Updates on Opus Speech Coding Enhancement

Jan Buethe

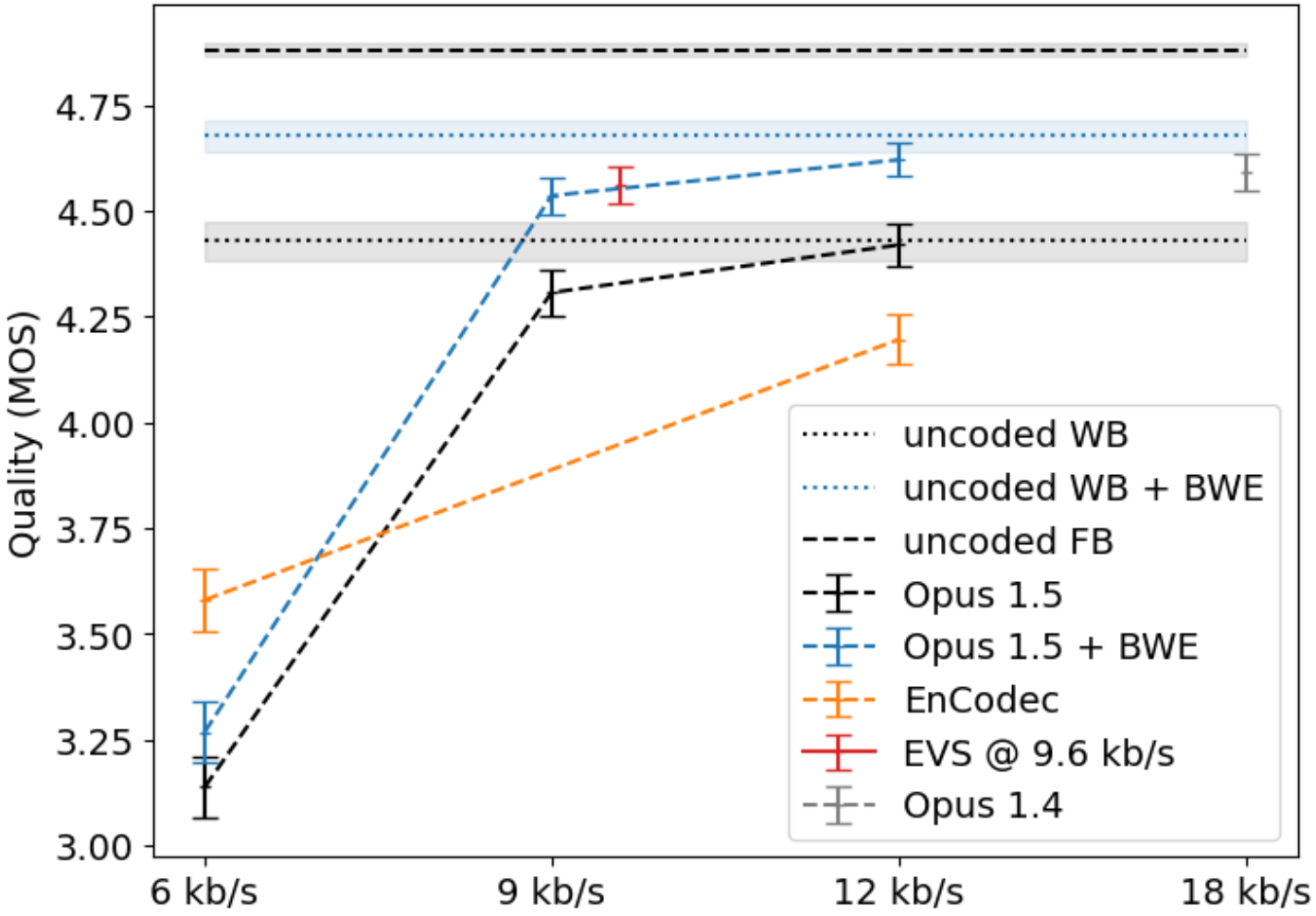
draft-buethe-opus-speech-coding-enhancement-03

Update on Algorithm Development

Blind bandwidth extension for wideband speech

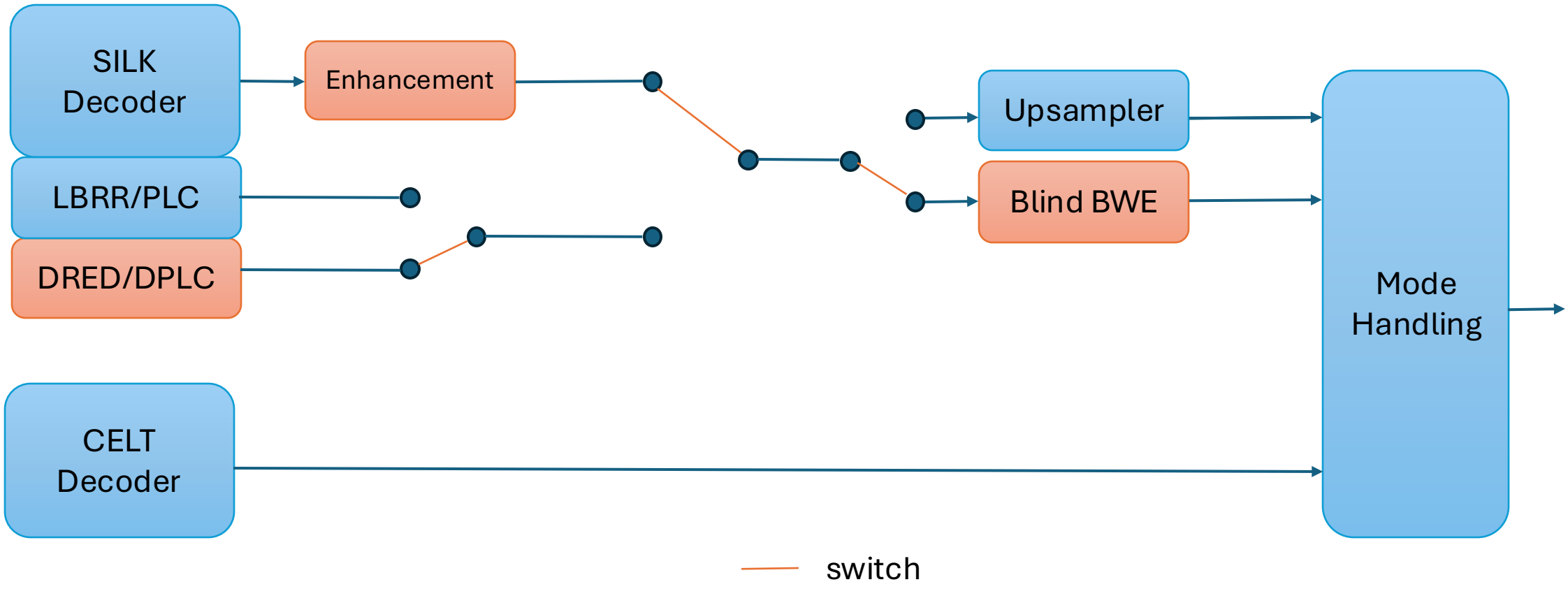
- Both SILK decoder and vocoder for DRED / deep PLC produce wideband speech
- Proposal: add a blind bandwidth extension algorithm for wideband to fullband extension
- Result: significant quality improvement with small DNN with 370 K parameters and 410 MFLOPS (205 MMACS)
  - Same delay as SILK 16->48 kHz resampler (13 samples @ 48 kHz)
  - Python code and model weights on exp_bwe branch of Opus repo (https://gitlab.xiph.org/xiph/opus/-/tree/exp_bwe)

P.808 Test Results



- BWE gives significant improvement at 9, 12 kb/s and for clean speech input
- Opus 1.5 + BWE at 9 kb/s matches EVS at 9.6 kb/s
- Opus 1.5 + BWE matches quality of Opus 1.4 at 18 kb/s somewhere between 9 and 12 kb/s
- Opus 1.5 at 9 kb/s outperforms neural EnCodec codec at 12 kb/s (likely a robustness issue)

Envisioned Integration into Opus Decoder



Draft Proposal for Qualifying Wideband SILK Enhancement Algorithms

Overview and Scope

- Enhancing SILK decoder breaks mandatory conformance test
- Proposed draft updates Opus specification (RFC6716) to allow for enhancement algorithms that meet a set of requirements instead
- Preferable to specifying individual algorithm since it leaves room for future improvement
- Goals of requirements are
  - Ensuring the enhancement algorithm meets basic quality requirements
  - Maintaining Opus decoder integrity
  - Ensuring interoperability with legacy decoders

Quality

- Gold standard for evaluating audio quality is subjective testing
- Objective methods, while convenient, are not reliable
- To make that point clear, draft states that "…any SILK enhancement algorithm SHOULD undergo subjective evaluation before integration into the Opus decoder."
- The best method and test material might depend on intended use case => no mandatory test specified
- Recommendation: Use P.800 or P.808 listening tests for evaluating new algorithms and at least informal listening for modifications of existing algorithms

Decoder Integrity I

- Designed large scale objective test based on previous experiments with MOC degradation metric
- Data: 81 languages and dialects from Mozilla Common Voice dataset + 13 music clips from free music archive => 82 test groups
- Testing large set of encoder operating points combining
  - Multiple bitrates and bitrate switching
  - Constant bitrate and variable bitrate
  - 10 and 20 ms frame sizes
  - Multiple encoder complexity settings
  - Switching between SILK and CELT

Decoder Integrity II

- Python script to run tests available: https://gitlab.xiph.org/xiph/opus/-/blob/osce-testing/dnn/torch/osce/stndrd/qualification/run_osce_tests.py
- ~ 4 GB of input testvectors: https://media.xiph.org/opus/ietf/osce_testvectors_v0.zip
- Enhancement algorithms LACE and NoLACE pass all tests
- Barely trained LACE and NoLACE models (simulation of bad algorithms) fail all tests by large margin => can leave substantial headroom for future enhancement algorithms

Interoperability

- Opus encoder provides much freedom
- Possibility of decoder enhancement can provide incentive to abuse this freedom
- Example: Get the encoder to produce a specific bit sequence that is interpreted by an enhancement method with complete disregard of how legacy decoded bitstream sounds
- Goal: prevent this without restricting current encoder freedom
- Proposed requirement: "if an Opus encoder produces a bitstream that can be decoded into a reproduction of the encoded signal with an enhanced Opus decoder, then that bitstream MUST also result in a reproduction of the encoded signal when decoded with a legacy Opus decoder."

Next steps

- The MOC metric is currently implemented in Python with external library dependencies and should be implemented in C
- Current testvectors occasionally trigger PLC. Corresponding tests fail if different PLC method is used => add masking to MOC to eliminate PLC frames
- Decide on final thresholds
- Does the working group find the draft in its current form ready for adoption?

Thank you!