff. GF($2^w$)</th>
<th>Packet</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>A</td>
</tr>
<tr>
<td>b</td>
<td>B</td>
</tr>
<tr>
<td>c</td>
<td>C</td>
</tr>
<tr>
<td>d</td>
<td>D</td>
</tr>
</tbody>
</table>

Coded block: $aA \text{ xor } bB \text{ xor } cC \text{ xor } dD$
Why RLC?

• Data reconstruction is **not** required
  - Storage overhead is still needed centrally
  - Using *sparse* coding, this overhead is lower

• Same redundancy with lower storage overhead; better **reliability**

• *Assuming* authenticity, complete **decentralization** is possible
Why not RLC?

- Compute intensive
Thank you!

https://github.com/avisegradi/rnc-lib

Adam Visegradi <adam.visegradi@sztaki.mta.hu>