

LACE: a light-weight, causal model for enhancing coded speech through adaptive convolutions

Jan Bütche, Jean-Marc Valin, Ahmed Mustafa

Paper #61

Linear-Adaptive Coding Enhancer (LACE)

Task

Enhance decoded speech from existing codec (Opus), without breaking compatibility
Not trying to remove background noise

Constraints

Low complexity, no added delay, approximately phase/waveform preserving

Existing methods

- Classical methods: low complexity, no added delay, limited effectiveness
- DNN methods: high effectiveness, high complexity and/or added delay

Proposed Solution

Hybrid approach: mix DNNs with classical adaptive filtering

Why Is This Important?

Extend useful life of existing codecs

Some codecs already included in billions of devices

Deploying a new codec is slow and expensive

Time to user

Allows for incremental deployment

Faster/easier than a hard switch

Older/small devices

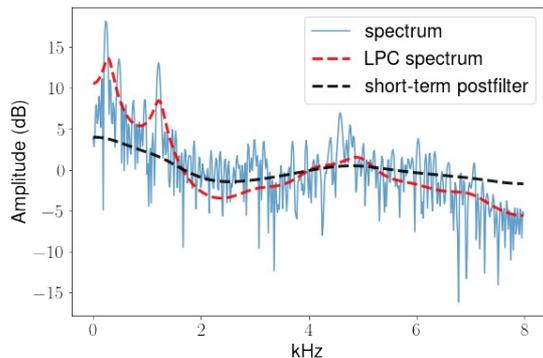
Not every device is capable of running an ML-based codec

Classical Speech Coding Enhancement: Noise Reduction with Adaptive Postfilters

[1] J.-H. Chen and A. Gersho, "Adaptive Postfiltering for Quality Enhancement of Coded Speech," *IEEE Transactions on Speech and Audio Processing*, vol. 3, no. 1, pp. 59–71, 1995.

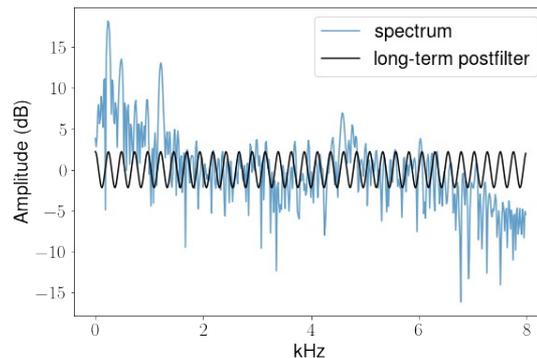
Formant Enhancement

- Attenuates noise in spectral valleys
- Adjusts spectral envelope
- Linear filter calculated on a per-frame basis using short-term prediction filter



Pitch Enhancement

- Attenuates noise between harmonics
- Reduces roughness
- Linear filter calculated on a per-frame basis using pitch period



Adaptive Postfilters with DNNs

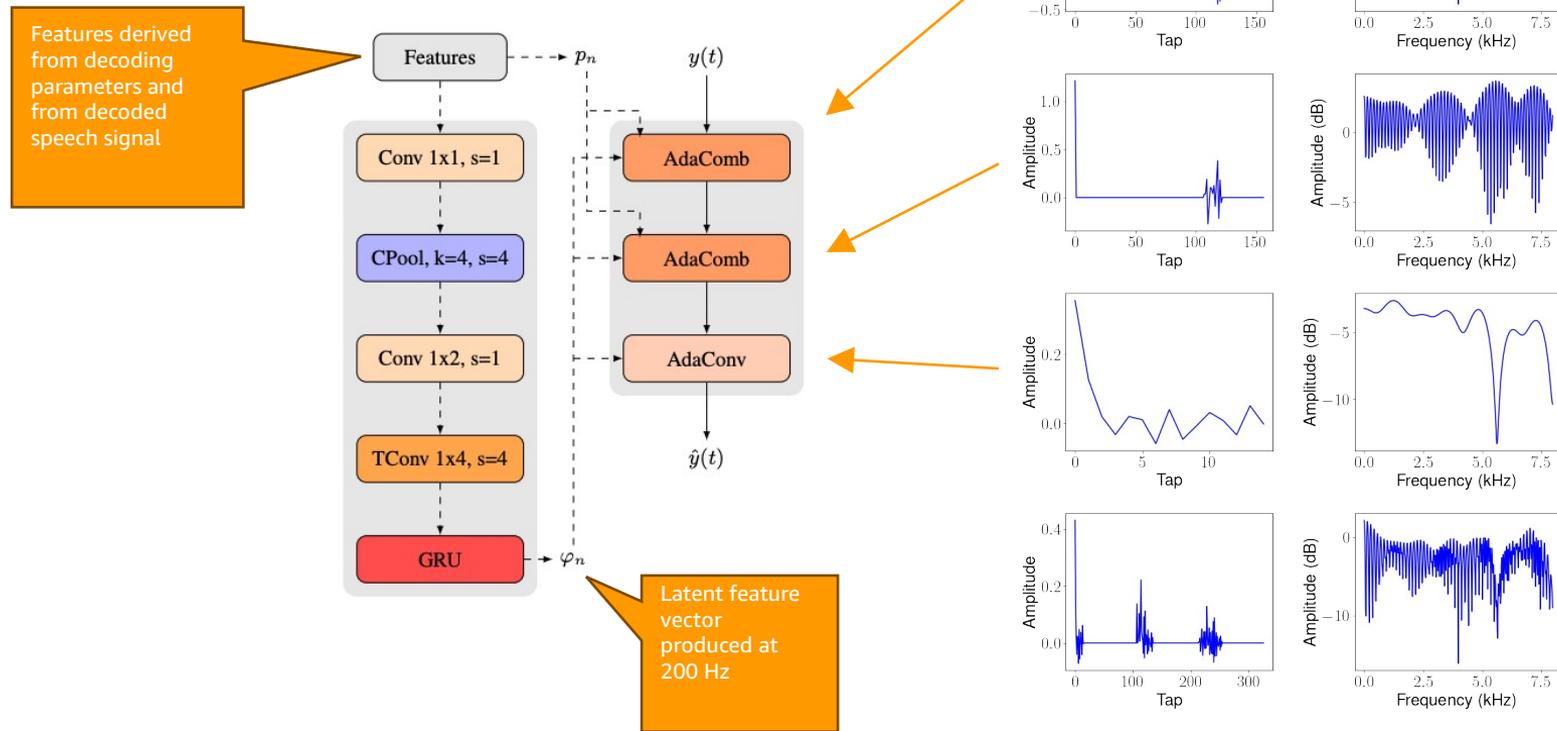
Challenges

- Convolutional layer weights are fixed after training
 - Not well suited for an adaptive postfilter
- Need scale-invariant processing
 - Behavior should not change based on signal amplitude

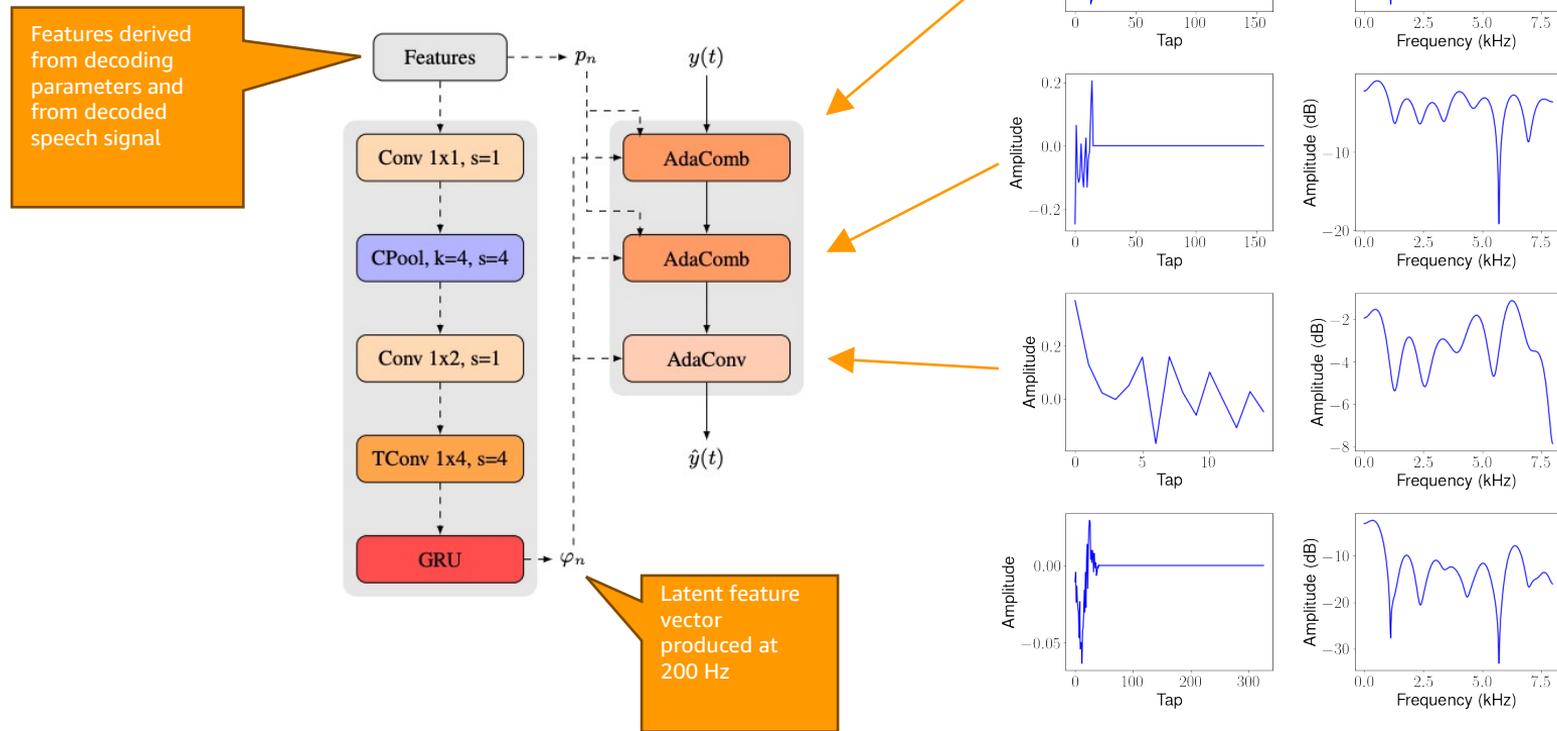
Solution

- Keep traditional filtering structure
- Compute filter coefficients with DNN
 - Allows for more powerful filters
 - Keeps complexity low

LACE: DNN-driven adaptive Filtering



LACE: DNN-driven adaptive Filtering



Practical Considerations

Using LACE in Opus (SILK)

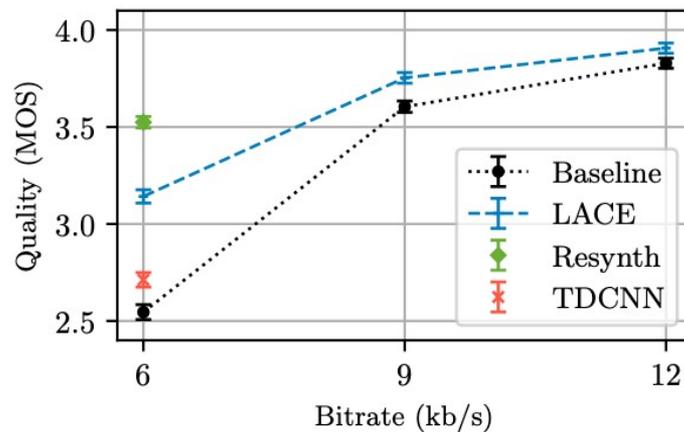
- Modern codecs support many rates, modes of operation
- LACE uses number of bits as additional input (can be variable)
- Randomize other operating points (blind)

P.808 Listening Test

Conditions

- Baseline: Opus WB, speech coding mode
- Benchmark: TDCNN from
 - [2] Z. Zhao, H. Liu, and T. Fingscheidt, “Convolutional Neural Networks to Enhance Coded Speech,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 27, no. 4, pp. 663–678, 2019.
- Reference point: LPCNet resynthesis. Non-causal model (25 ms lookahead), 3 GFLOPS
 - [3] J. Skoglund and J.-M. Valin, “Improving Opus Low Bit Rate Quality with Neural Speech Synthesis,” in *Proc. INTERSPEECH*, 2019.
- LACE: Baseline with LACE enhancement

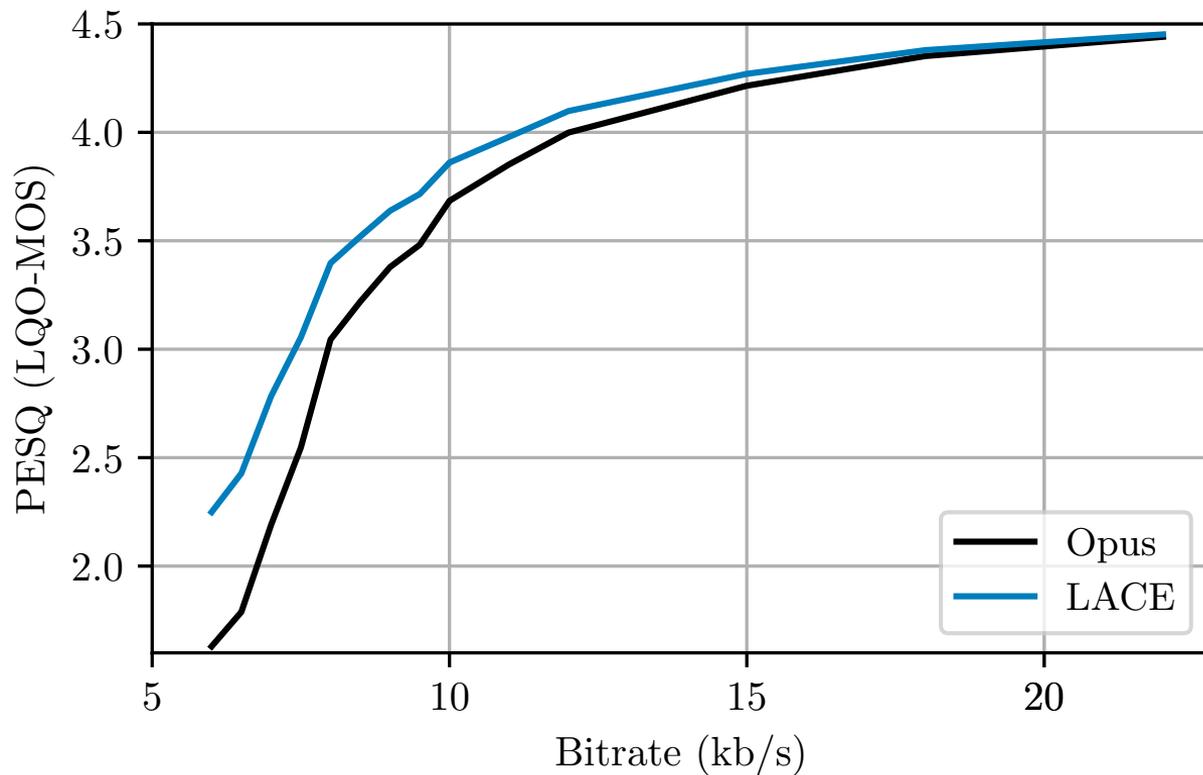
Results



Testing More Bitrates

PESQ-WB

- Agrees with MOS test
- Improvement at all bitrates



Summary

Linear-Adaptive Coding Enhancer

- New: single-channel adaptive convolutions with time-varying coefficients from DNN
- Lightweight: 300 K parameters and complexity of 100 MFLOPS
- Significantly improves Opus over a wide range of bitrates
- Plug-in solution: zero delay and waveform-preserving
- Standardization under way in new mlcodec WG at IETF

Resources

Demo samples



<https://282fd5fa7.github.io/LACE/>

LACE code



<https://gitlab.xiph.org/xiph/opus/-/tree/opus-ng/dnn/torch/osce>

LACE paper (arxiv)



<https://arxiv.org/abs/2307.06610>

New NoLACE paper (arxiv)



<https://arxiv.org/abs/2309.14521>