Improving the Opus Codec While Preserving Compatibility

Presenter: Jean-Marc Valin
jmvalin@amazon.com

IETF 116

Opus

- Speech/audio codec standardized as RFC 6716 in 2012
- MTI audio codec for WebRTC
- Supports wide range of applications, sampling rates, bitrates, frame sizes

Proposal

- Improve Opus using recent advances in audio coding (e.g. using deep learning)
- Maintain full compatibility with original specification
- Proposed goals/extensions:
  1) Improve robustness to packet loss through redundancy
  2) Improve low-bitrate speech quality (w/ and w/o side info)
  3) Improve low-bitrate music quality (w/ and w/o side info)

Why?

- Recent advancements make it possible to do things we didn’t think were possible back in 2012
- Deploying a new codec is expensive, time-consuming
- Opus is already deployed to billions of devices
- Having a single codec reduces interoperability issues

How?

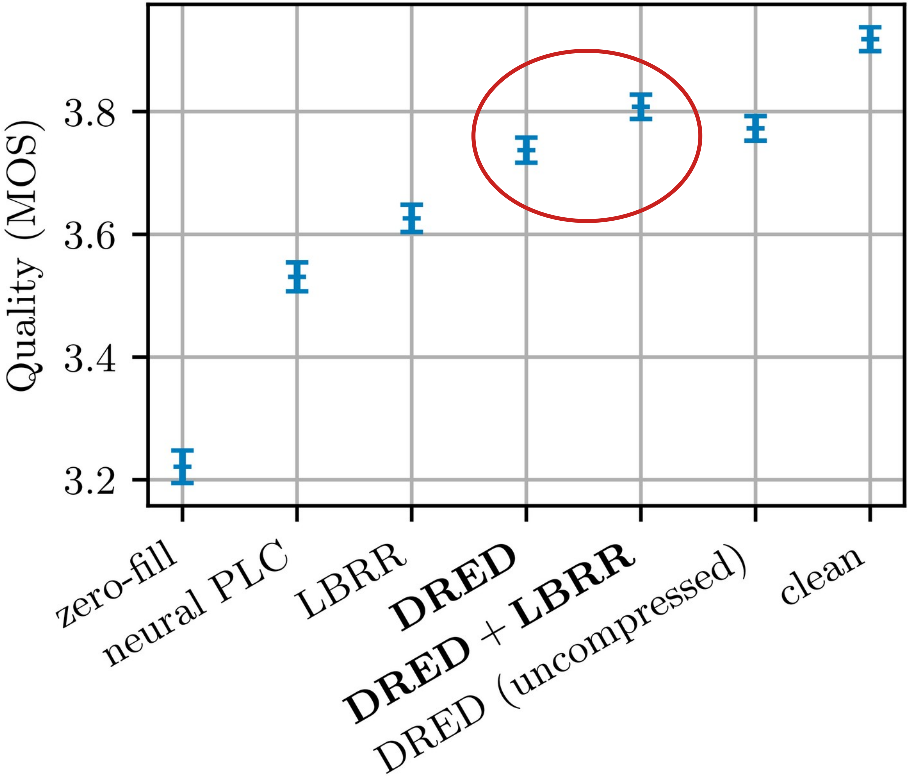
- We propose to transmit extensions as Opus padding
  - No change to the meaning of the other bits
  - Extensions will be ignored by older decoder
- Details in draft-valin-opus-extension-00

Deep REDundancy (DRED)

- Code very low-bitrate redundant acoustic features for past audio in each packet
  - Lost packets synthesized by neural vocoder
  - Can code 1 second (50x) redundancy with 32 kb/s
- Audio examples
  - Opus w/o loss (40 kb/s)
  - Opus+LBRR (40 kb/s)
  - Opus+LBRR+DRED (72 kb/s)
- See draft-valin-opus-dred-00

DRED Results

- Mean Opinion Score (MOS) on Deep PLC Challenge data



...and running code

- DRED running code (experimental) at https://gitlab.xiph.org/xiph/opus in neural_fec3 branch
  - Takes 5-10% CPU on a recent laptop (can be improved)
- DRED paper: https://arxiv.org/pdf/2212.04453
- No running code yet for other goals, but experiments with promising

Proposed Path

- Re-open codec WG, or
- Create new mlcodec WG