Network-Coding with Tetrys: On-the-fly, convolutional coding for the erasure channel

Jonathan Detchart
E. Lochin, J. Lacan,
P-U Tournoux, V. Roca
• Blocks FEC Coding:

```
block i  block i+1  block i+2
```

• Sliding encoding window:

```
continuous data flow
```

```
sliding encoding window
```

• Tetrys, elastic encoding window:

```
elastic encoding window
```

```
continuous data flow
```
• The encoding window can be adjusted depending on the feedback

code rate : 2/3
• Blocks FEC Coding:
  > burst erasure protection increases with the block size
  > as well as the encoding/decoding complexity
  > there are limits on the block size with real time flows

• Sliding encoding window:
  > same complexity and same recovery time if +/- losses

• Tetrys, elastic encoding window:
  > full reliability is achievable
  > the recovery delay is independent of the RTT
  > recovery delay easily adjustable with the code-rate
• Impact of the finite field size on the decoder:

  - Bernoulli channel:
    - \( PLR = 20\% \)
    - \( R = 2/3 \)
• Impact of the finite field size on the decoder:

  - Gilbert-Elliot channel:
    - $PLR = 20\%$
    - $R = \frac{2}{3}$
    - Average erasure burst size = 3
• Real time transmissions
  − Video-conferencing

• Delay Tolerant Networking
  − On going discussion with the DTNRG (IRTF)

• Reliable UDP tunnel
  − See the demo (with Tetrys Release 2.0 and TUNTAP module)

• Reliable multicast
• Simple API using callbacks system
  – Tetrys automatically manages encoded/decoded packets
  – Source packets can have a variable size (up to 64 kB)

• Easy to integrate Tetrys inside existing code
  – VLC integration
  – TUNTAP usage

• Can get some [en|de]coding statistics
  – Matrix size, memory usage, ...
Questions ?

See for further details :
- http://websites.isae.fr/tetrys