Inter-University Research Institute Corporation /
Research Organization of Information and Systems
National Institute of Informatics

Speech synthesis workshop
Vienna, Austria, 2019

Harmonic-plus-noise neural source-filter waveform model with trainable maximum-voiced frequency

Xin WANG, Junichi YAMAGISHI
National Institute of Informatics, Japan

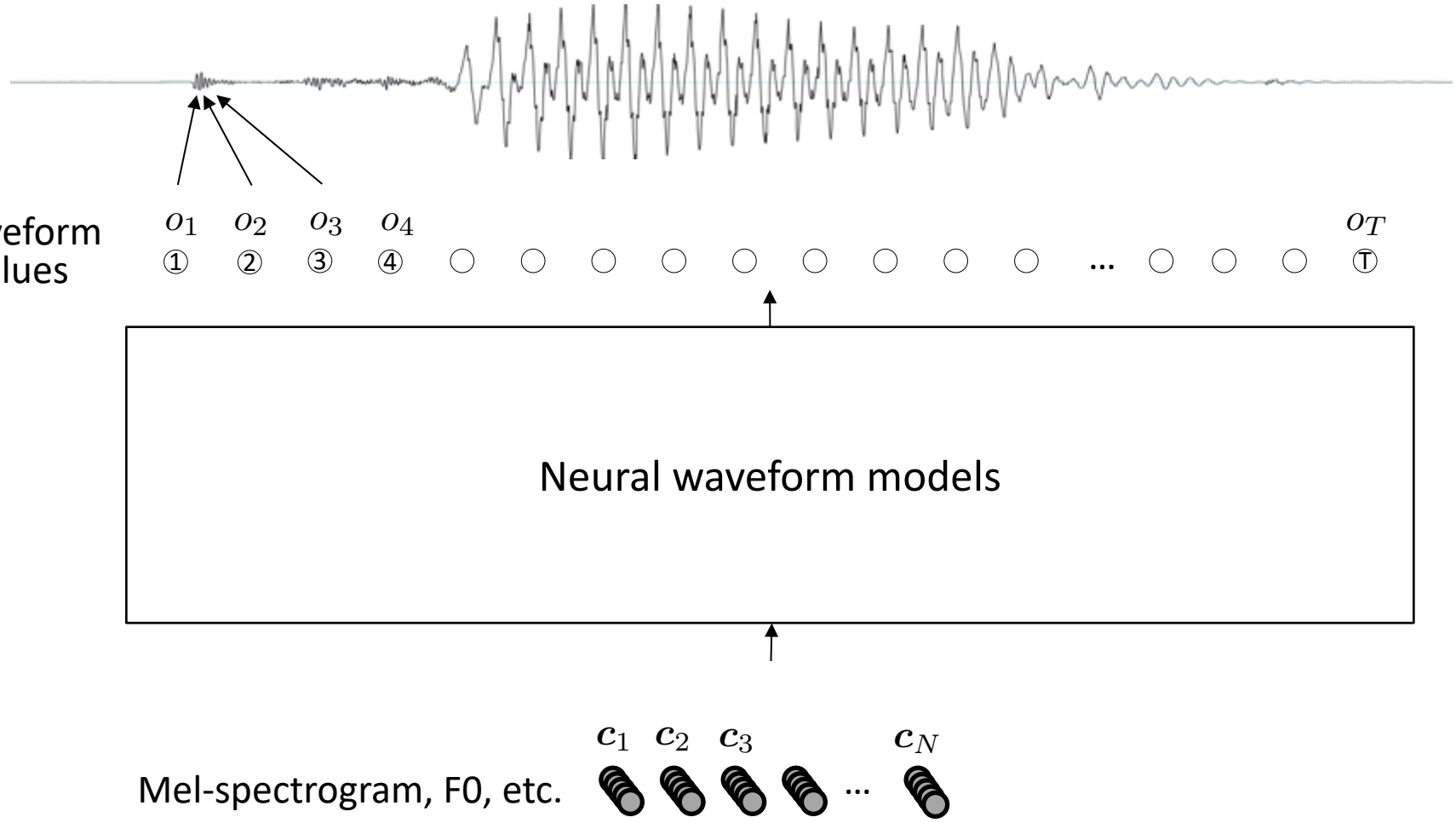
contact: wangxin@nii.ac.jp
we welcome critical comments, suggestions, and discussion

CONTENTS

- Introduction
- Proposed model
- Experiments
- Summary

INTRODUCTION

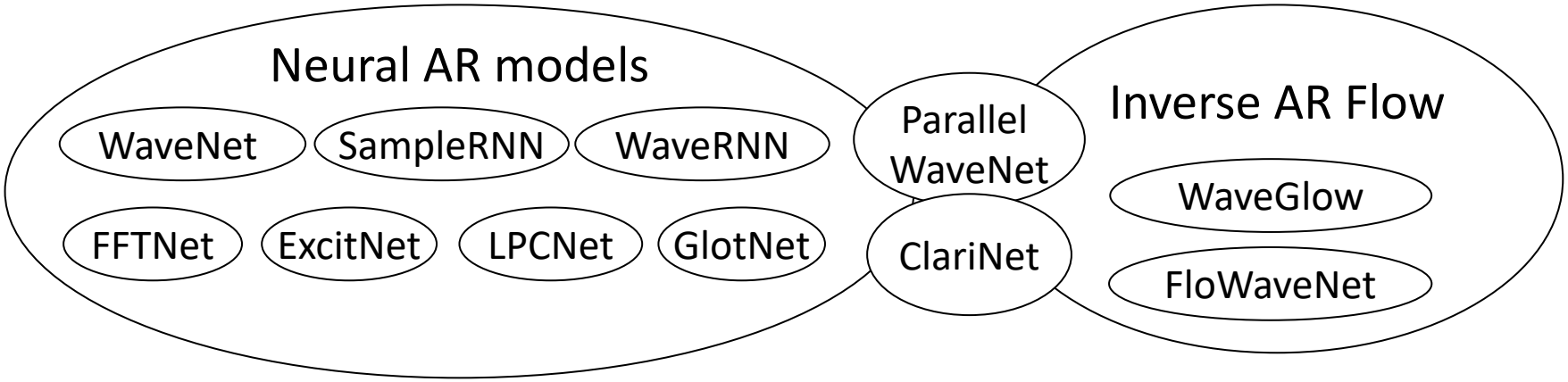
Task



INTRODUCTION

Models

- Neural autoregressive (AR) and inverse AR flow



- No AR or inverse AR flow

Neural source-filter (NSF) models

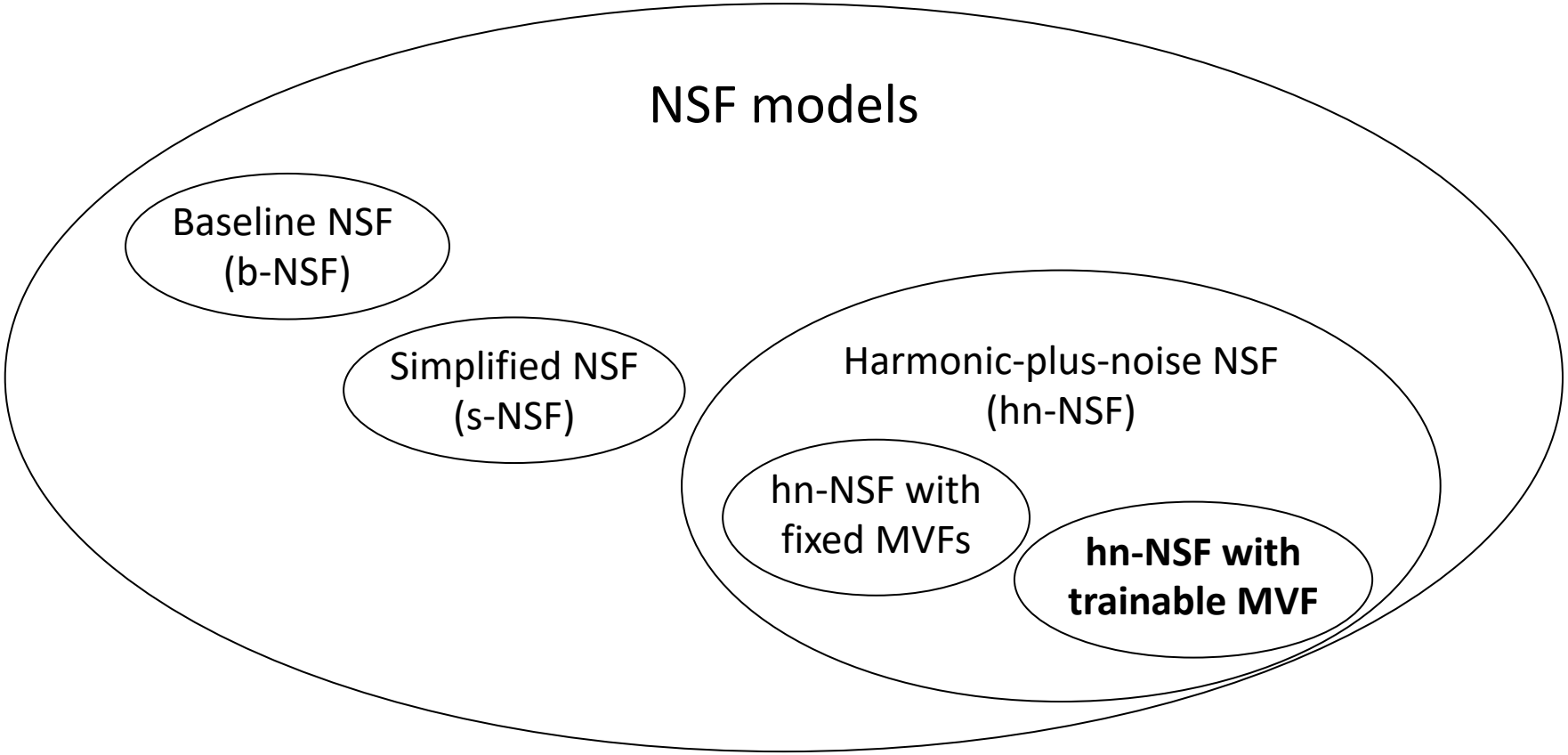
Multi-head CNN

GELP

- Spectral-domain training criterion
- Source-filter architecture

☛ Reference in appendix
☛ More details: https://www.slideshare.net/jyamagis/

INTRODUCTION



ICASSP 2019 | Journal paper submitted | **SSW 2019**

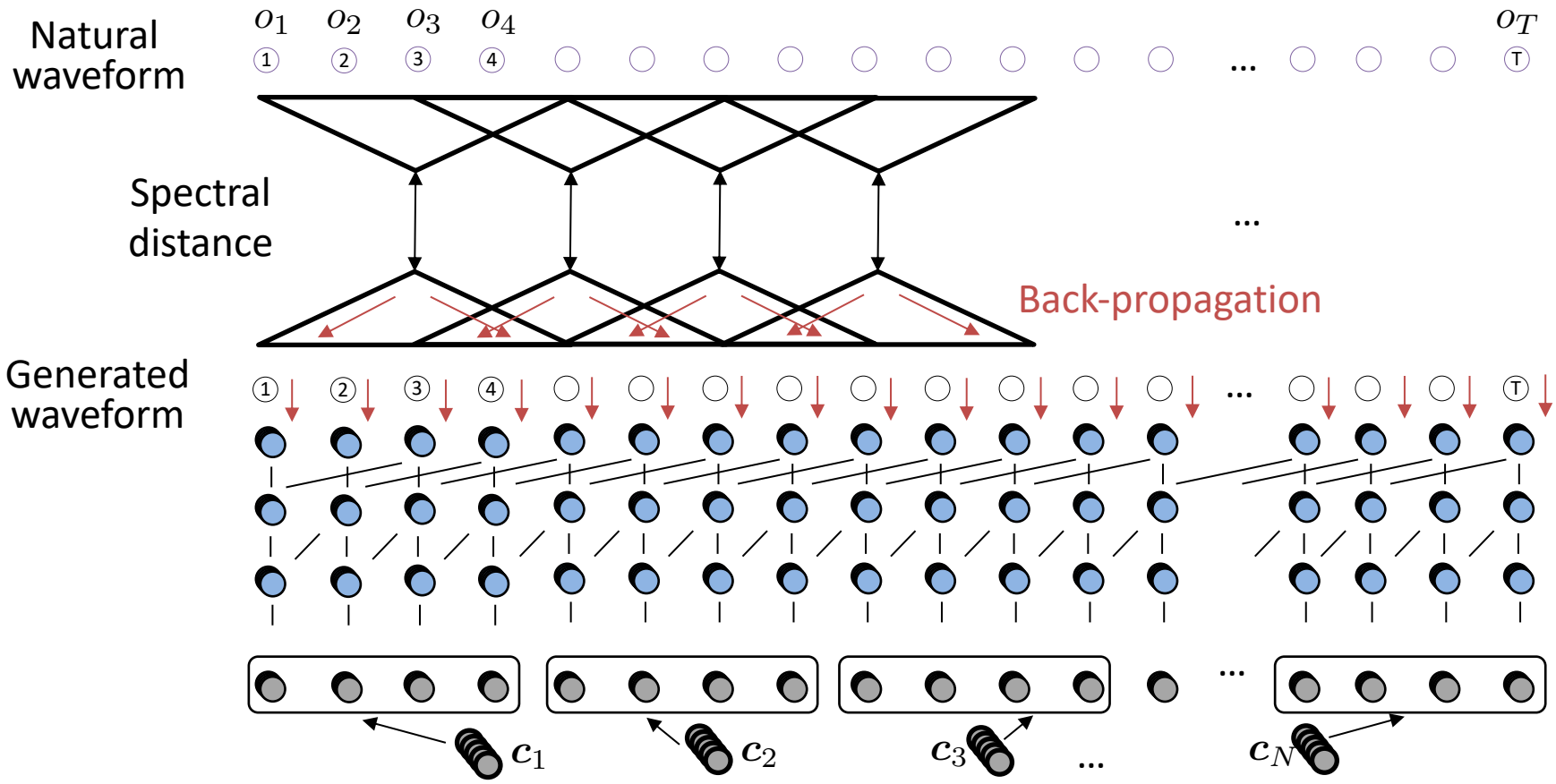
❖ MVF: Maximum voiced frequency

CONTENTS

- Introduction
- NSF model
- Experiments
- Summary

NEURAL SOURCE-FILTER MODEL

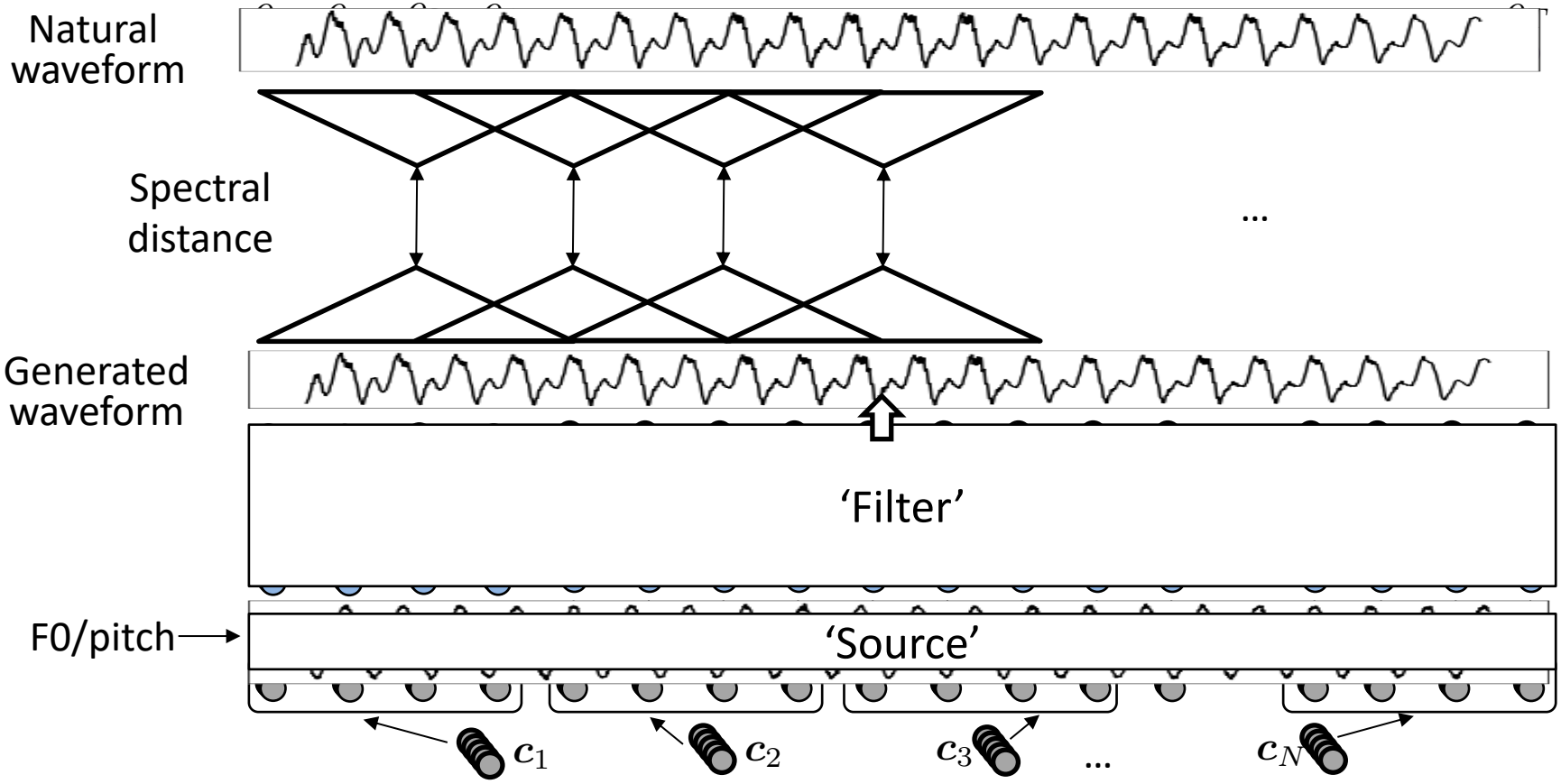
Idea 1: spectral domain criterion



- Based on short time Fourier transform (STFT)

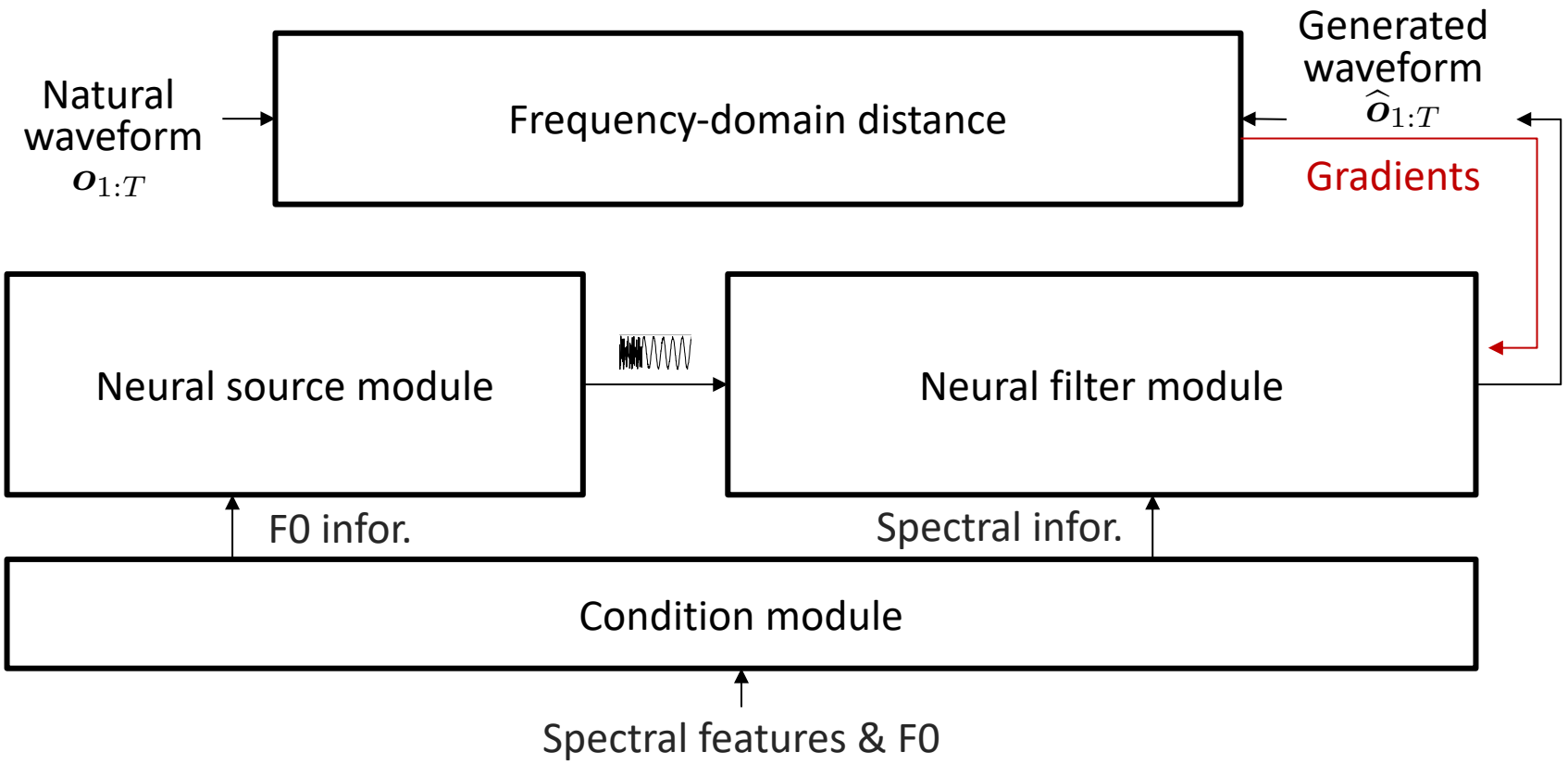
NEURAL SOURCE-FILTER MODEL

Idea 2: source-filter



NEURAL SOURCE-FILTER MODEL

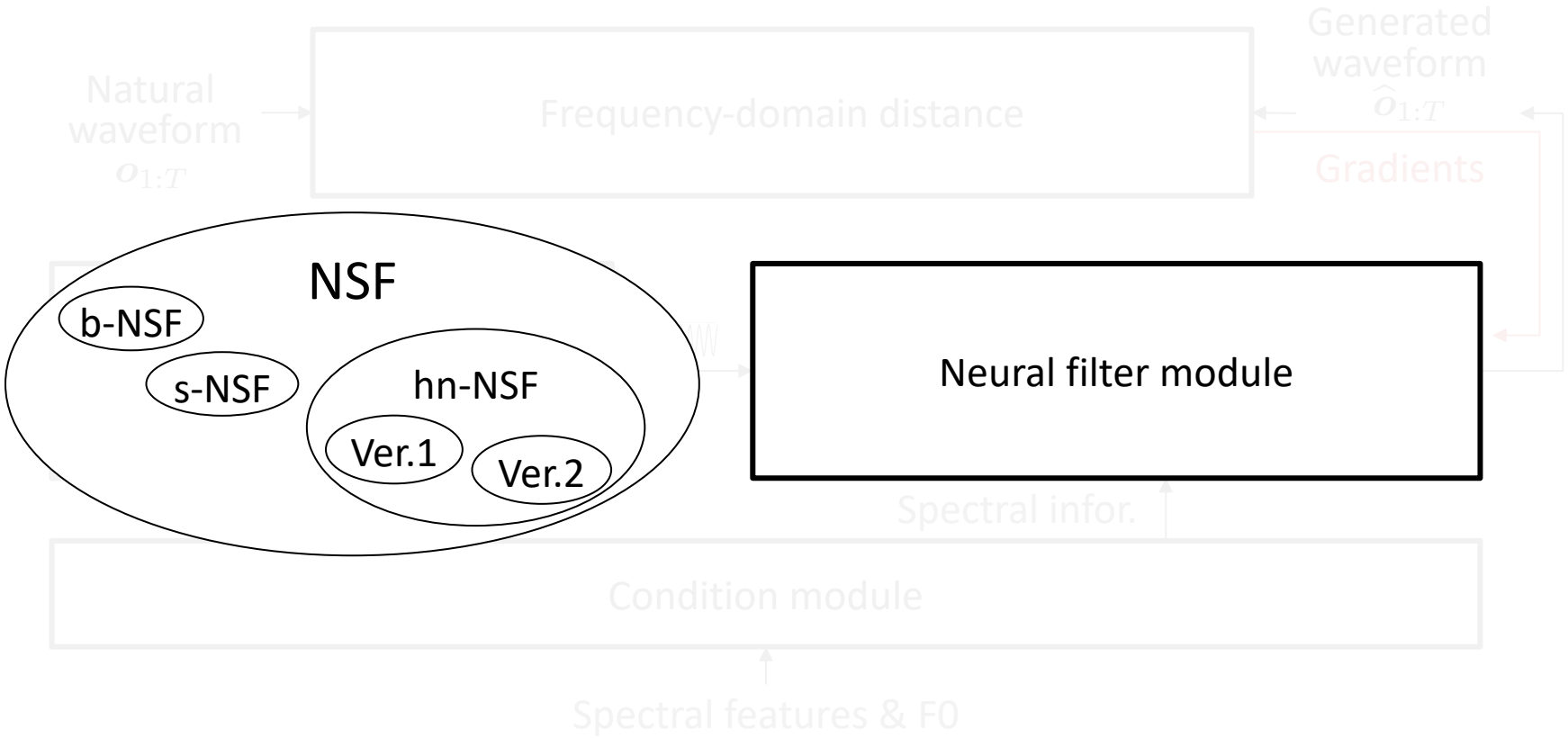
General framework



- No AR or inverse AR flow

NEURAL SOURCE-FILTER MODEL

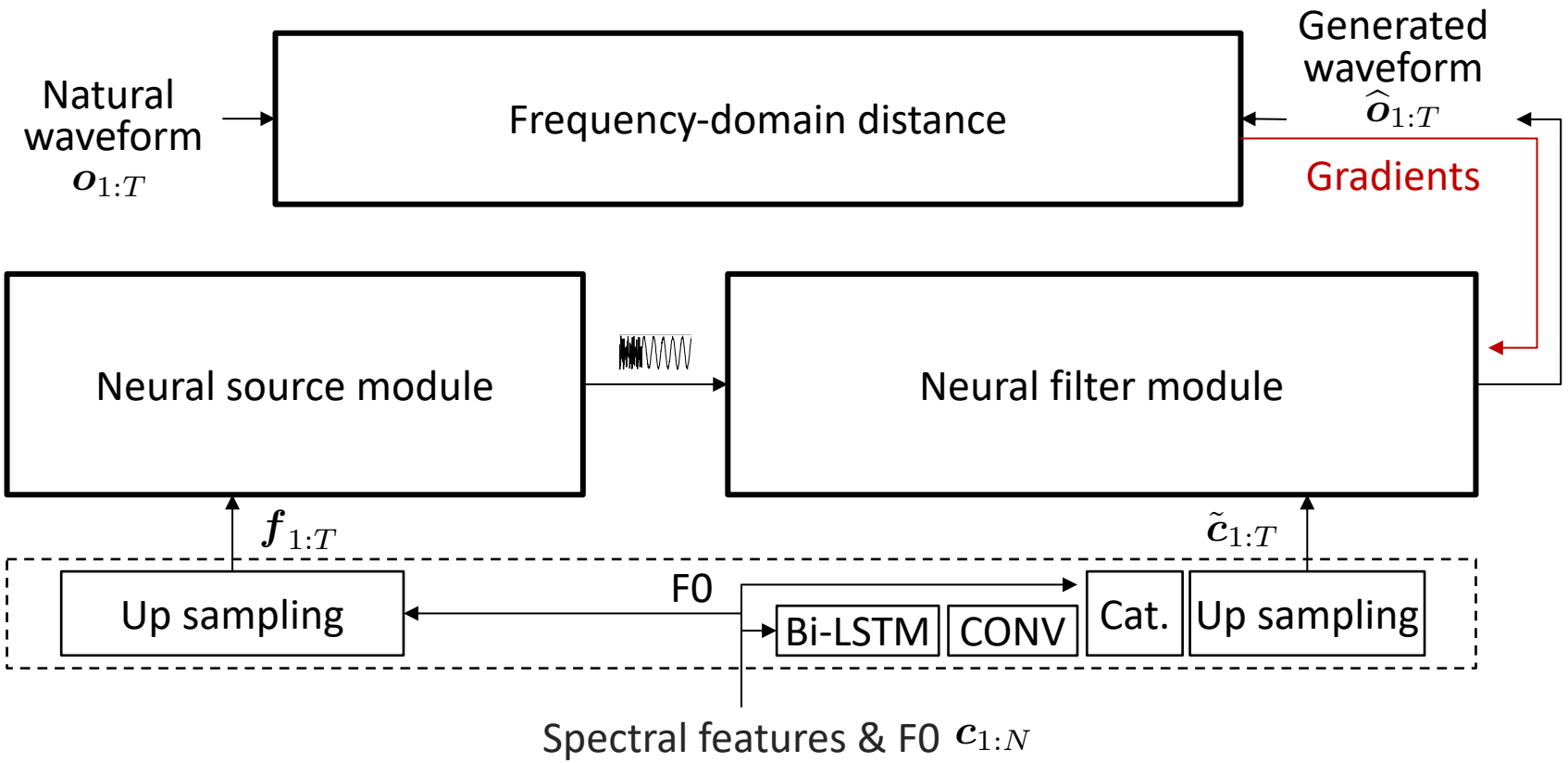
General framework



- Different neural filter modules

NEURAL SOURCE-FILTER MODEL

General framework

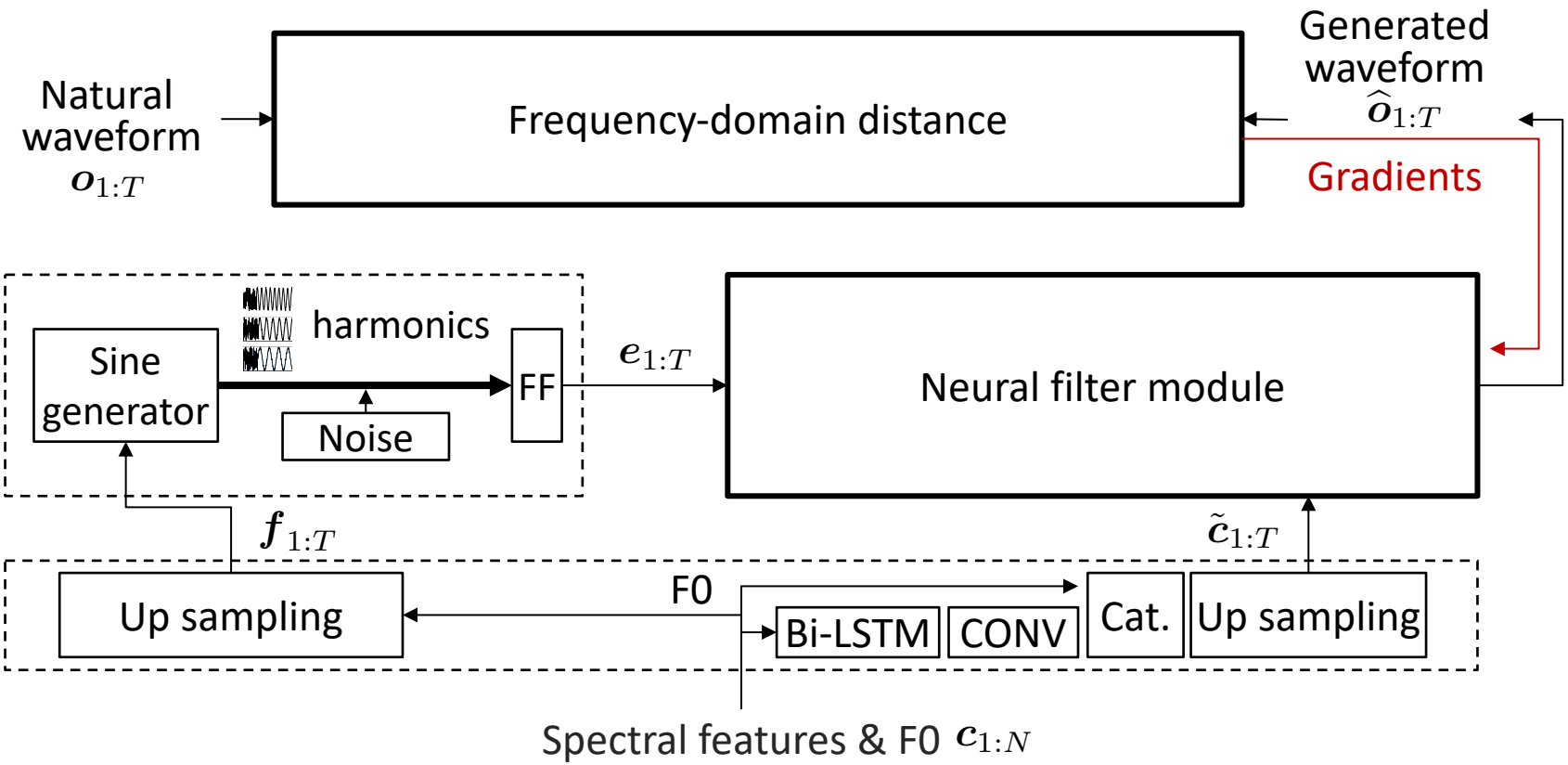


- Condition module:
  - Up sampling
  - Dimension change

- CONV: convolution
- Cat.: concatenation

NEURAL SOURCE-FILTER MODEL

General framework



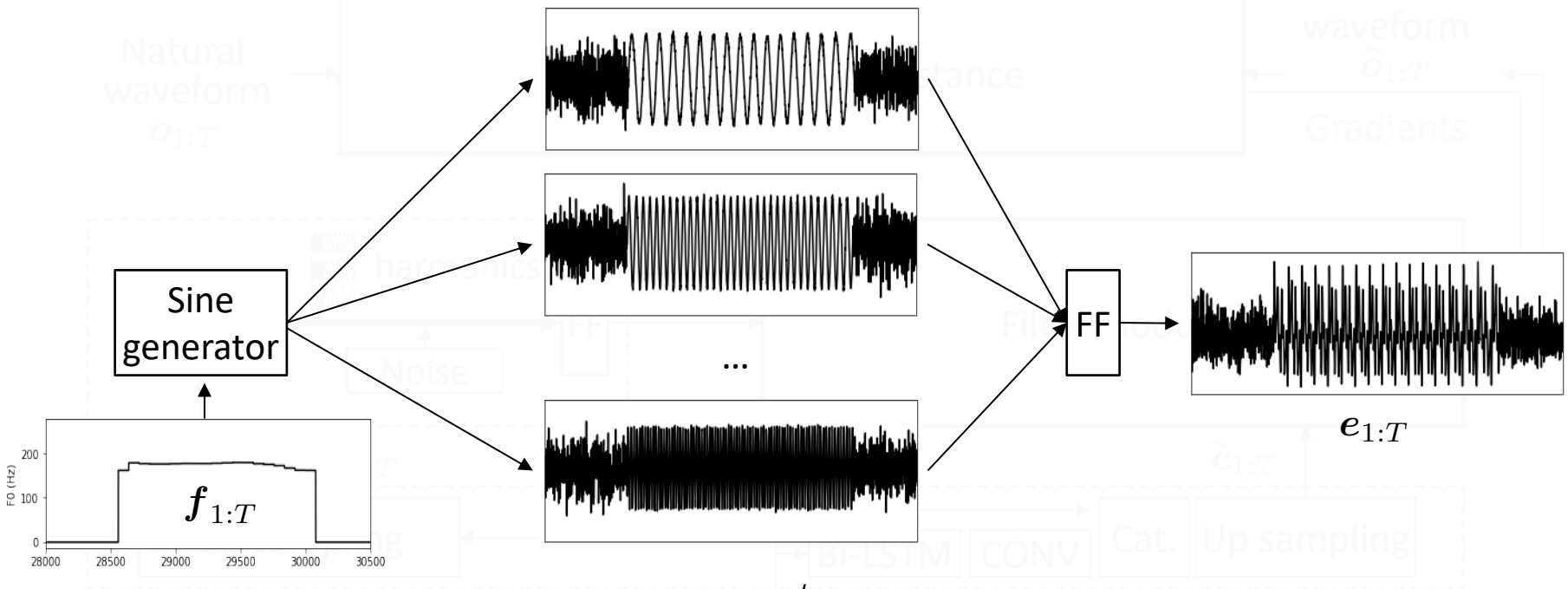
- Source module: generate sine-based excitation given F0

$$f_t \in \{0\} \cup \mathbb{R}^+ \longrightarrow e_t \in \mathbb{R}, \forall t \in \{1, \cdots, T\}$$

- FF: feedforward layer with Tanh

NEURAL SOURCE-FILTER MODEL

General framework



Fundamental component

Voiced: $\sin(\sum_{k=1}^{t} 2\pi \frac{f_k}{N_s} + \phi) + n_t$

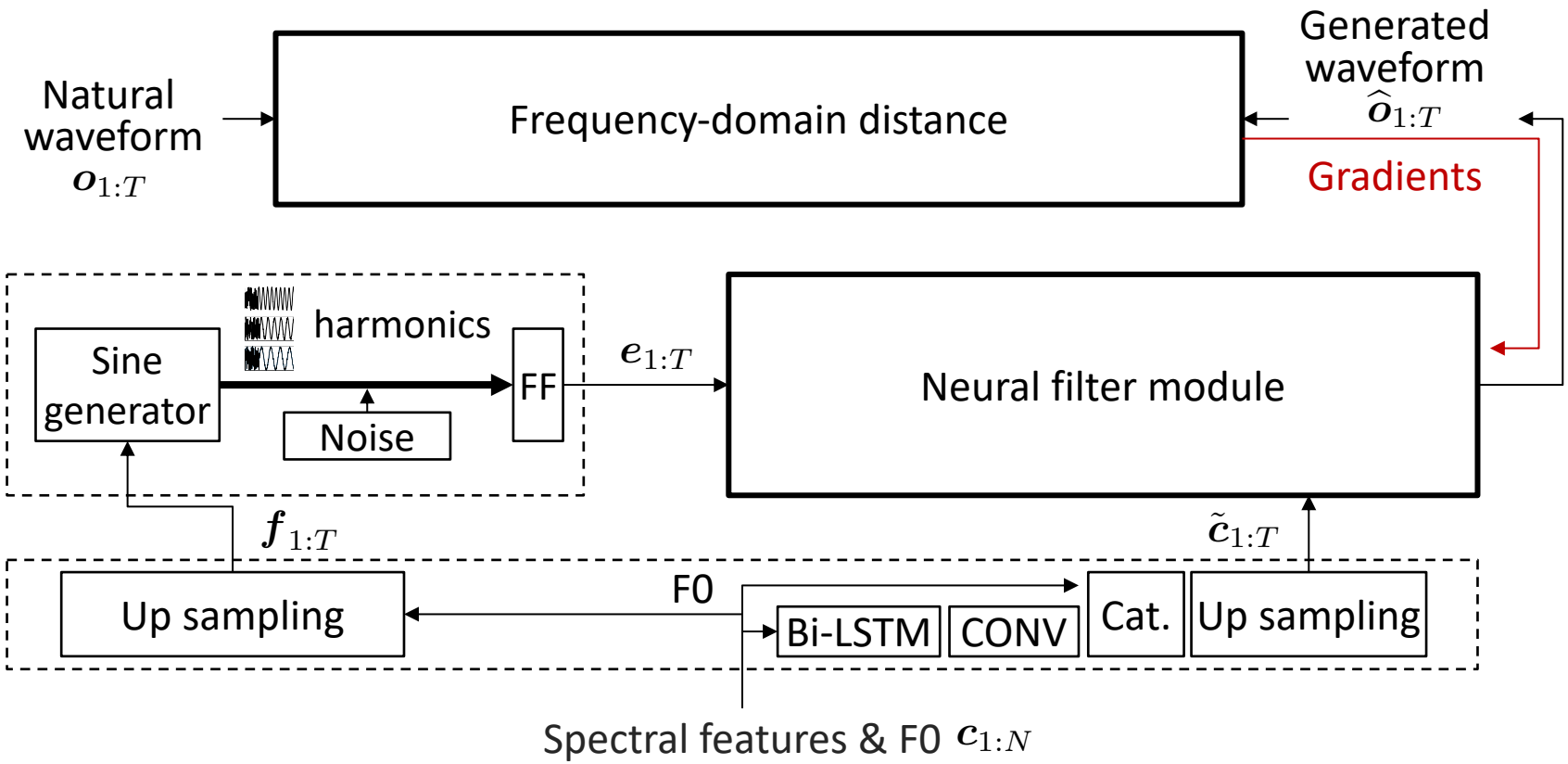
- n_t → Noise
- ϕ → Random initial phase
- $\frac{f_k}{N_s}$ → Sampling rate

Unvoiced: noise

- FF: feedforward layer with Tanh

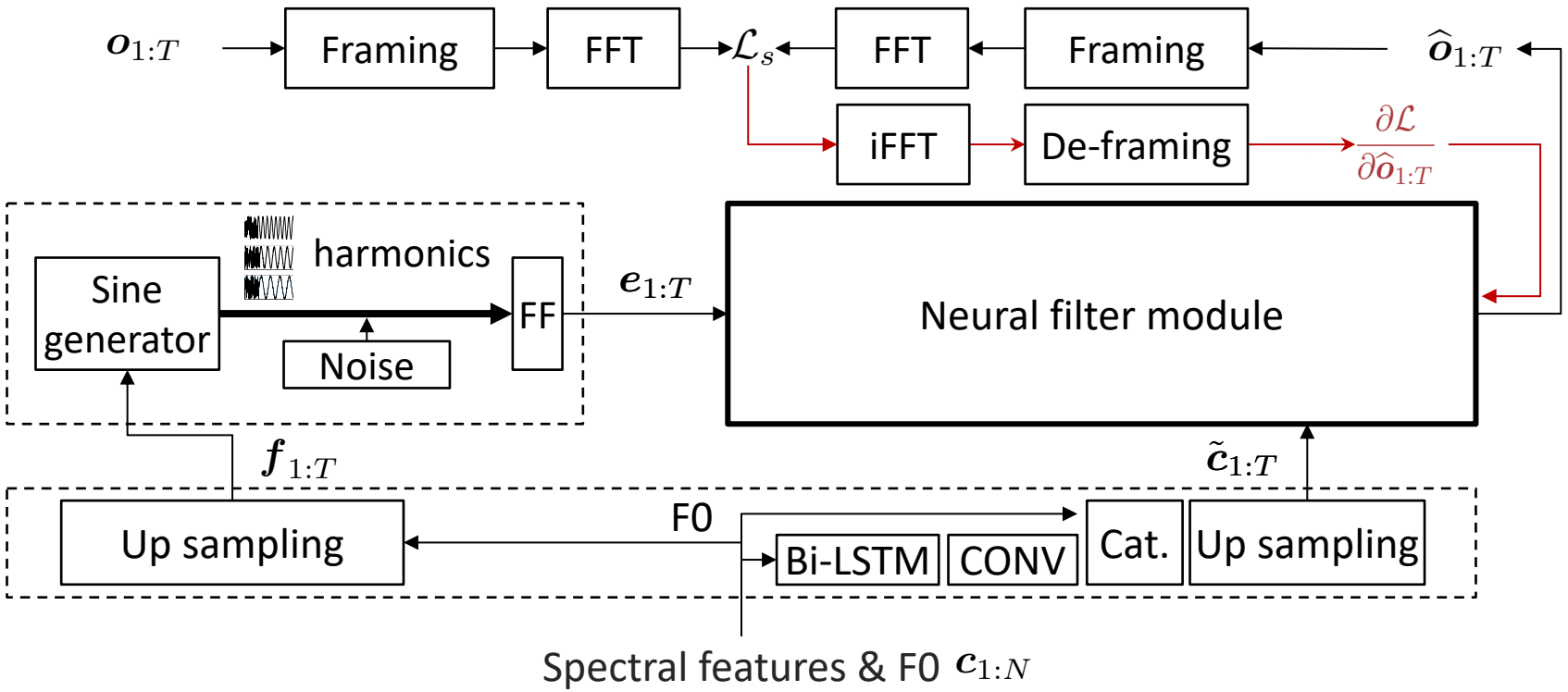
NEURAL SOURCE-FILTER MODEL

General framework



NEURAL SOURCE-FILTER MODEL

General framework

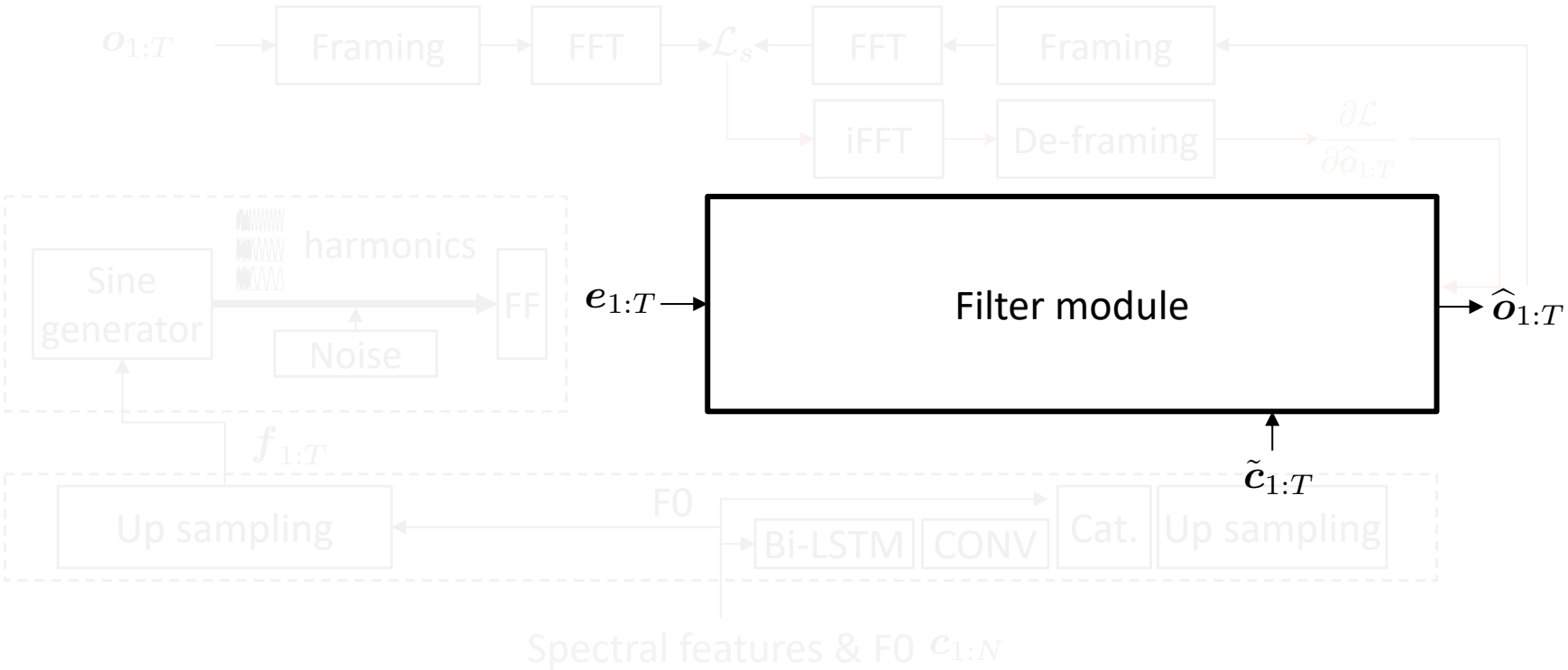


- Multiple L_s : different frame shift & length (☛ ICASSP 2019)

❖ FFT: fast Fourier transform

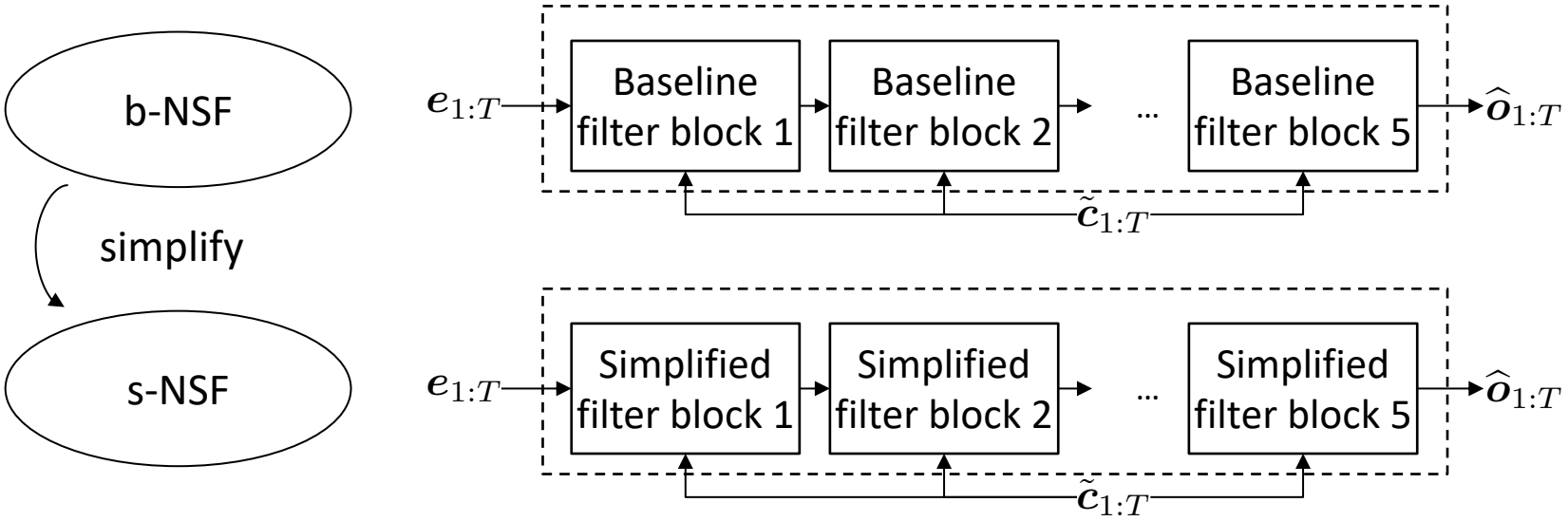
NEURAL SOURCE-FILTER MODEL

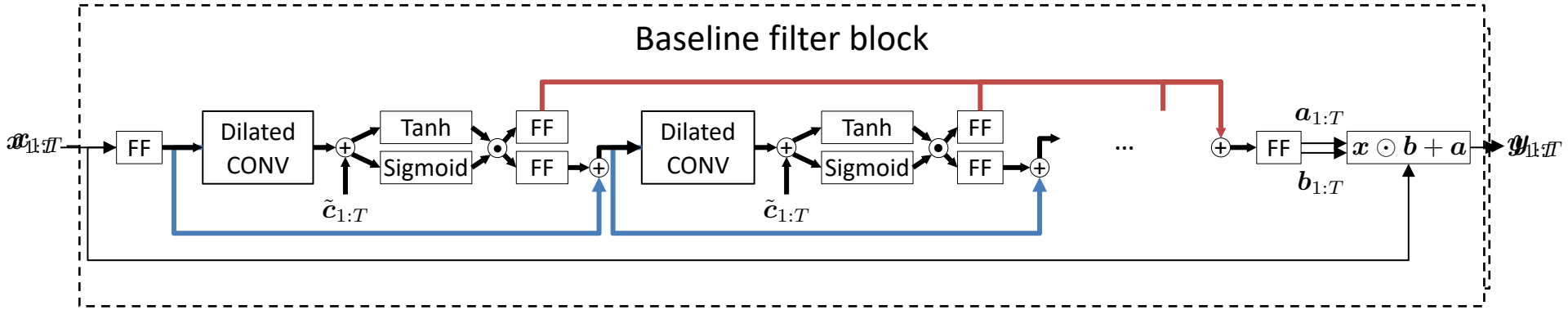
General framework



NEURAL SOURCE-FILTER MODEL

Filter modules in NSF models

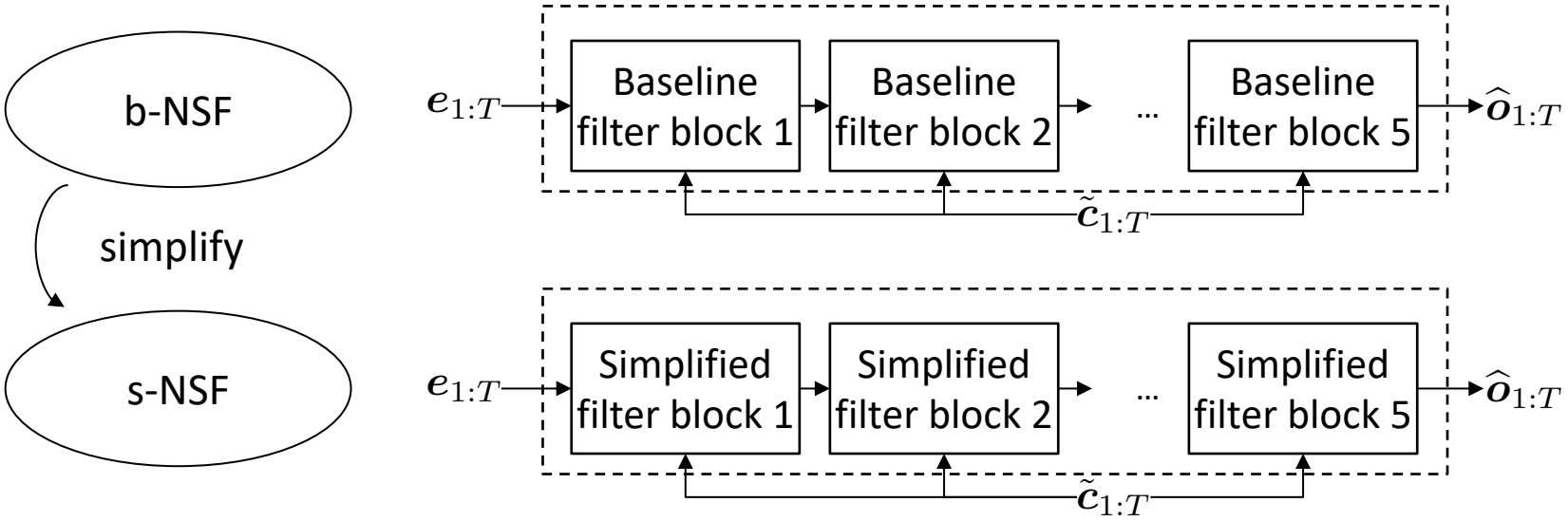




- $x_t, y_t, \widehat{o}_t, a_t \in \mathbb{R},\ b_t \in \mathbb{R}^+,\ \tilde{\boldsymbol{c}}_t \in \mathbb{R}^{64},\ \forall t \in \{1, \cdots, T\}$
- Element-wise multiplication $\odot$

NEURAL SOURCE-FILTER MODEL

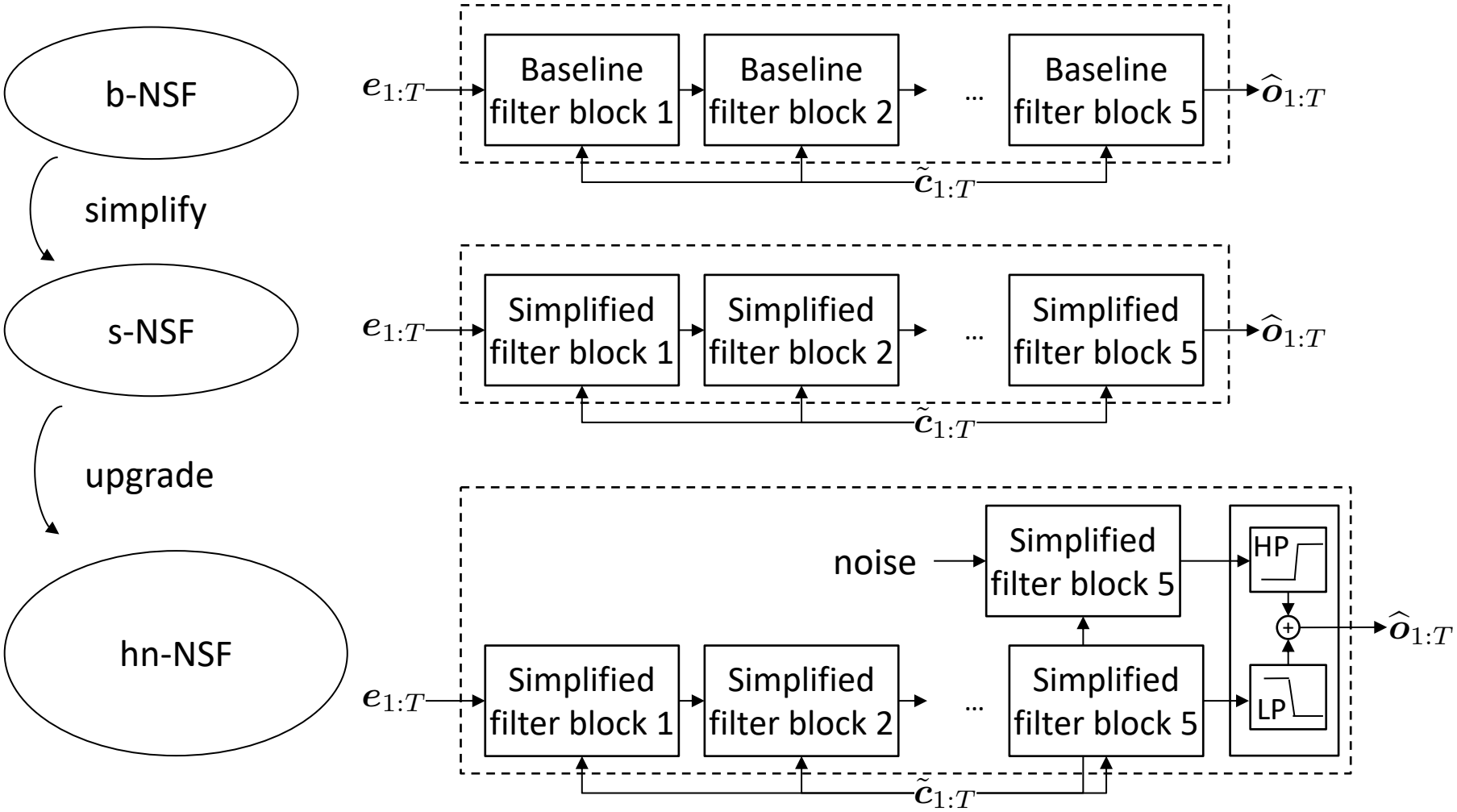
Filter modules in NSF models



- Artifacts in both models (☛ journal paper):
  1. Strong harmonics in high-frequency band
  2. Bad unvoiced sounds
  3. …

NEURAL SOURCE-FILTER MODEL

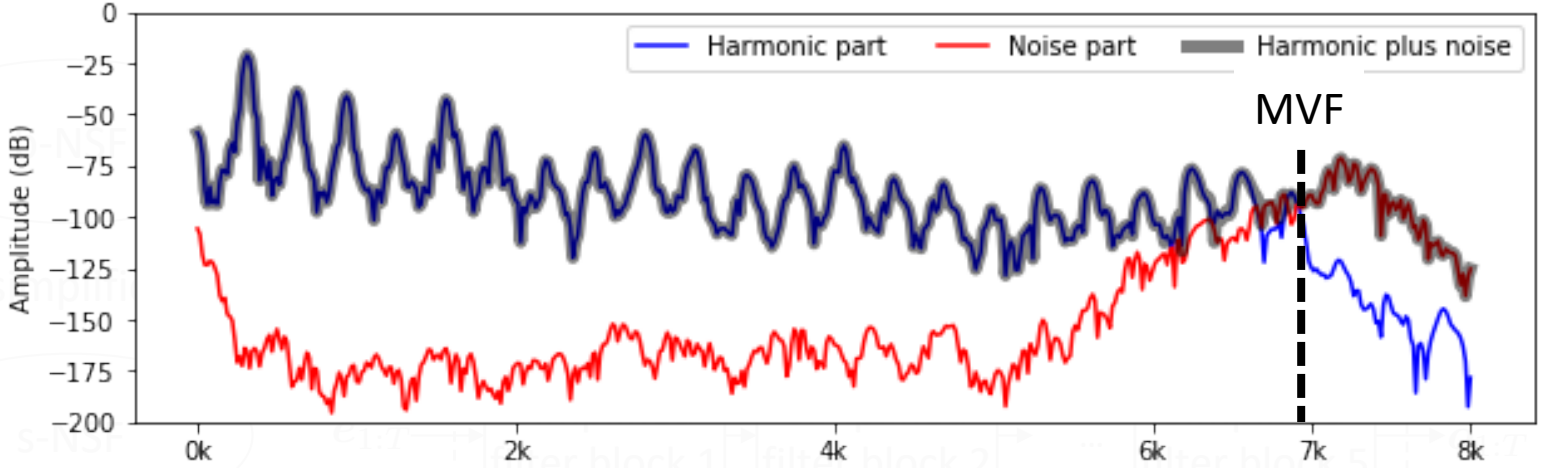
Filter modules in NSF models



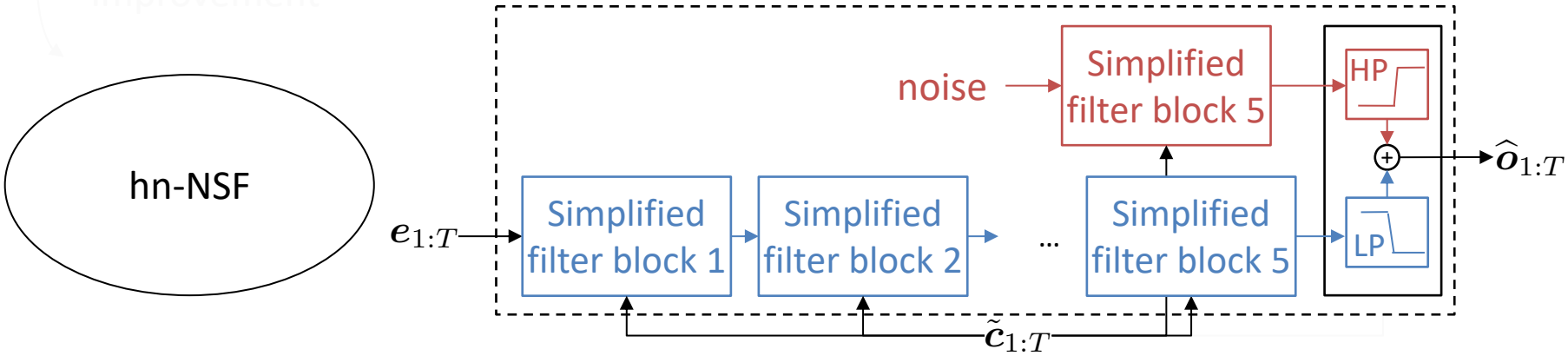
- HP, LP: high- and low-pass finite-impulse-response (FIR) filter

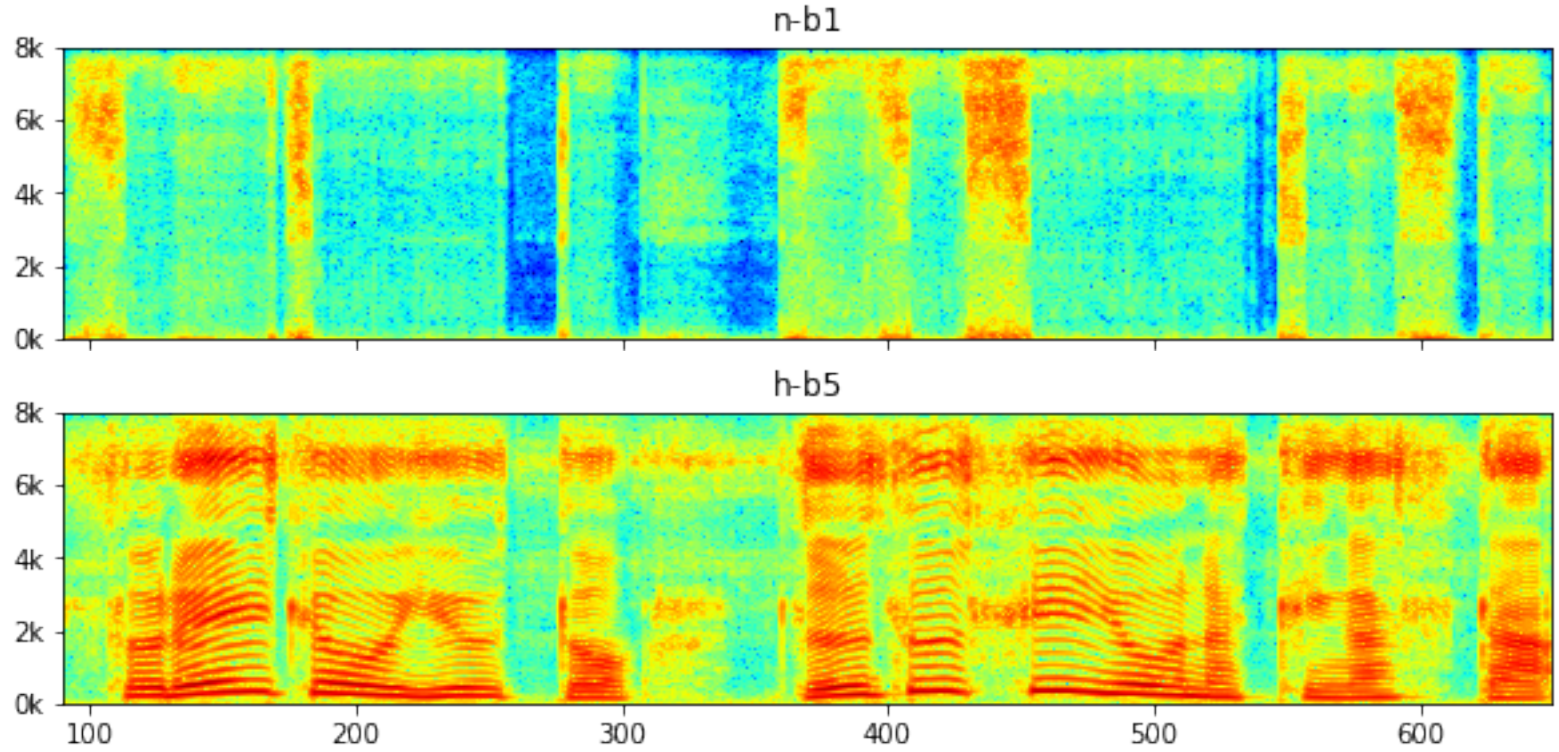
NEURAL SOURCE-FILTER MODEL

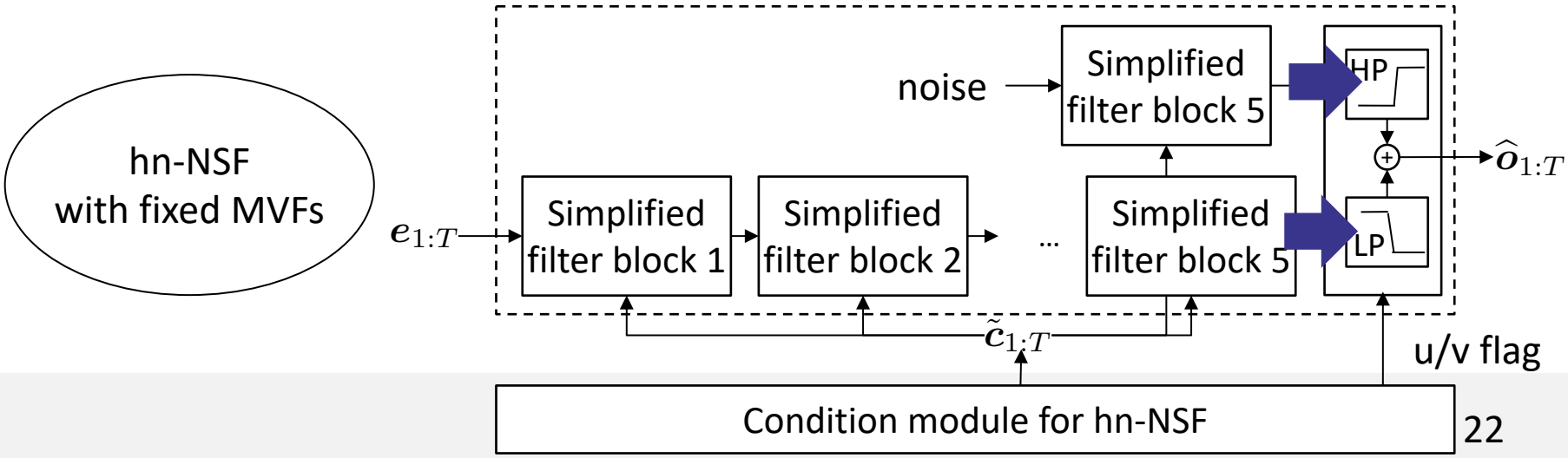
Harmonic-plus-noise NSF

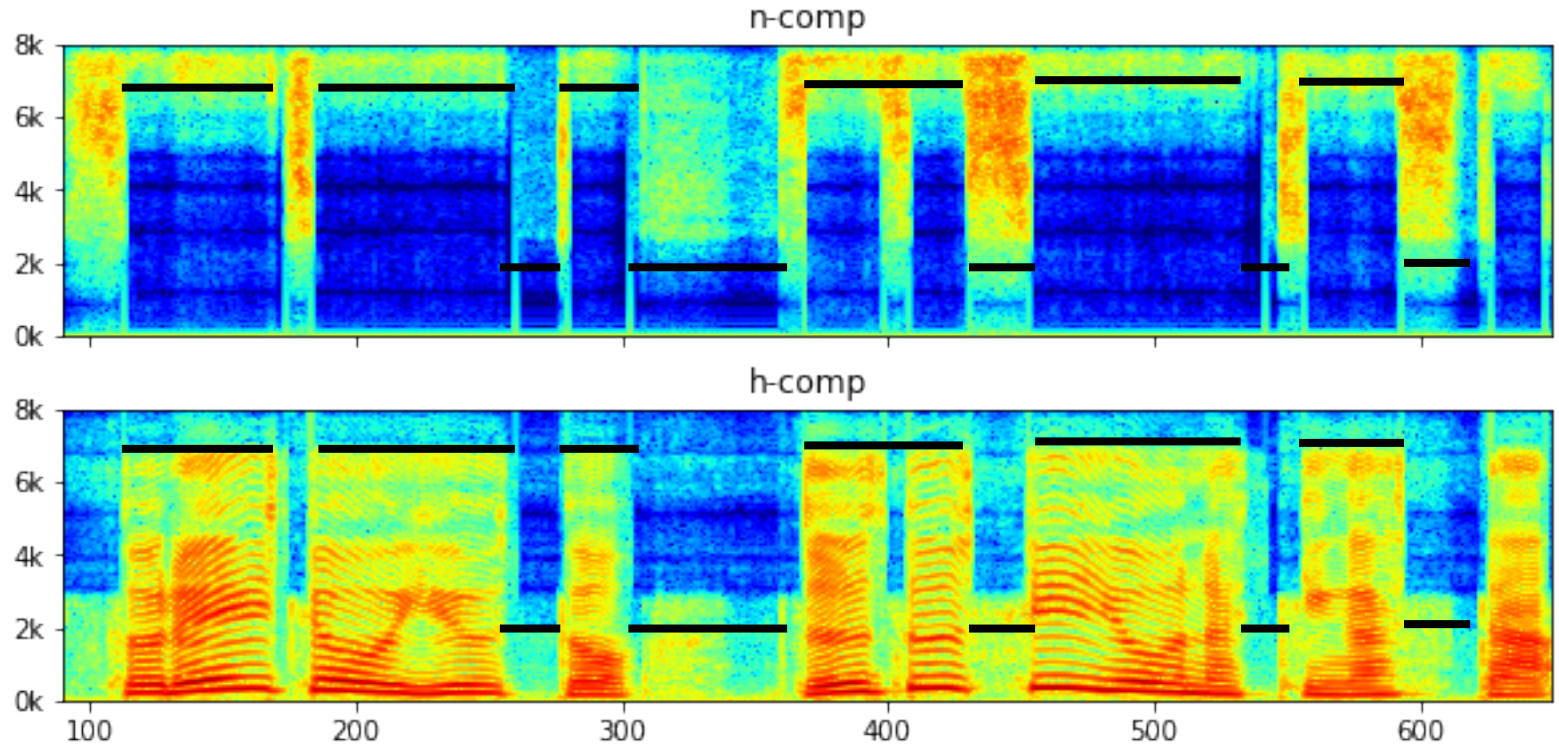


- Filtering in time domain




n-b1
8k
6k
4k
2k
0k
h-b5
8k
6k
4k
2k
0k
100
200
300
400
500
600


hn-NSF
with fixed MVFs
$e_{1:T}$
Simplified
filter block 1
Simplified
filter block 2
...
noise
Simplified
filter block 5
Simplified
filter block 5
HP
LP
$\hat{o}_{1:T}$
$\tilde{c}_{1:T}$
u/v flag
Condition module for hn-NSF



hn-NSF
with fixed MVFs

$e_{1:T}$

Simplified filter block 1

Simplified filter block 2

...

Simplified filter block 5

noise

Simplified filter block 5

HP

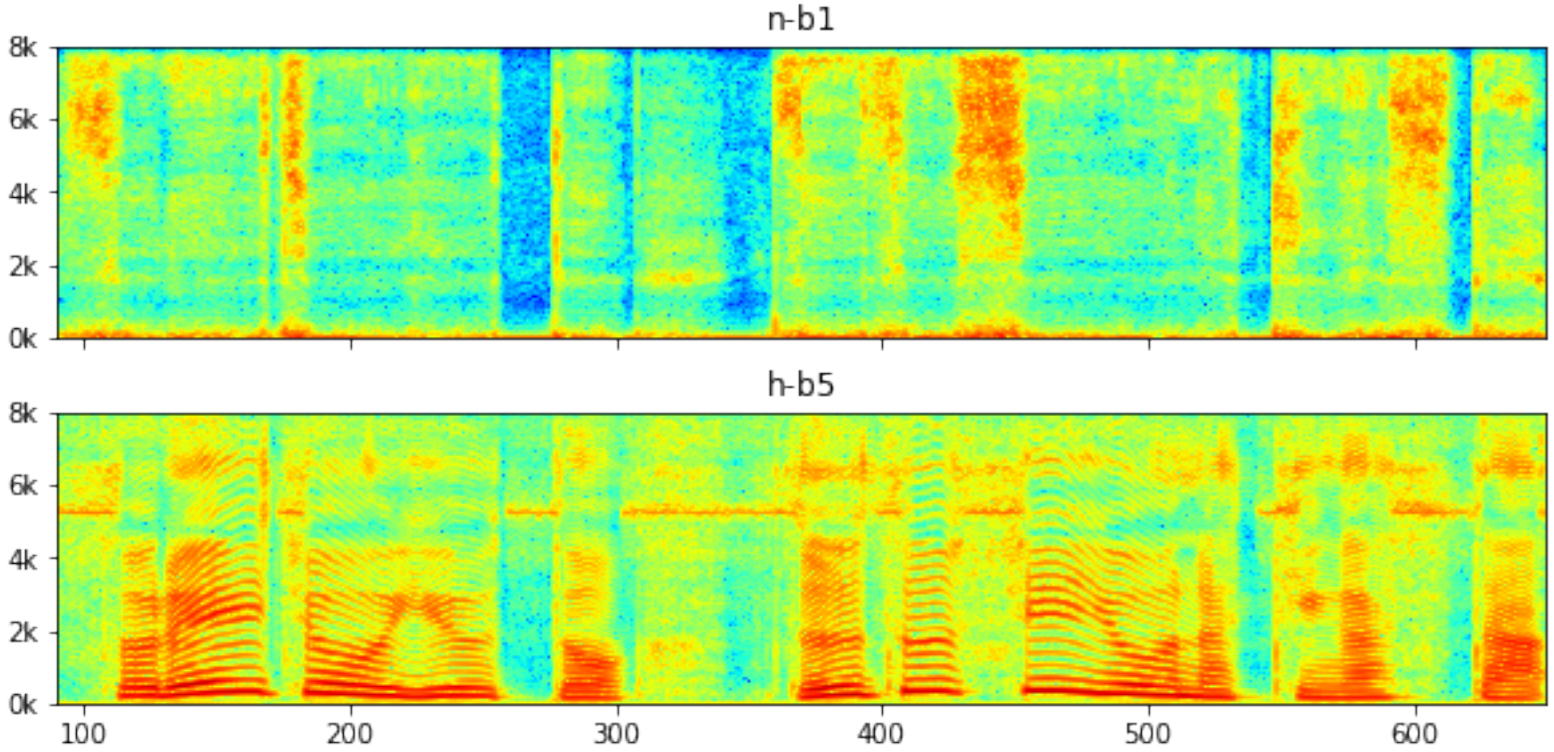
LP

$\hat{o}_{1:T}$

$\tilde{c}_{1:T}$

u/v flag

Condition module for hn-NSF



hn-NSF
with trainable MVFs

$e_{1:T}$

noise

Simplified filter block 1

Simplified filter block 2

...

Simplified filter block 5

Simplified filter block 5

HP

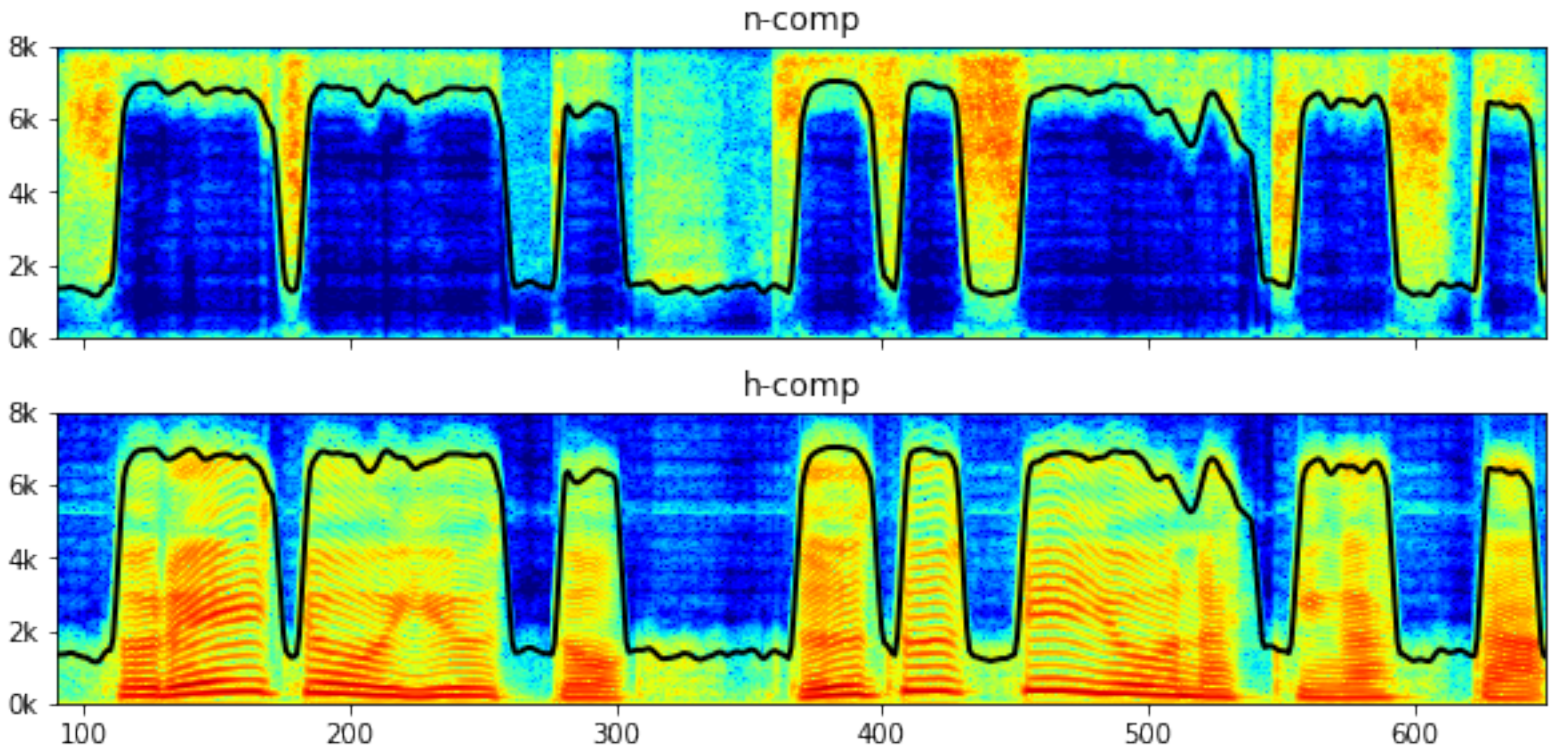
LP

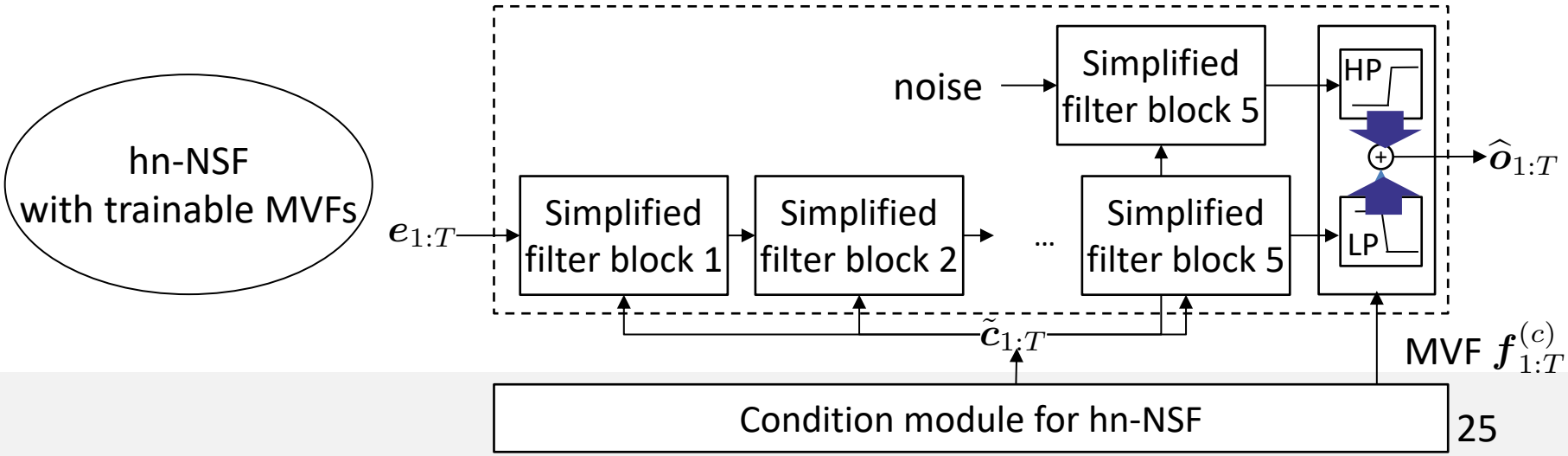
$\widehat{o}_{1:T}$

$\tilde{c}_{1:T}$

MVF $f^{(c)}_{1:T}$

Condition module for hn-NSF

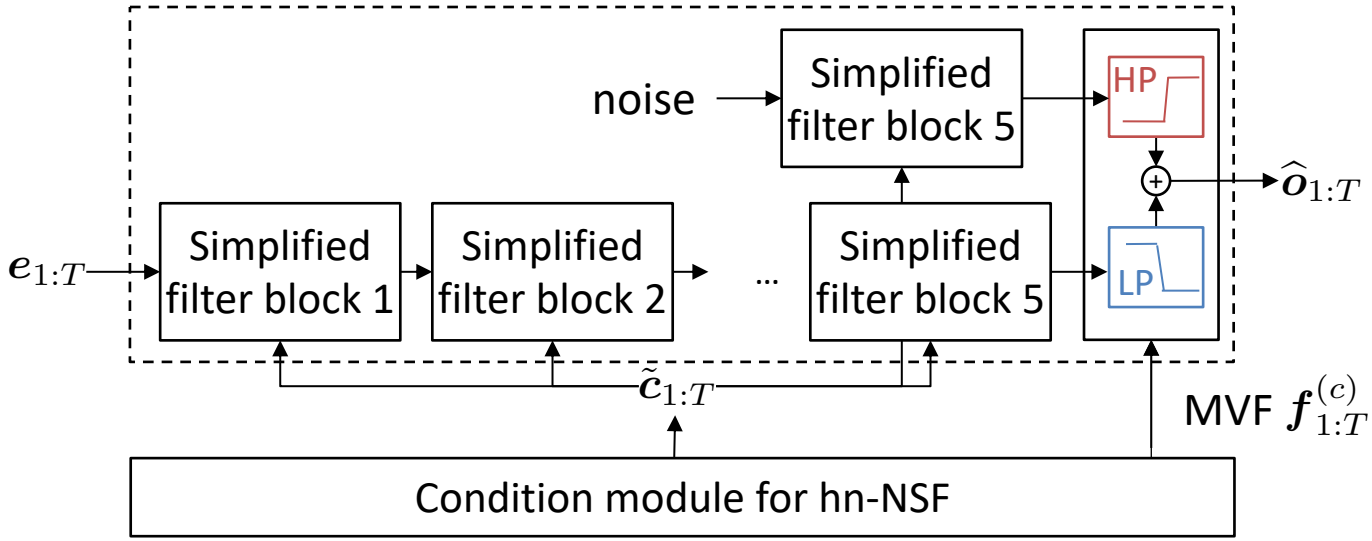

n-comp
8k
6k
4k
2k
0k
h-comp
8k
6k
4k
2k
0k
100
200
300
400
500
600


hn-NSF
with trainable MVFs
noise
Simplified filter block 5
HP
$\widehat{\boldsymbol{o}}_{1:T}$
$\boldsymbol{e}_{1:T}$
Simplified filter block 1
Simplified filter block 2
...
Simplified filter block 5
LP
$\tilde{\boldsymbol{c}}_{1:T}$
MVF $\boldsymbol{f}_{1:T}^{(c)}$
Condition module for hn-NSF

NEURAL SOURCE-FILTER MODEL

Harmonic-plus-noise NSF

- Version 2 (☛ ssw paper)

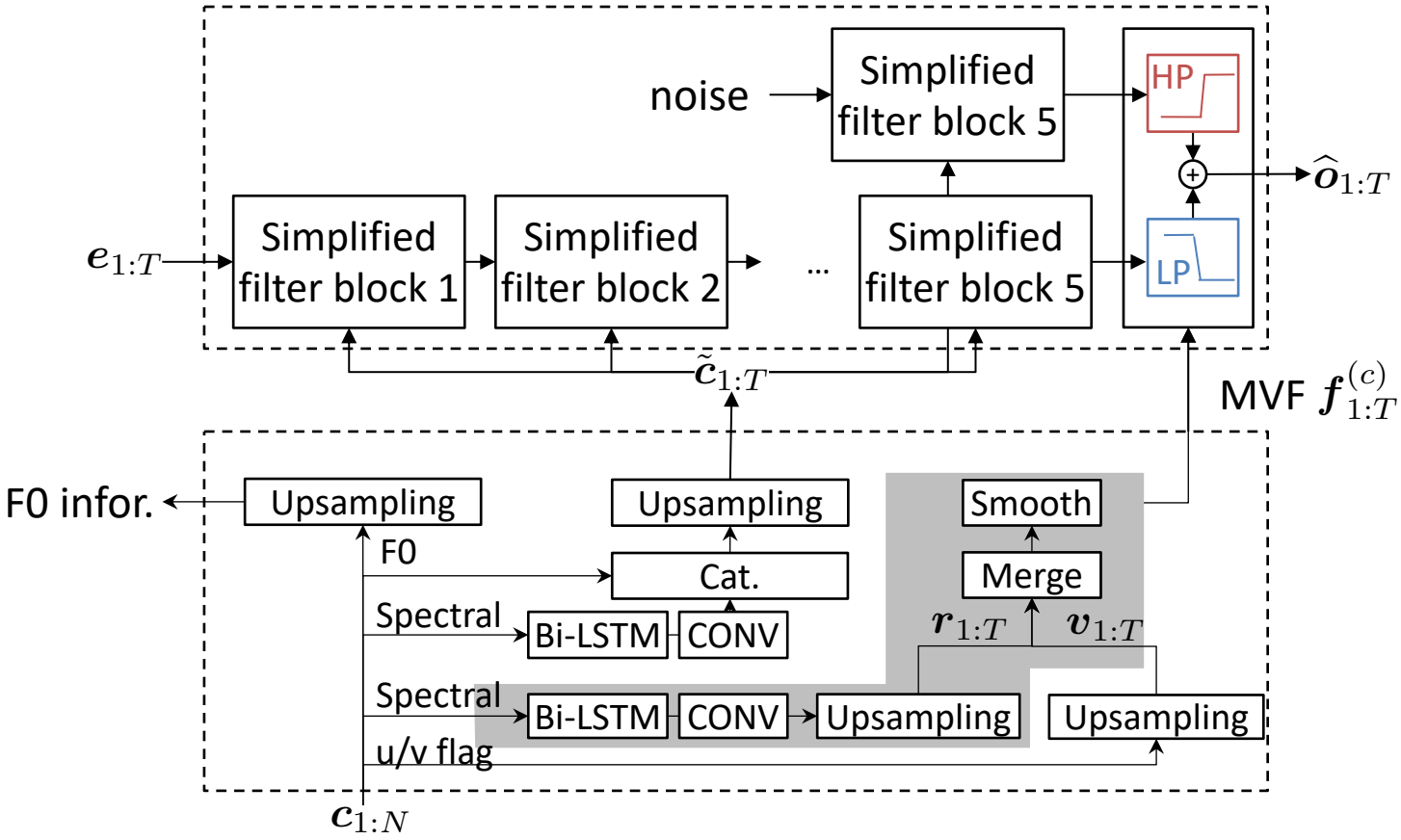


- How to predict MVF?

NEURAL SOURCE-FILTER MODEL

Harmonic-plus-noise NSF

- Version 2 (☛ ssw paper)



- Merge function: $f^{(c)}_{1:T} = \mathcal{F}(a v_{1:T} + b r_{1:T} + c)$
- Use unvoiced / voiced $v_{1:T}$ (u/v flag) as prior knowledge

CONTENTS

- Introduction
- Proposed model
- Experiments
- Summary

EXPERIMENTS

Configuration

❑ Data and features

Corpus	Size	Note
ATR Ximera F009 [1]	15 hours	16kHz, Japanese, neutral style

	Feature	Dimension
Acoustic	Mel-spectra	80
	F0	1

❑ Models

- WaveNet, hn-NSF with fixed (manually optimized) MVF
- Three hn-NSFs with trainable MVF
  1. u/v + predicted feature $f_t^{(c)} = v_t + 0.2 r_t$
  2. Predicted feature $f_t^{(c)} = 0.5 r_t + 0.5$
  3. Fully trainable $f_t^{(c)} = \mathrm{sigmoid}(a v_t + b r_t + c)$

$$v_t = \begin{cases} 0.7 & voiced \\ 0.3 & unvoiced \end{cases}$$

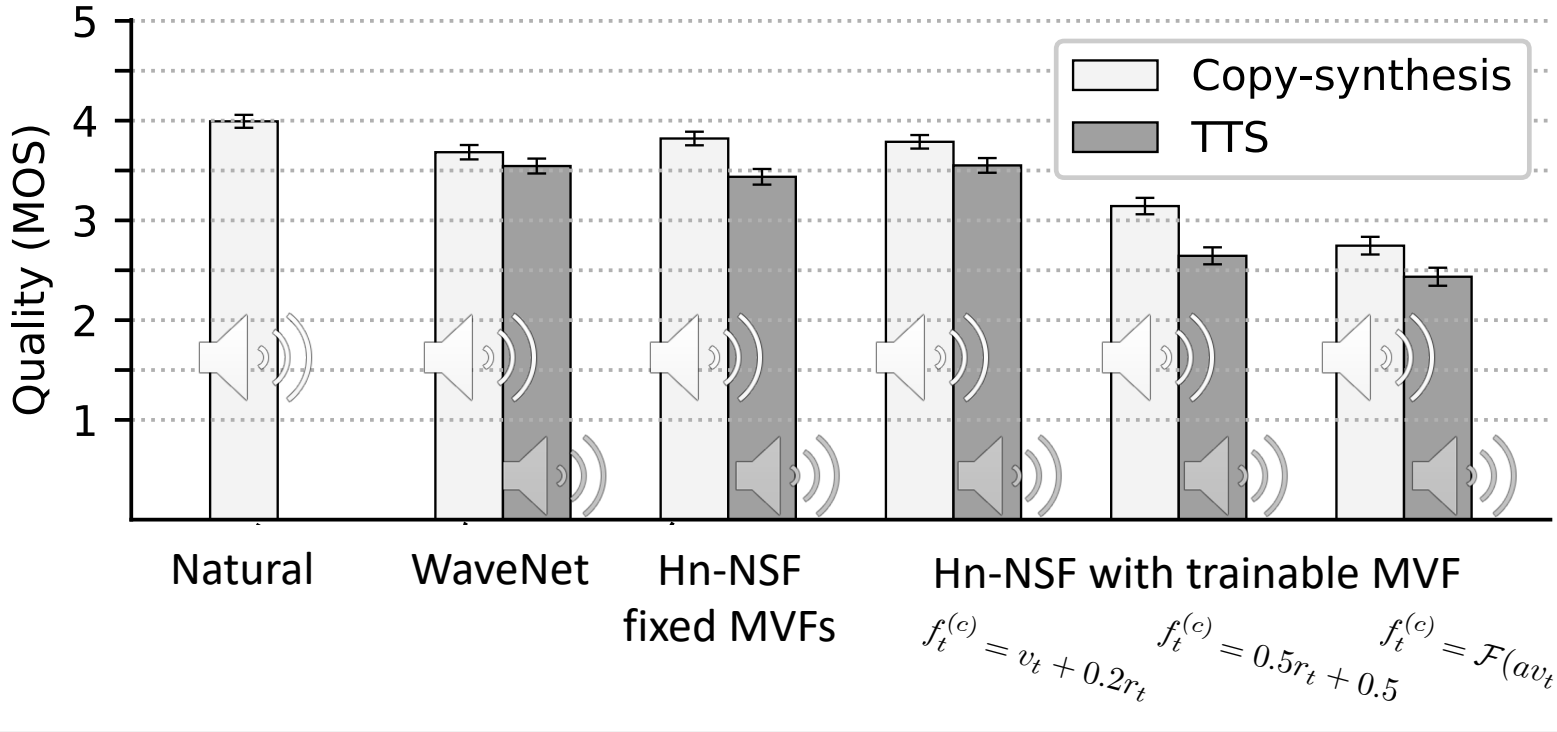
$$r_t \in (0, 1)$$

[1] Kawai, H., Toda, T., Ni, J., Tsuzaki, M., and Tokuda, K. (2004). Ximera: A new TTS from ATR based on corpus-based technologies. In Proc. SSW5, pages 179–184..

EXPERIMENTS

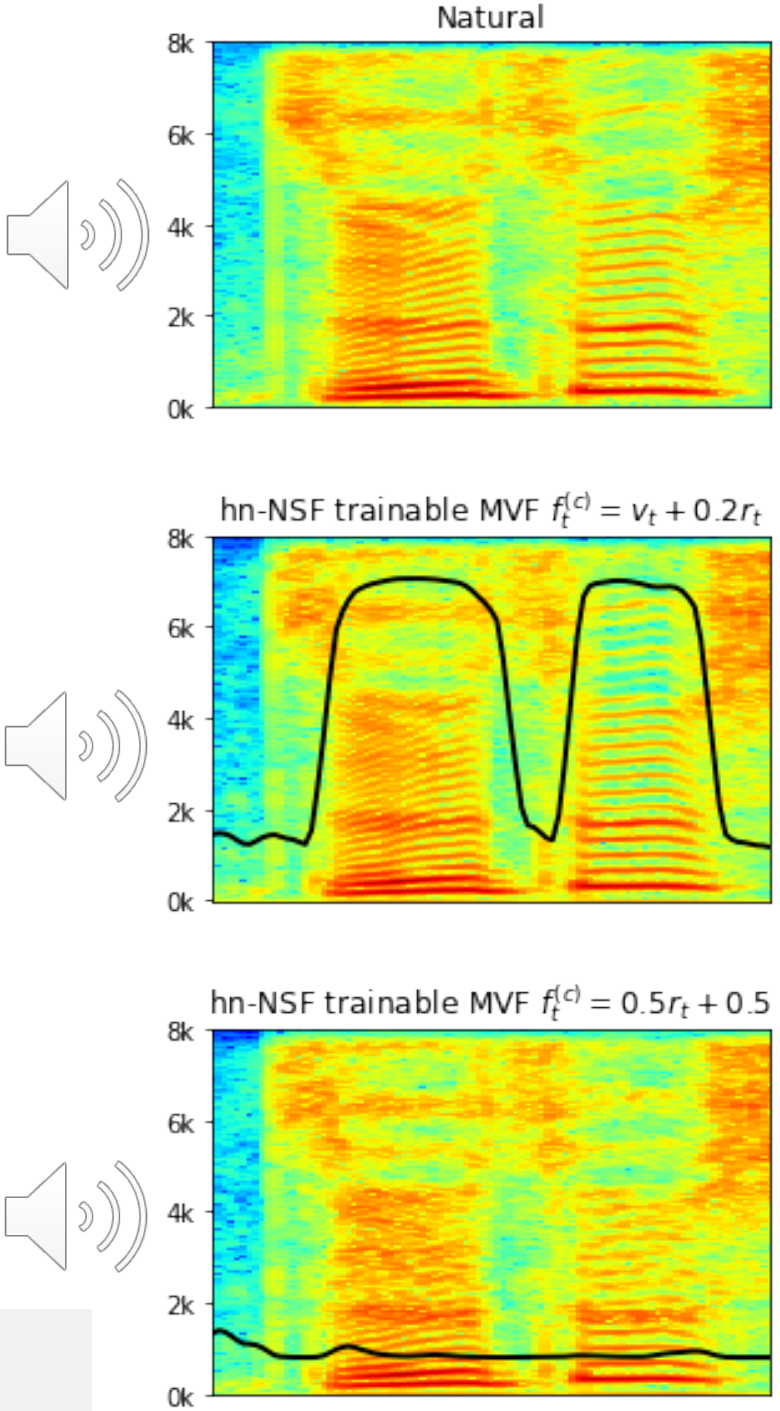
Results

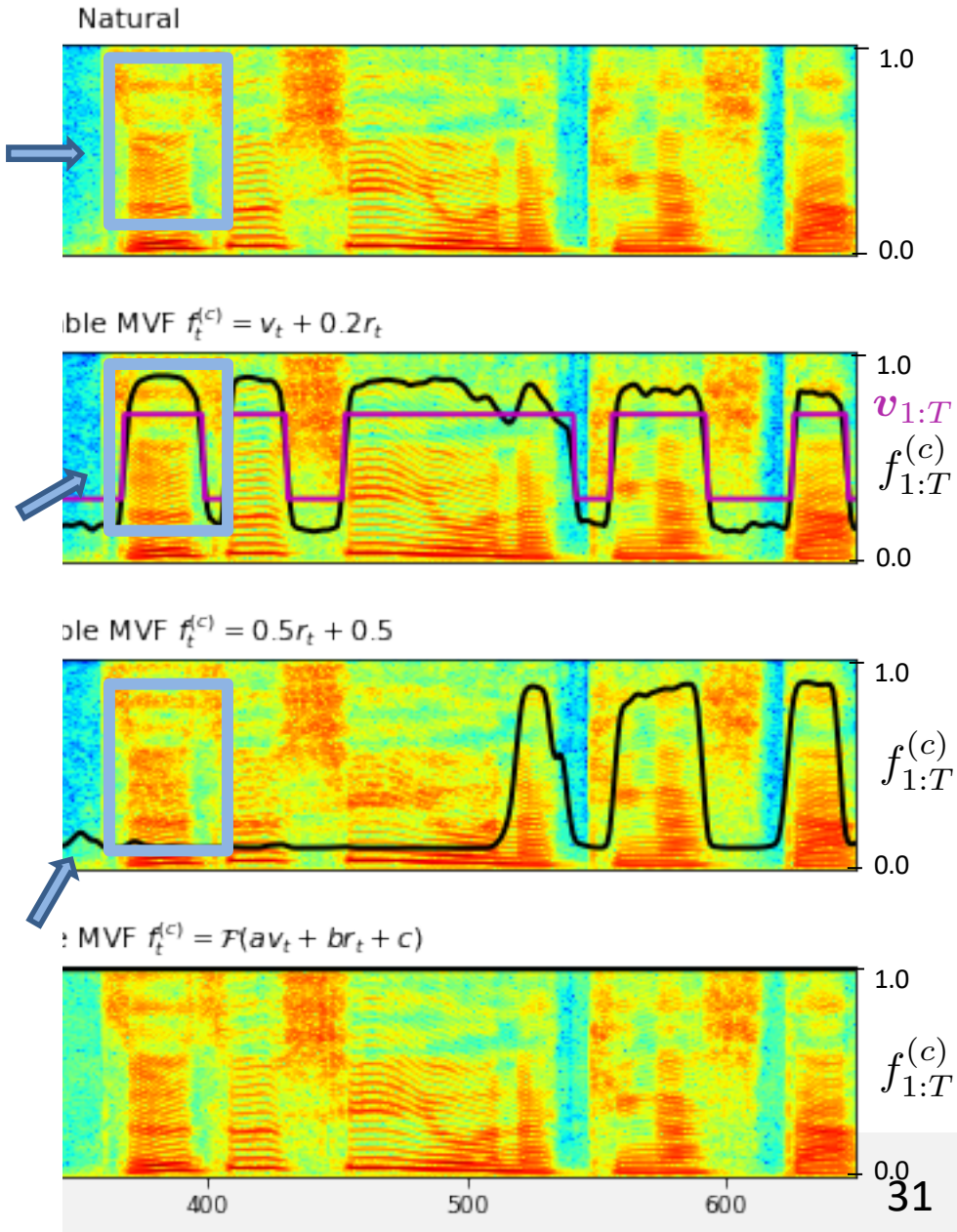
- Speech quality
  - ~150 paid evaluators, 1604 evaluation sets
    - **Copy-synthesis**: given natural Mel-spec/F0
    - **TTS**: given generated Mel-spec/F0 from acoustic models

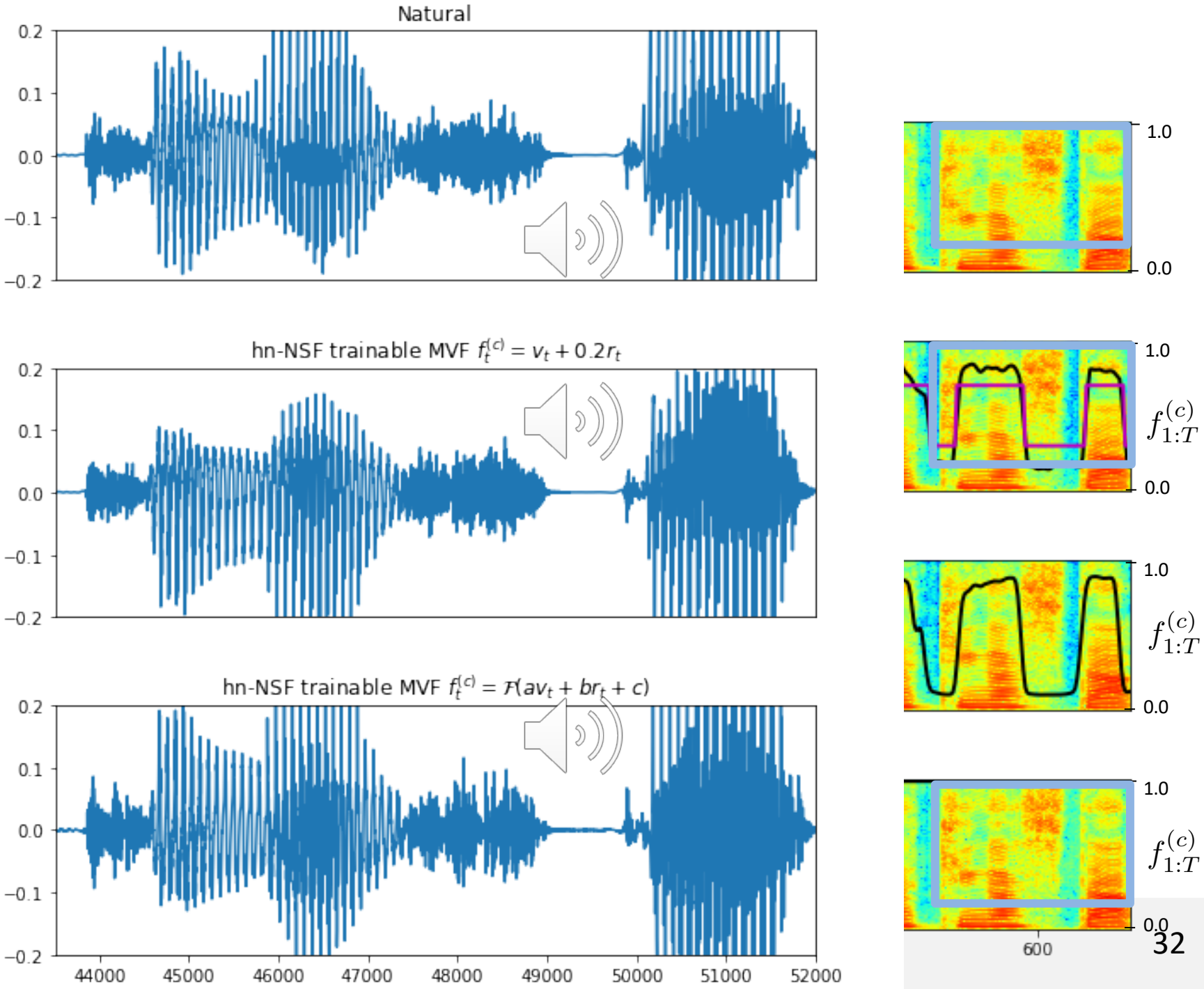


☛ Samples, models, codes: https://nii-yamagishilab.github.io/samples-nsf/nsf-v3.html

RIMENTS


Natural
8k
6k
4k
2k
0k
hn-NSF trainable MVF $f_t^{(c)} = v_t + 0.2r_t$
hn-NSF trainable MVF $f_t^{(c)} = 0.5r_t + 0.5$


Natural
1.0
0.0
ble MVF $f_t^{(c)} = v_t + 0.2r_t$
$v_{1:T}$
$f_{1:T}^{(c)}$
ble MVF $f_t^{(c)} = 0.5r_t + 0.5$
MVF $f_t^{(c)} = \mathcal{F}(av_t + br_t + c)$
400
500
600

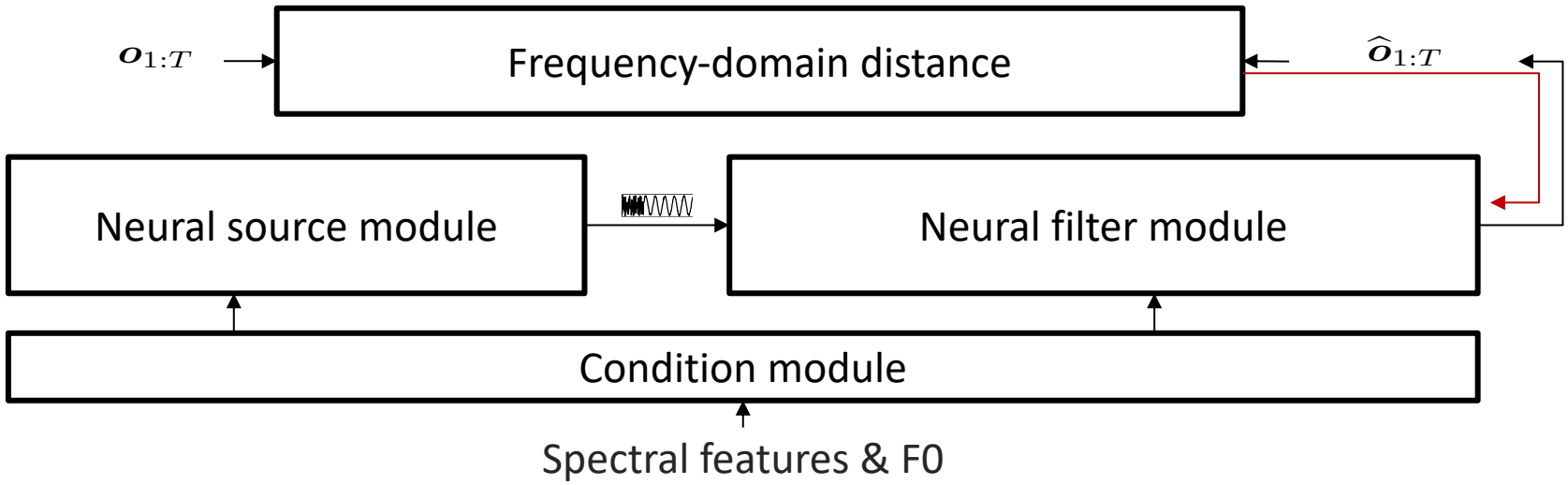

Natural
hn-NSF trainable MVF $f_t^{(c)} = v_t + 0.2r_t$
hn-NSF trainable MVF $f_t^{(c)} = \mathcal{F}(av_t + br_t + c)$
0.2
0.1
0.0
−0.1
−0.2
44000
45000
46000
47000
48000
49000
50000
51000
52000
1.0
0.0
$f_{1:T}^{(c)}$
600

CONTENTS

- Introduction
- Proposed model
- Experiments
- Summary

SUMMARY

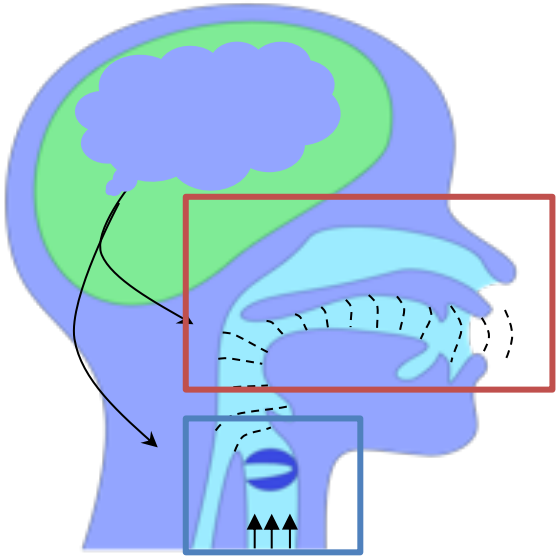
NSF framework



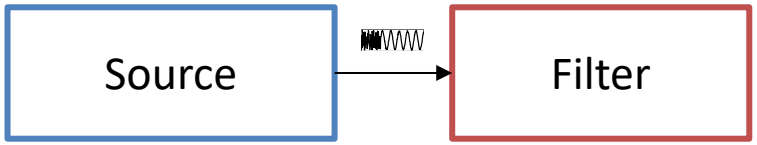
- No AR nor inverse AR flow
- Easy training & fast generation (☛ appendix)
- hn-NSF is recommended

SUMMARY

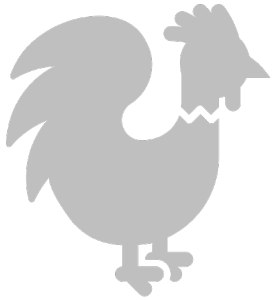
Future direction



(c.f. HTS Slides, by HTS Working Group)



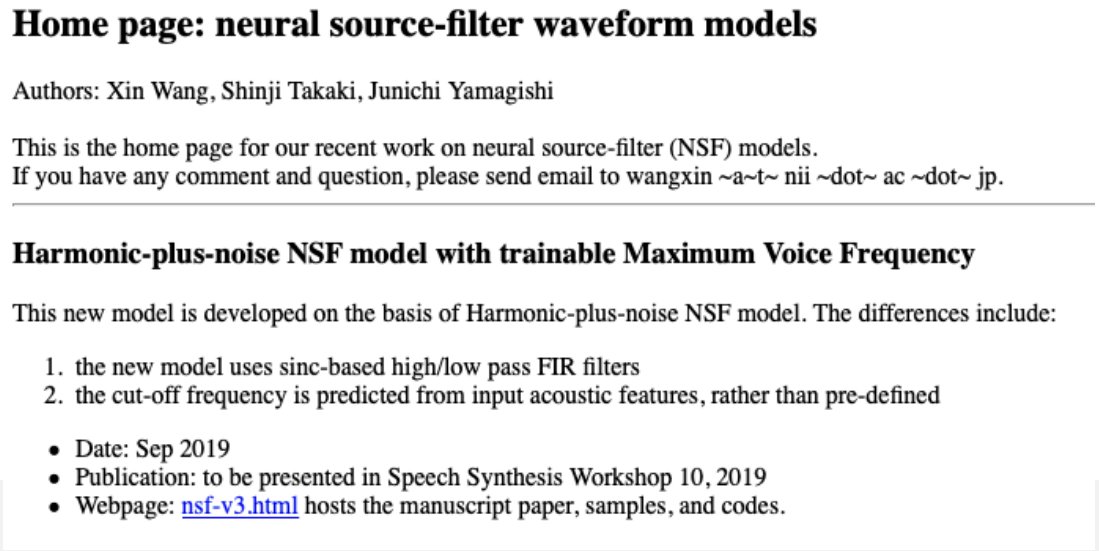
Natural		
Copy-synthesis		



❖ HTS Slides version 2.3, HTS Working Group : http://hts.sp.nitech.ac.jp/?Download

Questions & Comments are always Welcome!

https://nii-yamagishilab.github.io/samples-nsf/index.html



Home page: neural source-filter waveform models

Authors: Xin Wang, Shinji Takaki, Junichi Yamagishi

This is the home page for our recent work on neural source-filter (NSF) models.
If you have any comment and question, please send email to wangxin ~a~t~ nii ~dot~ ac ~dot~ jp.

Harmonic-plus-noise NSF model with trainable Maximum Voice Frequency

This new model is developed on the basis of Harmonic-plus-noise NSF model. The differences include:

1. the new model uses sinc-based high/low pass FIR filters
2. the cut-off frequency is predicted from input acoustic features, rather than pre-defined

- Date: Sep 2019
- Publication: to be presented in Speech Synthesis Workshop 10, 2019
- Webpage: nsf-v3.html hosts the manuscript paper, samples, and codes.

REFERENCE

WaveNet: A. van den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, and K. Kavukcuoglu. WaveNet: A generative model for raw audio. arXiv preprint arXiv:1609.03499, 2016.

SampleRNN: S. Mehri, K. Kumar, I. Gulrajani, R. Kumar, S. Jain, J. Sotelo, A. Courville, and Y. Bengio. Samplernn: An unconditional end-to-end neural audio generation model. arXiv preprint arXiv:1612.07837, 2016.

WaveRNN: N. Kalchbrenner, E. Elsen, K. Simonyan, et.al. Efficient neural audio synthesis. In J. Dy and A. Krause, editors, Proc. ICML, volume 80 of Proceedings of Machine Learning Research, pages 2410–2419, 10–15 Jul 2018.

FFTNet: Z. Jin, A. Finkelstein, G. J. Mysore, and J. Lu. FFTNet: A real-time speaker-dependent neural vocoder. In Proc. ICASSP, pages 2251–2255. IEEE, 2018.

Universal vocoder: J. Lorenzo-Trueba, T. Drugman, J. Latorre, T. Merritt, B. Putrycz, and R. Barra-Chicote. Robust universal neural vocoding. arXiv preprint arXiv:1811.06292, 2018.

Subband WaveNet: T. Okamoto, K. Tachibana, T. Toda, Y. Shiga, and H. Kawai. An investigation of subband wavenet vocoder covering entire audible frequency range with limited acoustic features. In Proc. ICASSP, pages 5654–5658. 2018.

Parallel WaveNet: A. van den Oord, Y. Li, I. Babuschkin, et. al.. Parallel WaveNet: Fast high-fidelity speech synthesis. In Proc. ICML, pages 3918–3926, 2018.

ClariNet: W. Ping, K. Peng, and J. Chen. Clarinet: Parallel wave generation in end-to-end text-to-speech. arXiv preprint arXiv:1807.07281, 2018.

FlowWaveNet: S. Kim, S.-g. Lee, J. Song, and S. Yoon. Flowavenet: A generative flow for raw audio. arXiv preprint arXiv:1811.02155, 2018.

WaveGlow: R. Prenger, R. Valle, and B. Catanzaro. Waveglow: A flow-based generative network for speech synthesis. arXiv preprint arXiv:1811.00002, 2018.

RNN+STFT: S. Takaki, T. Nakashika, X. Wang, and J. Yamagishi. STFT spectral loss for training a neural speech waveform model. In Proc. ICASSP (submitted), 2018.

NSF: X. Wang, S. Takaki, and J. Yamagishi. Neural source-filter-based waveform model for statistical para- metric speech synthesis. arXiv preprint arXiv:1810.11946, 2018.

LP-WavNet: M.-J. Hwang, F. Soong, F. Xie, X. Wang, and H.-G. Kang. Lp-wavenet: Linear prediction-based wavenet speech synthesis. arXiv preprint arXiv:1811.11913, 2018.

GlotNet: L. Juvela, V. Tsiaras, B. Bollepalli, M. Airaksinen, J. Yamagishi, and P. Alku. Speaker-independent raw waveform model for glottal excitation. arXiv preprint arXiv:1804.09593, 2018.

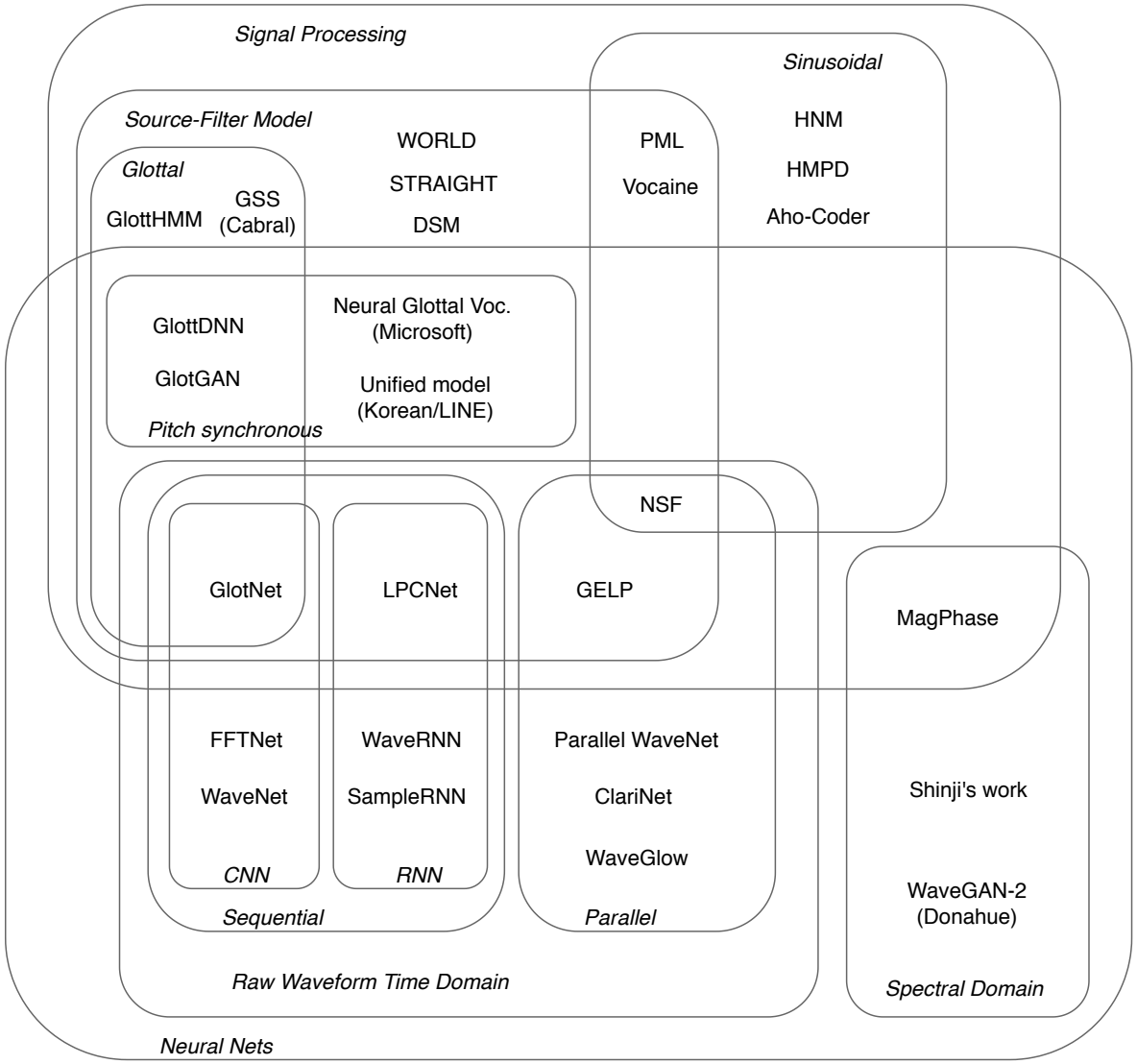
ExcitNet: E. Song, K. Byun, and H.-G. Kang. Excitnet vocoder: A neural excitation model for parametric speech synthesis systems. arXiv preprint arXiv:1811.04769, 2018.

LPCNet: J.-M. Valin and J. Skoglund. Lpcnet: Improving neural speech synthesis through linear prediction. arXiv preprint arXiv:1810.11846, 2018.

MCNN: S. O¨. Arık, H. Jun, and G. Diamos. Fast spectrogram inversion using multi-head convolutional neural networks. IEEE Signal Processing Letters, 26(1):94–98, 2018.

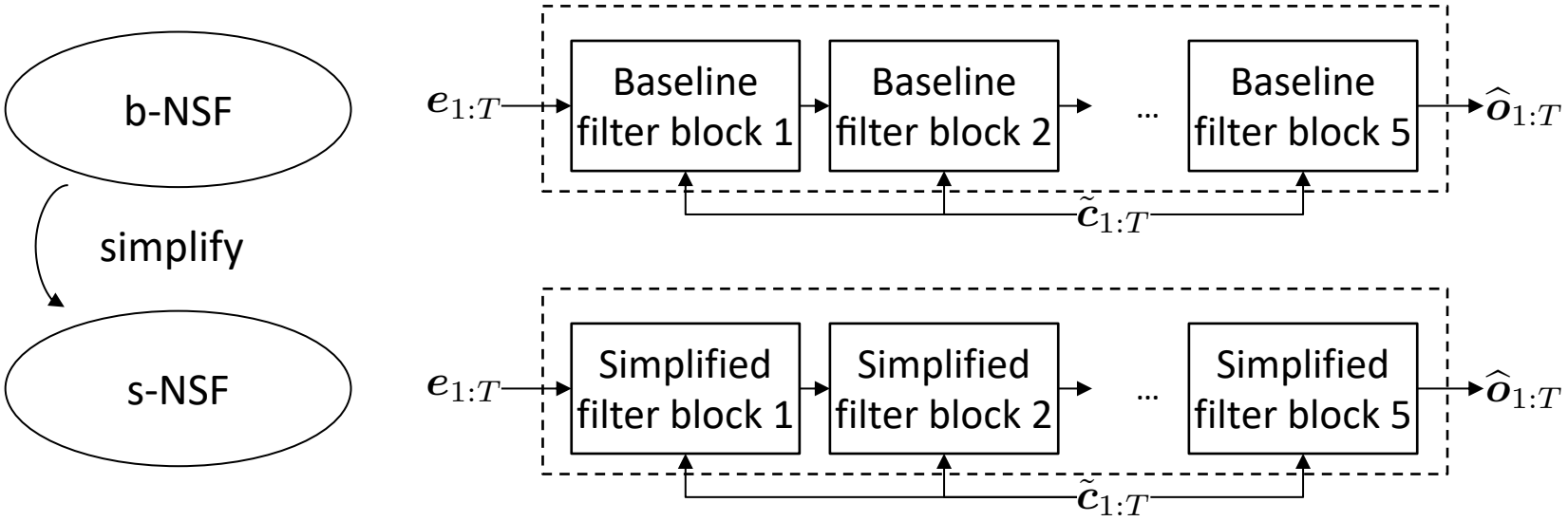
GELP: J. Lauri, et. al. GELP: GAN-Excited Linear Prediction for Speech Synthesis from Mel-spectrogram, Proc. Interspeech, 2019

REFERENCE


Signal Processing
Sinusoidal
Source-Filter Model
WORLD
STRAIGHT
DSM
PML
Vocaine
HNM
HMPD
Aho-Coder
Glottal
GSS
(Cabral)
GlottHMM
GlottDNN
GlotGAN
Neural Glottal Voc.
(Microsoft)
Unified model
(Korean/LINE)
Pitch synchronous
NSF
GlotNet
LPCNet
GELP
MagPhase
FFTNet
WaveNet
WaveRNN
SampleRNN
Parallel WaveNet
ClariNet
WaveGlow
Shinji's work
WaveGAN-2
(Donahue)
CNN
RNN
Sequential
Parallel
Raw Waveform Time Domain
Spectral Domain
Neural Nets

NEURAL SOURCE-FILTER MODEL

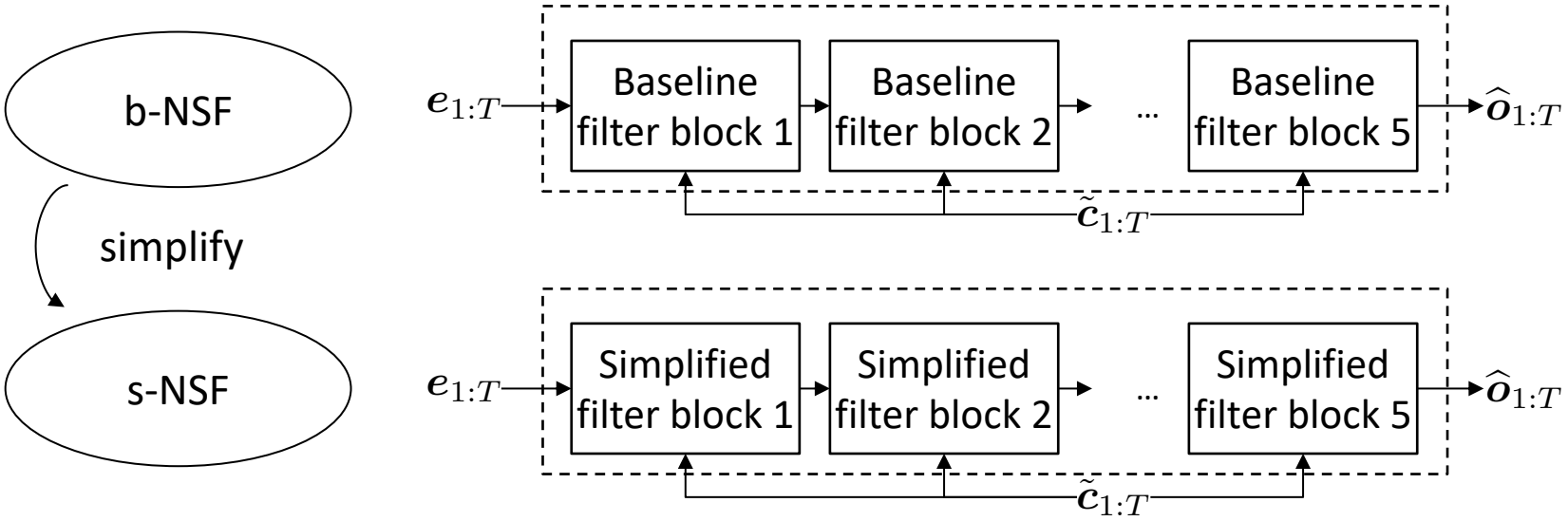
Baseline and simplified NSF

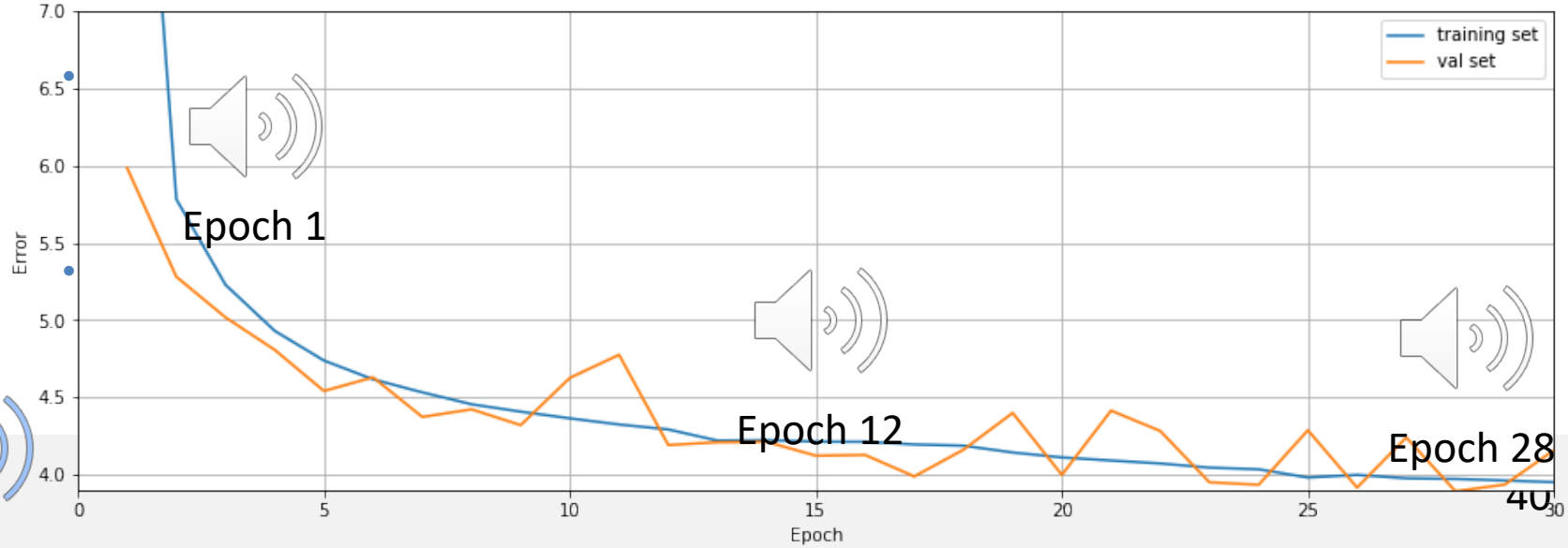


- Both models:
  1. Strong harmonics in high-frequency bands
  2. Awful unvoiced (fricative) sounds

NEURAL SOURCE-FILTER MODEL

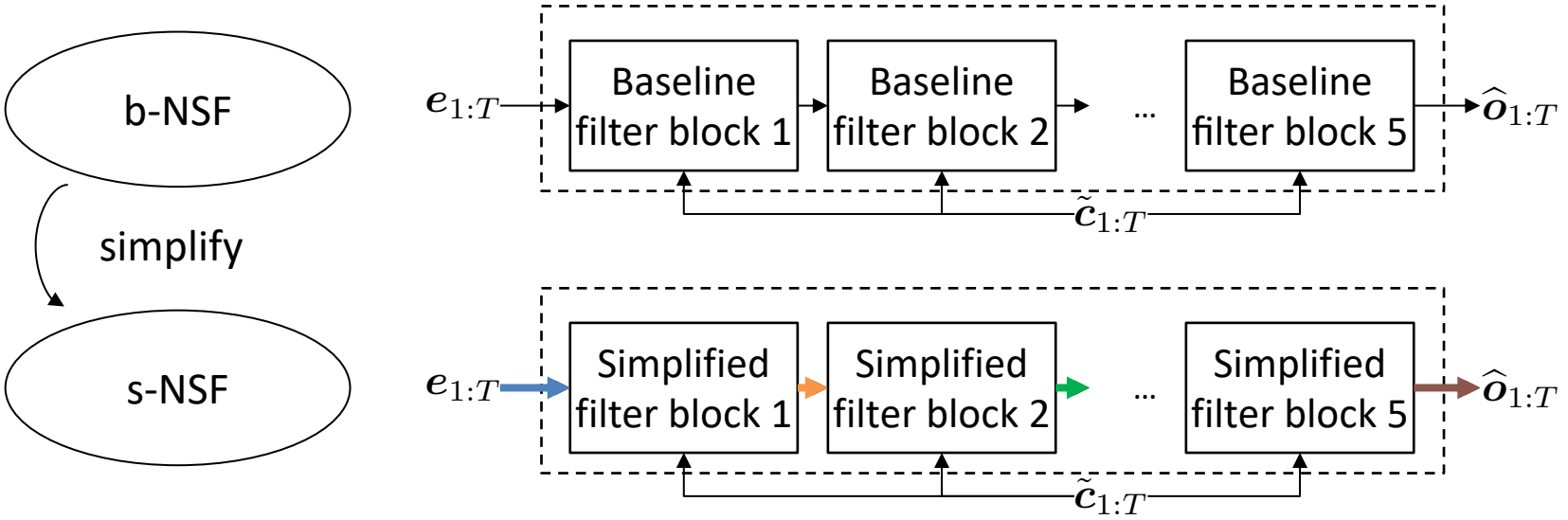
Baseline and simplified NSF

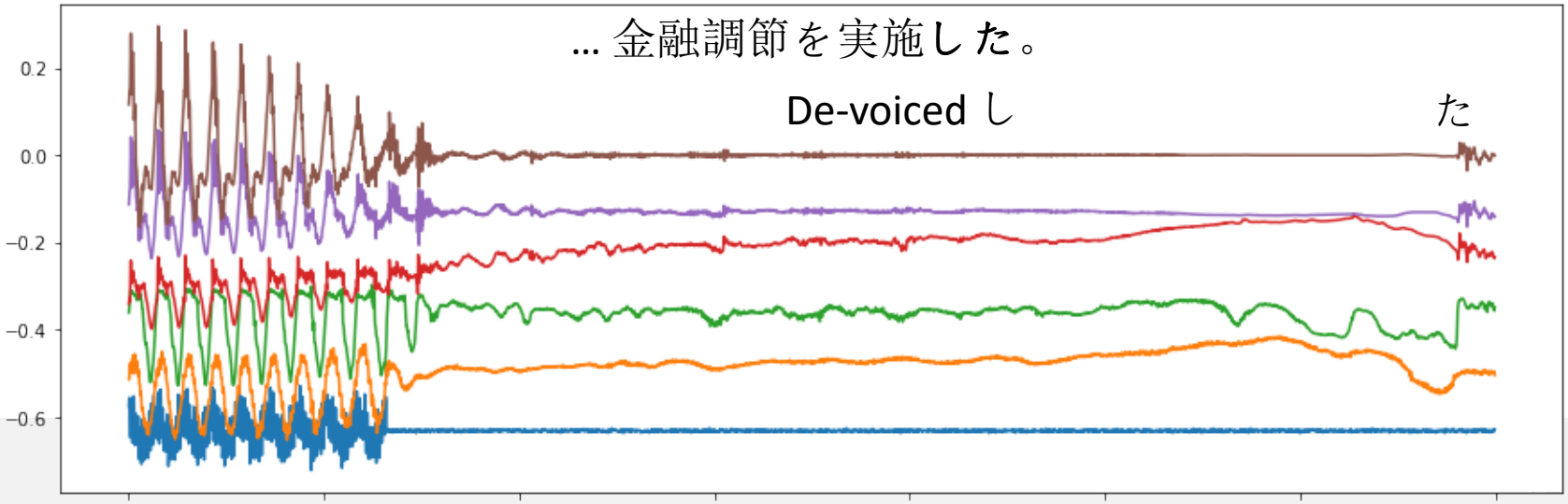




NEURAL SOURCE-FILTER MODEL

Baseline and simplified NSF

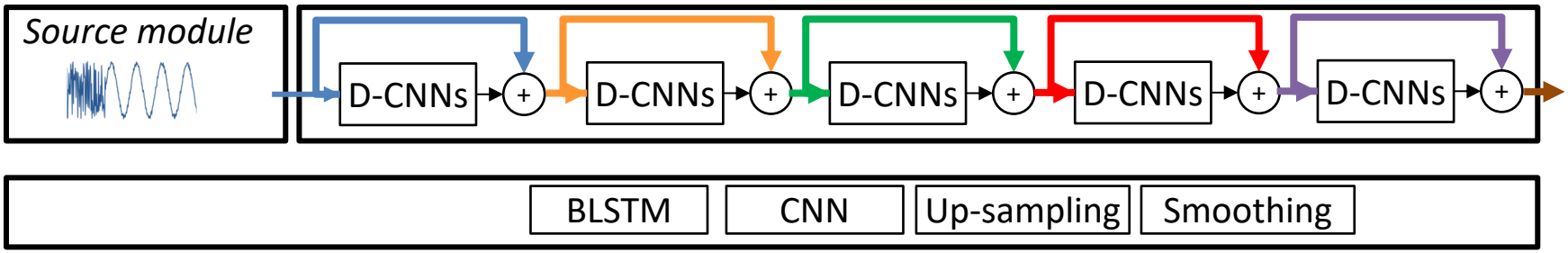


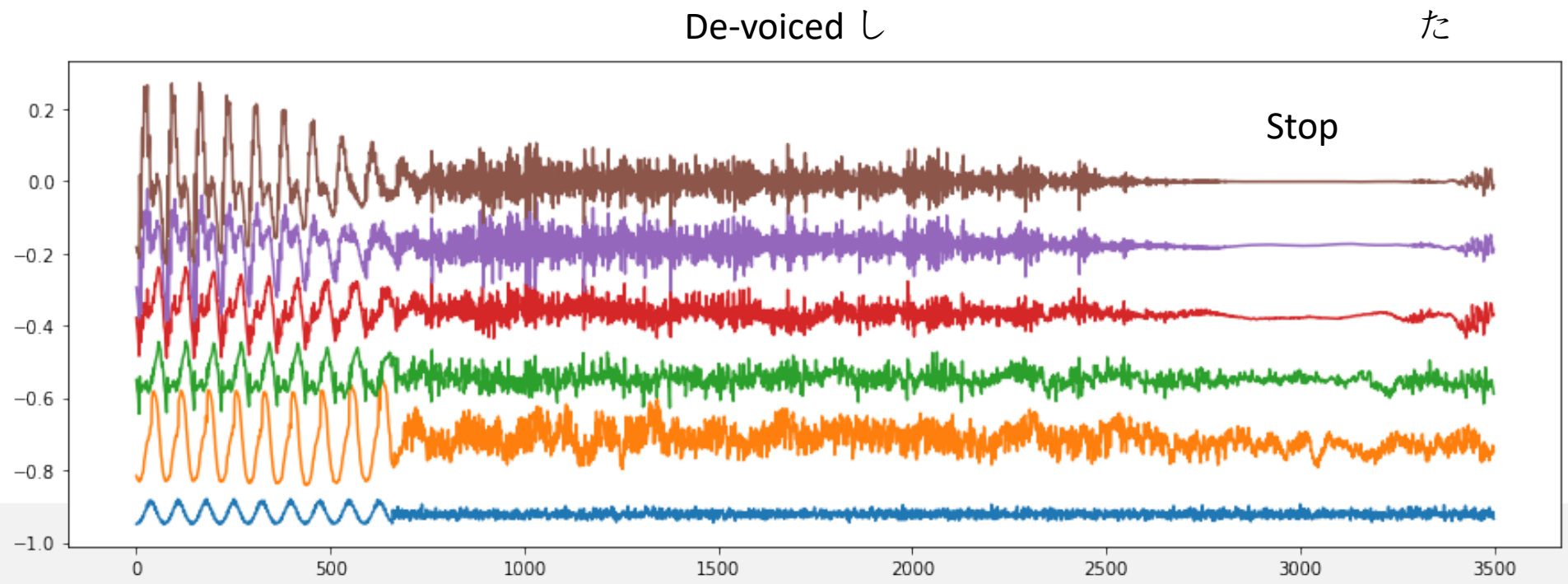


Waveform Modeling

Simplified NSF

- F009 15 hour's data

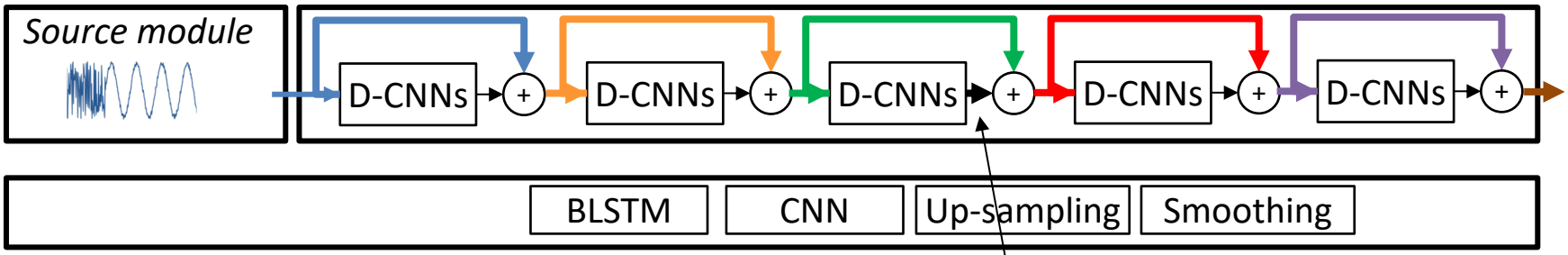


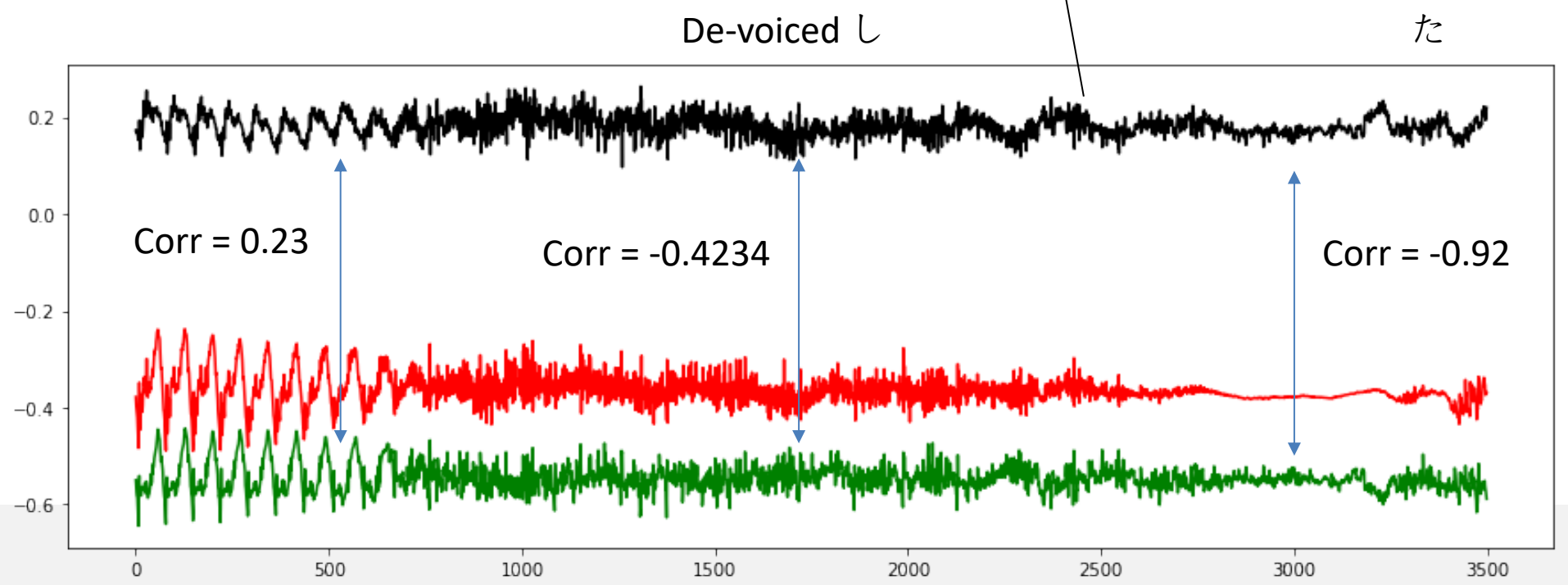


WAVEFORM MODELING

Simplified NSF

- F009 15 hour's data



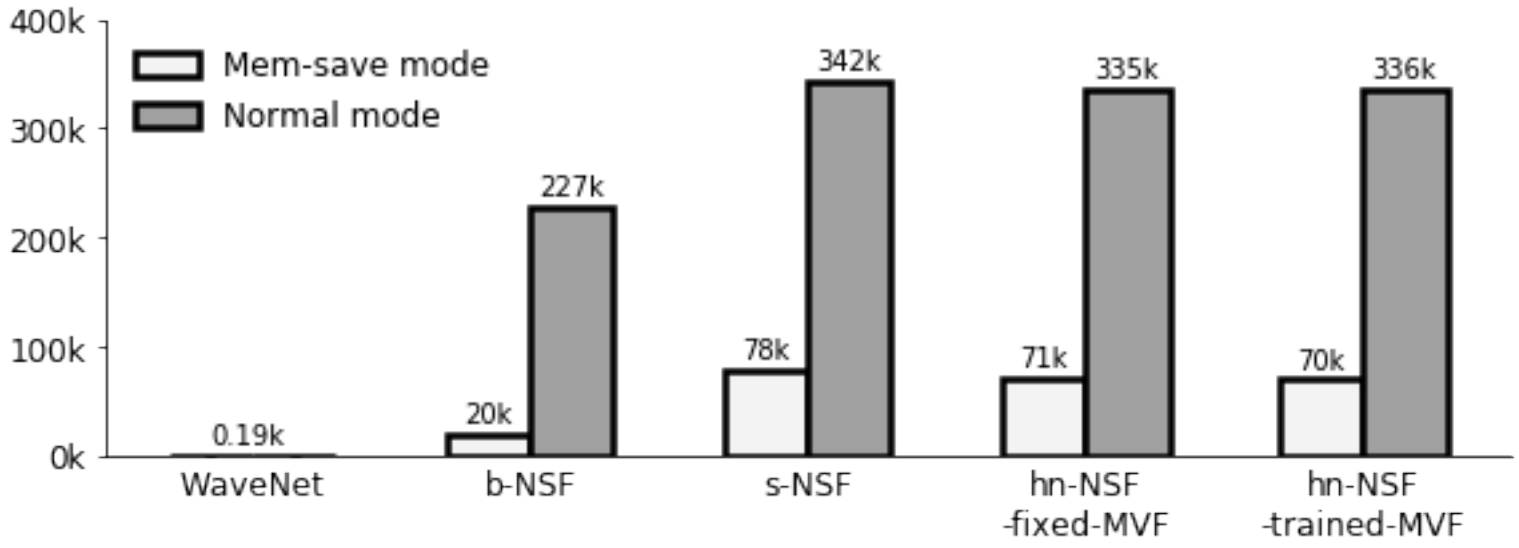


EXPERIMENTS

Analysis

- Generation speed

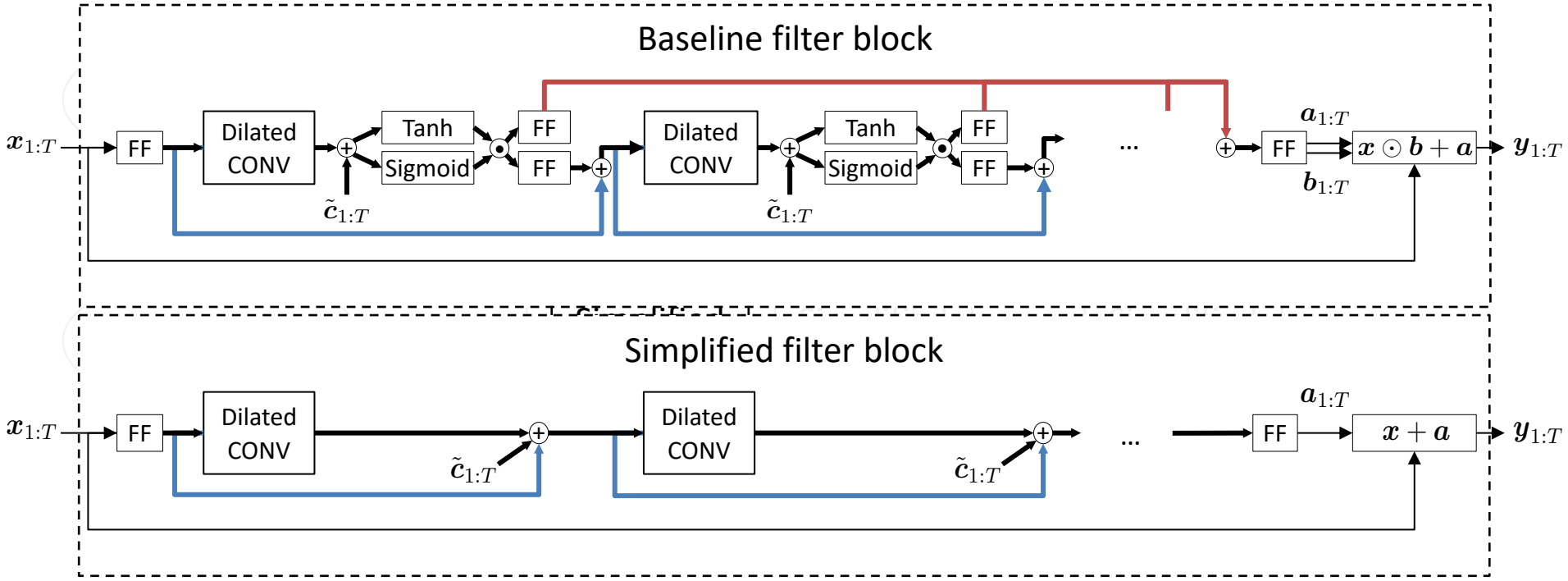
How many waveform points can be generated in 1s (Tesla p100)?



- Mem-save mode: release and allocate GPU memory layer by layer (limited by our CUDA implemetation)
- Normal mode: allocate GPU memory once

NEURAL SOURCE-FILTER MODEL

Filter modules in NSF models

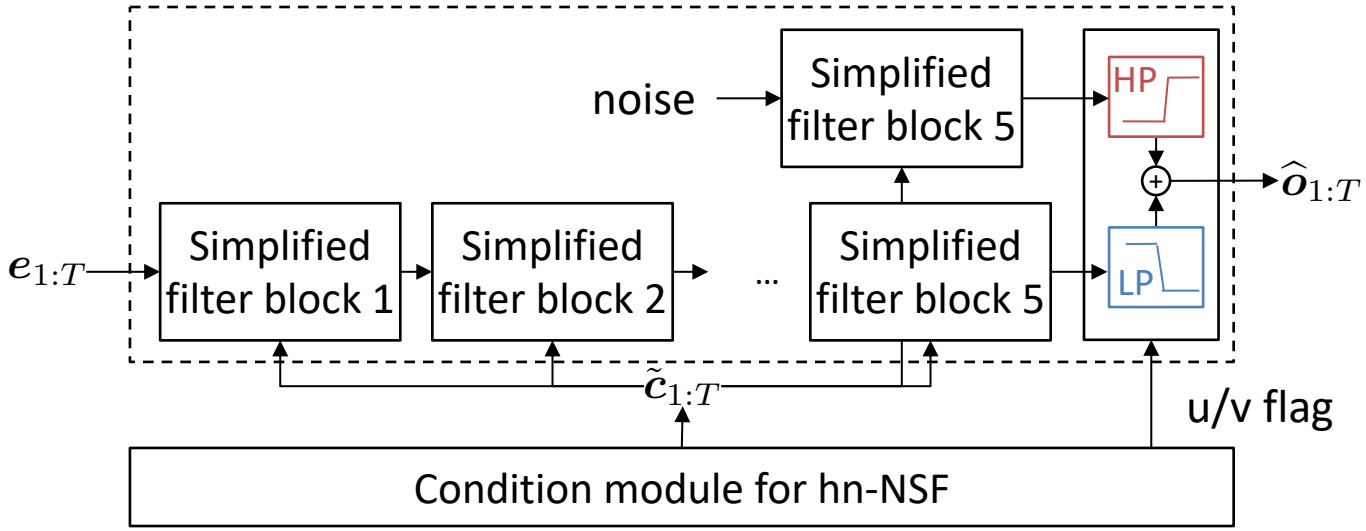


- $x_t, y_t, \widehat{o}_t, a_t \in \mathbb{R},\ b_t \in \mathbb{R}^+,\ \tilde{\boldsymbol{c}}_t \in \mathbb{R}^{64},\ \forall t \in \{1, \cdots, T\}$
- Element-wise multiplication $\odot$

NEURAL SOURCE-FILTER MODEL

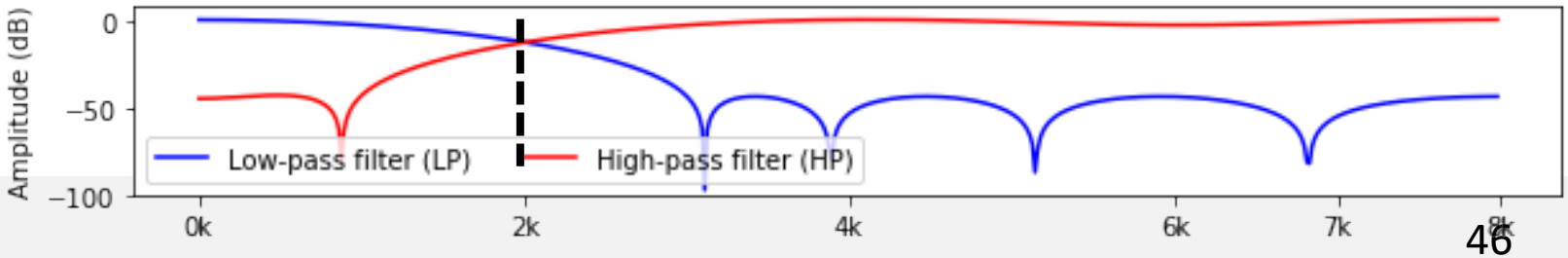
Harmonic-plus-noise NSF

- Version I: choose MVF based on u/v



Voiced sounds

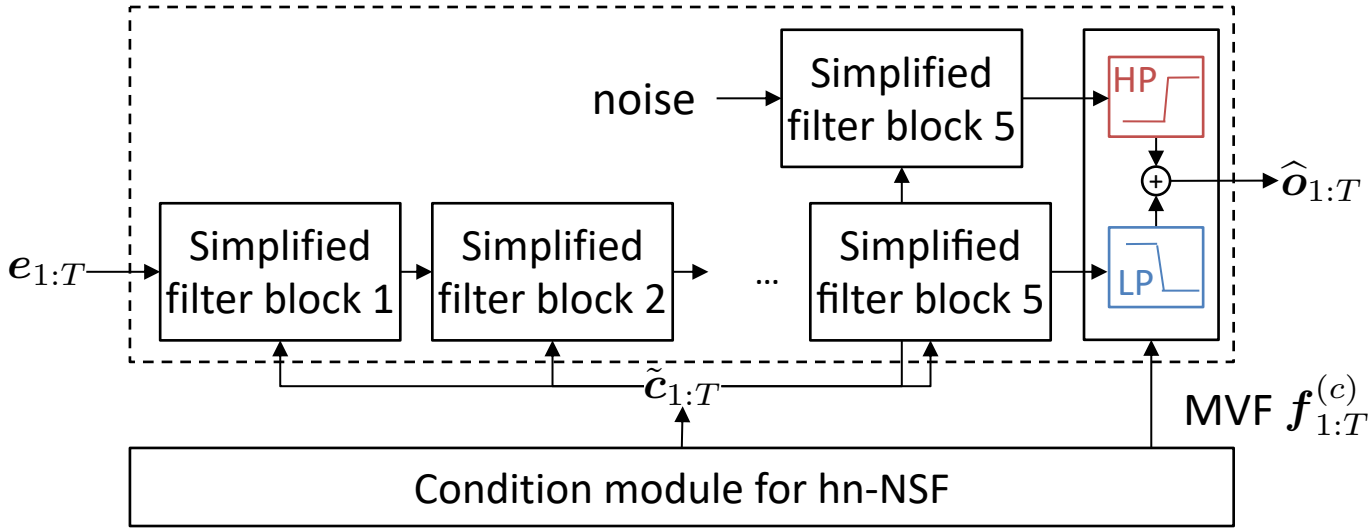
Unvoiced sounds



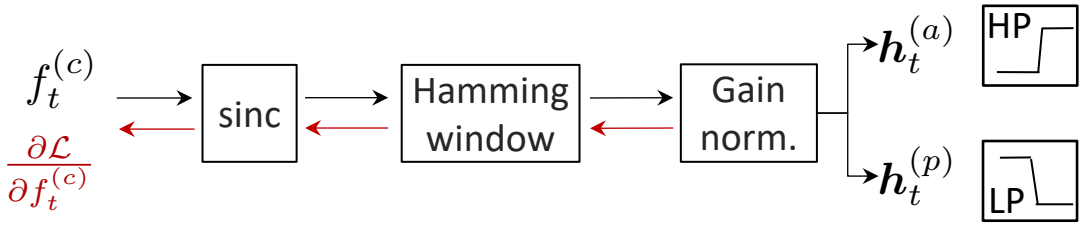
NEURAL SOURCE-FILTER MODEL

Harmonic-plus-noise NSF

- Version II: predict MVF from input features



- Forward and backward propagation (SSW paper section 3)



NEURAL SOURCE-FILTER MODEL

Harmonic-plus-noise NSF

- Time domain filtering

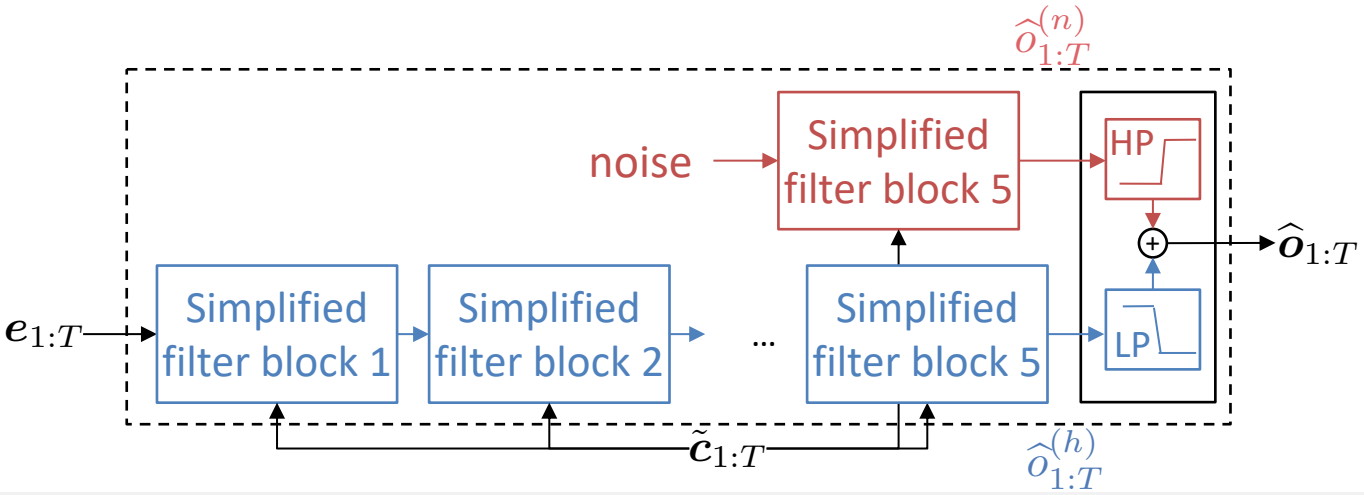
Noise component

Harmonic component

$$\hat{o}_t = a_{t,0}\hat{o}_t^{(n)} + a_{t,1}\hat{o}_{t-1}^{(n)} + \cdots + a_{t,M}\hat{o}_{t-M}^{(n)} + b_{t,0}\hat{o}_t^{(h)} + b_{t,1}\hat{o}_{t-1}^{(h)} + \cdots + b_{t,N}\hat{o}_{t-N}^{(h)}$$

High-pass filter coefficients

Low-pass filter coefficients



NEURAL SOURCE-FILTER MODEL

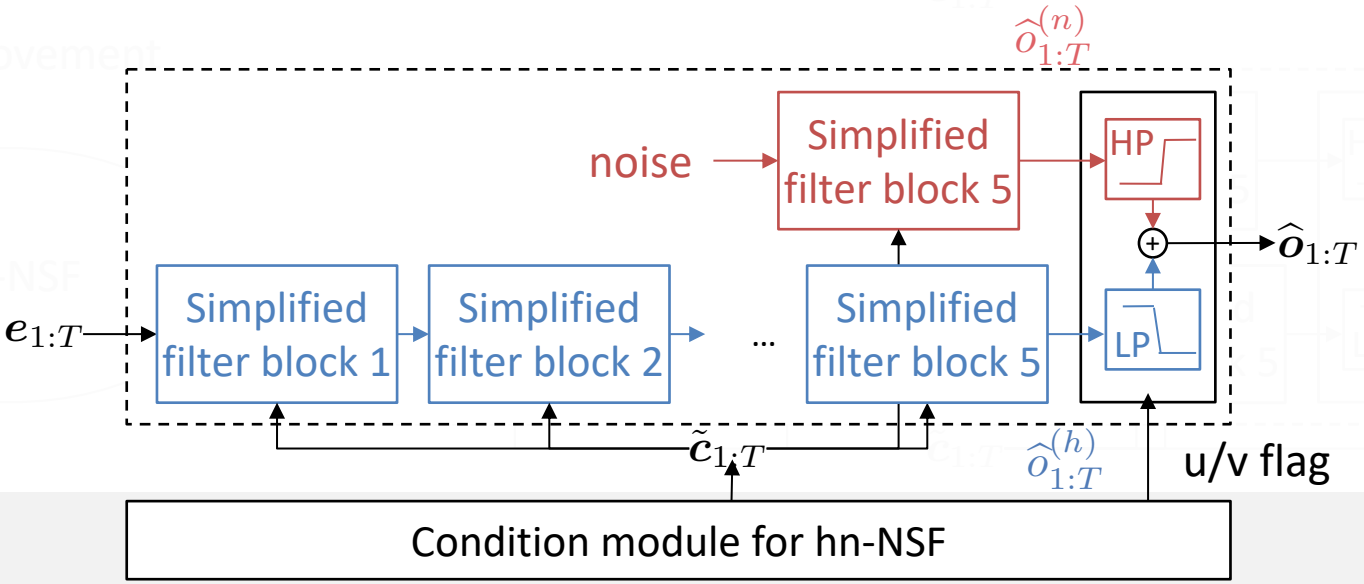
Harmonic-plus-noise NSF

- Version I: pre-defined filters coefficients
  - Select one pair of HP-LP filters based on u/v flag

Noise component

Harmonic component

$$\widehat{o}_t = \begin{cases} a_0\widehat{o}_t^{(n)} + a_1\widehat{o}_{t-1}^{(n)} + \cdots + a_M\widehat{o}_{t-M}^{(n)} + b_0\widehat{o}_t^{(h)} + b_1\widehat{o}_{t-1}^{(h)} + \cdots + b_N\widehat{o}_{t-N}^{(h)}, & t \text{ is voiced} \\ c_0\widehat{o}_t^{(n)} + c_1\widehat{o}_{t-1}^{(n)} + \cdots + c_M\widehat{o}_{t-M}^{(n)} + d_0\widehat{o}_t^{(h)} + d_1\widehat{o}_{t-1}^{(h)} + \cdots + d_N\widehat{o}_{t-N}^{(h)}, & t \text{ is unvoiced} \end{cases}$$



NEURAL SOURCE-FILTER MODEL

Harmonic-plus-noise NSF

- Version I: pre-defined filters coefficients
  - Select one pair of HP-LP filters based on u/v flag

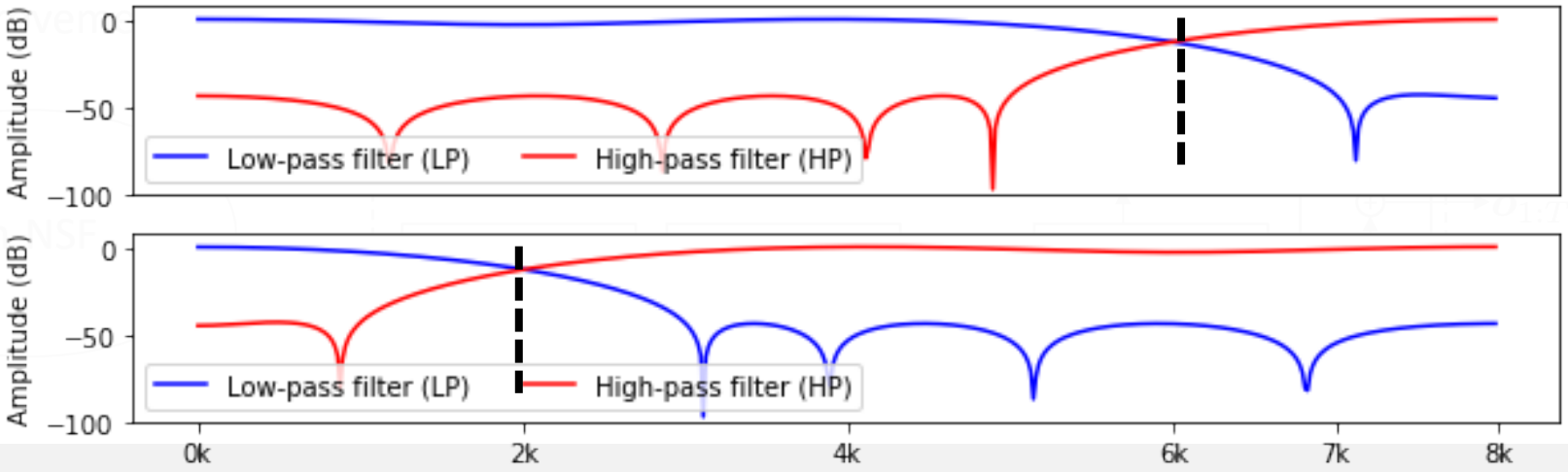
Noise component

Harmonic component

$$\widehat{o}_t = \begin{cases} a_0\widehat{o}_t^{(n)} + a_1\widehat{o}_{t-1}^{(n)} + \cdots + a_M\widehat{o}_{t-M}^{(n)} + b_0\widehat{o}_t^{(h)} + b_1\widehat{o}_{t-1}^{(h)} + \cdots + b_N\widehat{o}_{t-N}^{(h)}, & t \text{ is voiced} \\ c_0\widehat{o}_t^{(n)} + c_1\widehat{o}_{t-1}^{(n)} + \cdots + c_M\widehat{o}_{t-M}^{(n)} + d_0\widehat{o}_t^{(h)} + d_1\widehat{o}_{t-1}^{(h)} + \cdots + d_N\widehat{o}_{t-N}^{(h)}, & t \text{ is unvoiced} \end{cases}$$

Voiced sounds

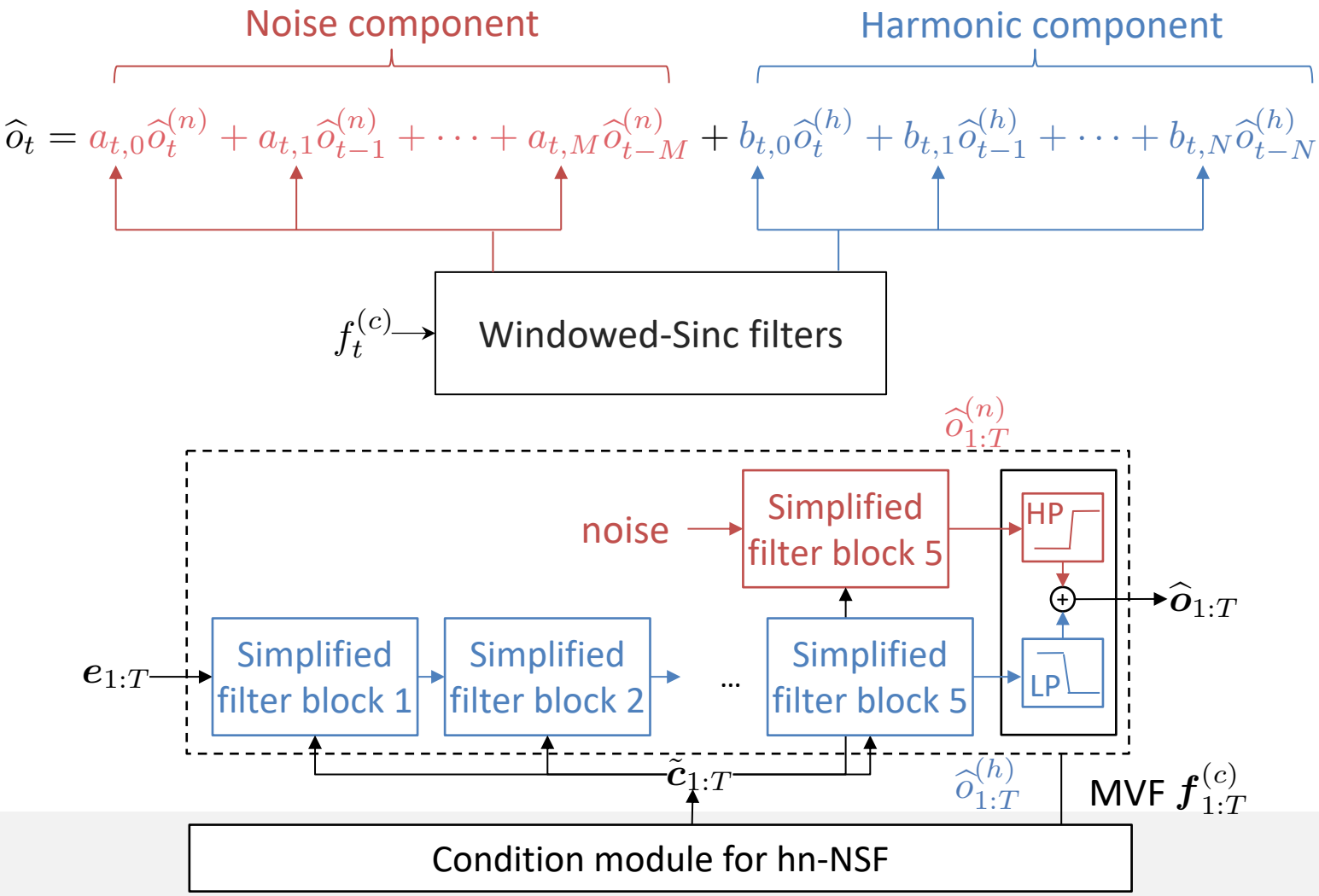
Unvoiced sounds



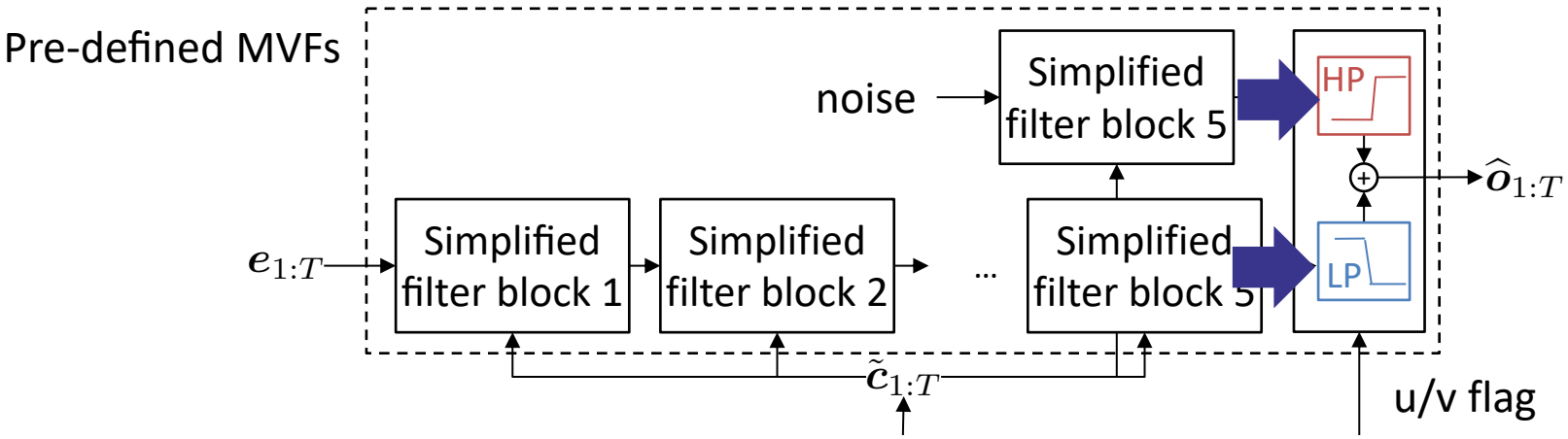
NEURAL SOURCE-FILTER MODEL

Harmonic-plus-noise NSF

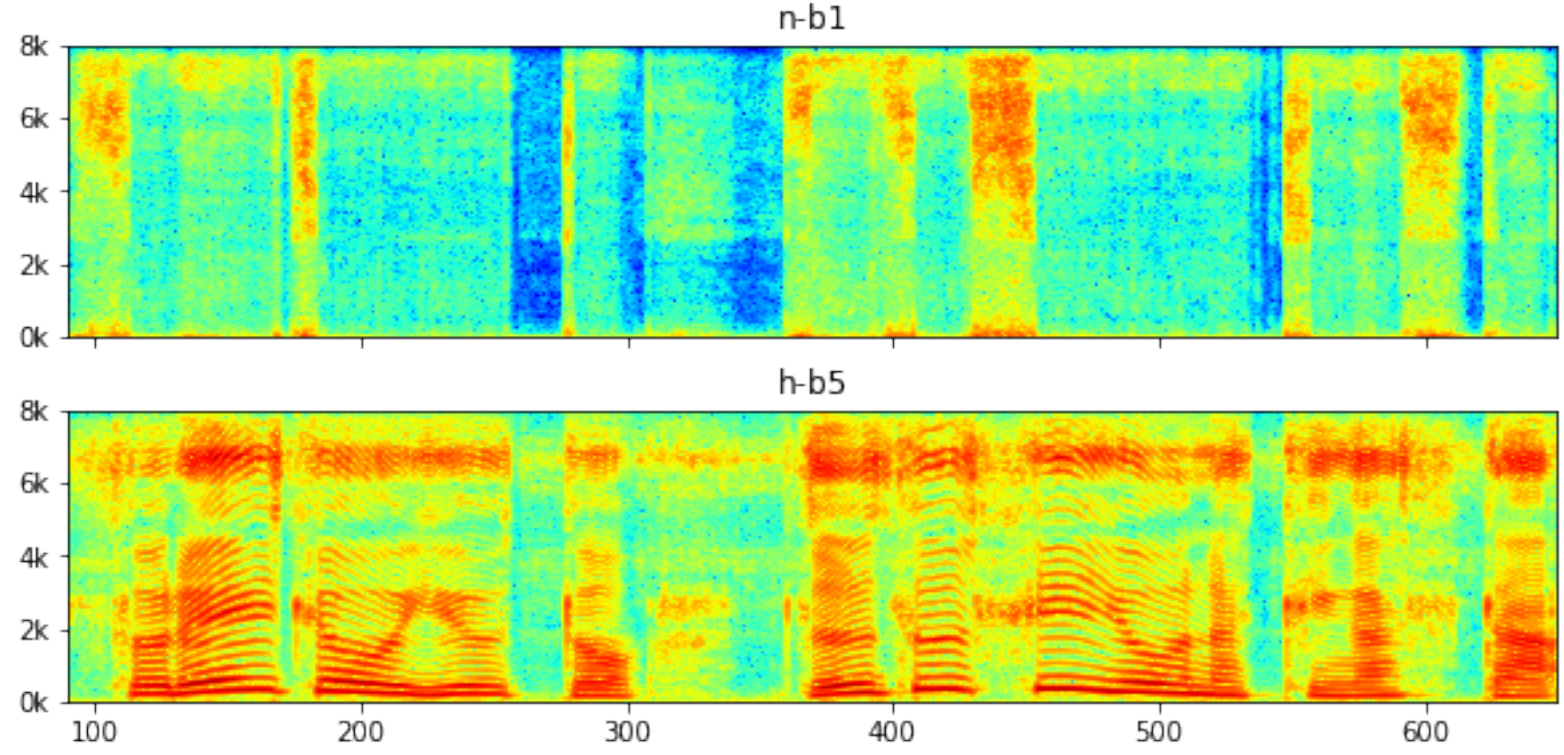
- Version II: predicted filter coefficients



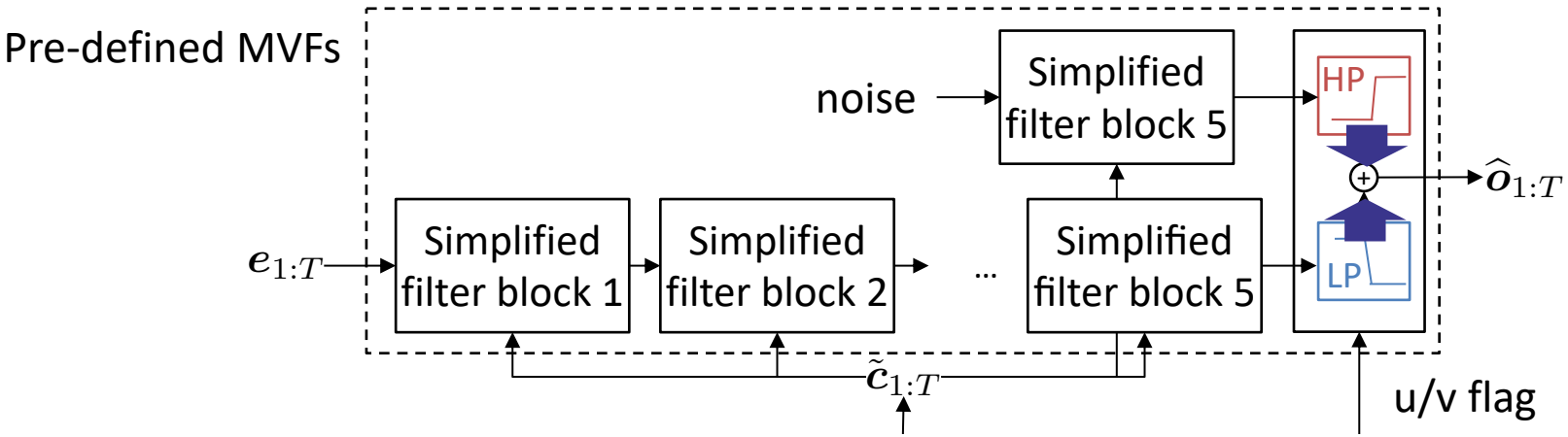
NEURAL SOURCE-FILTER MODEL



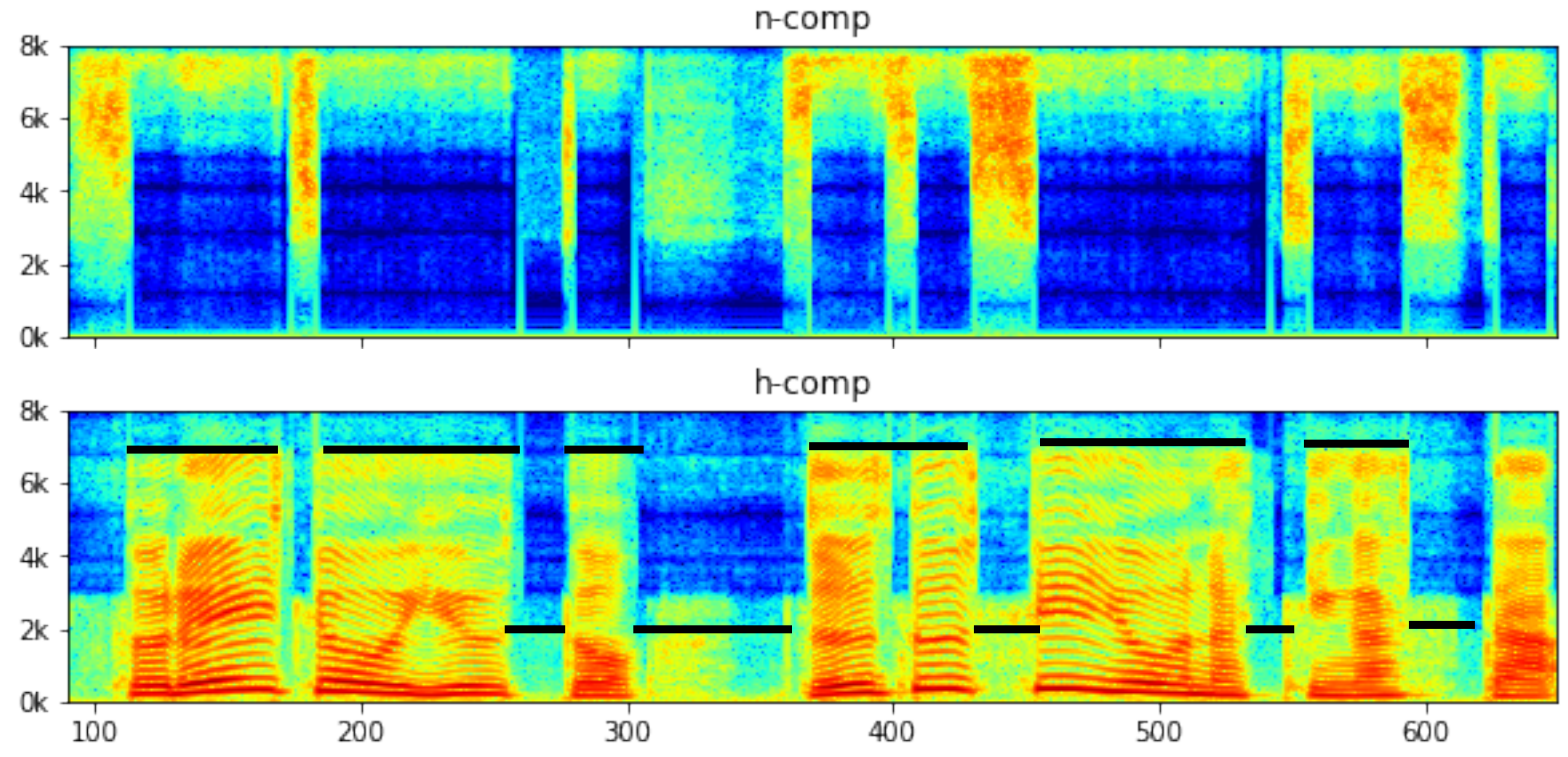
Predi

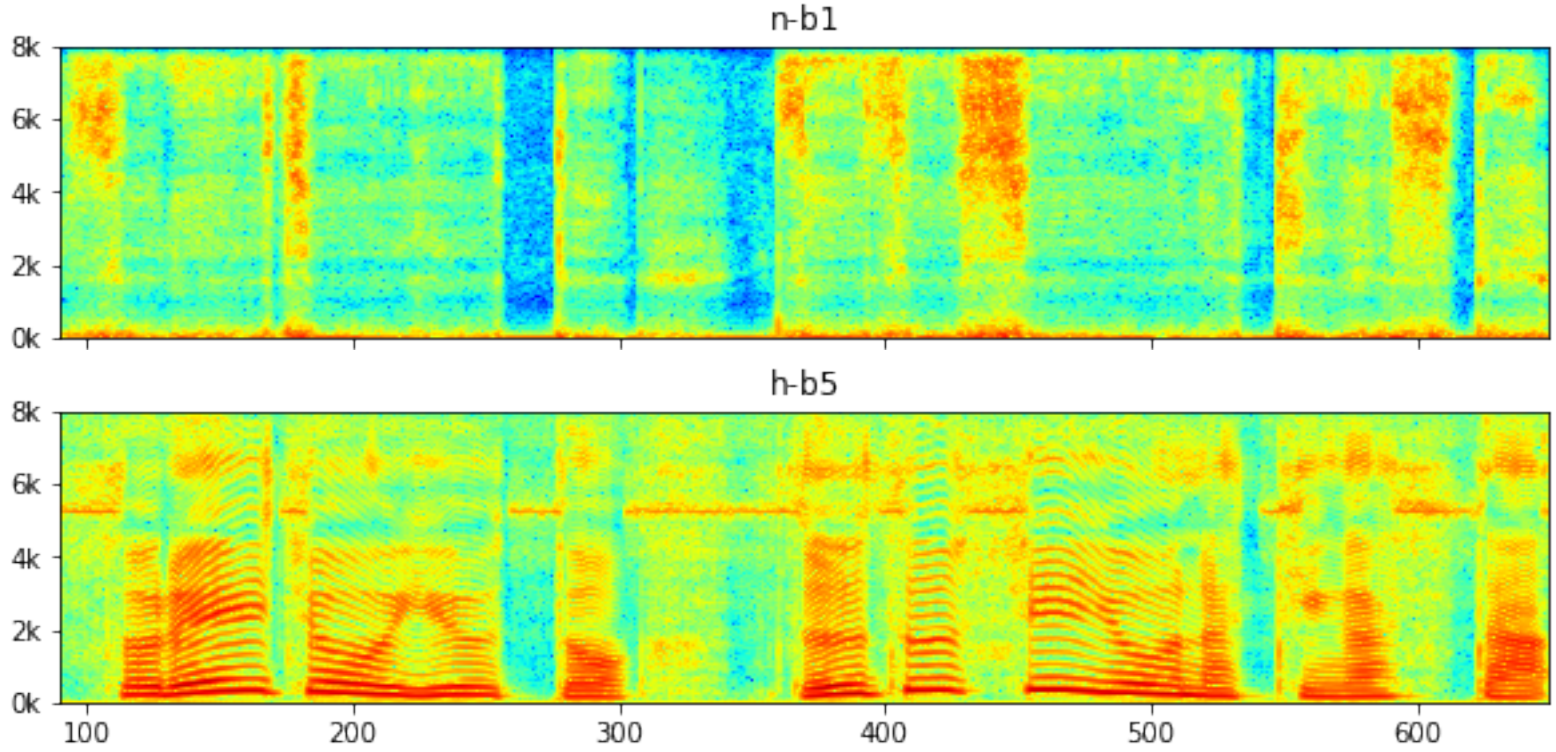


NEURAL SOURCE-FILTER MODEL



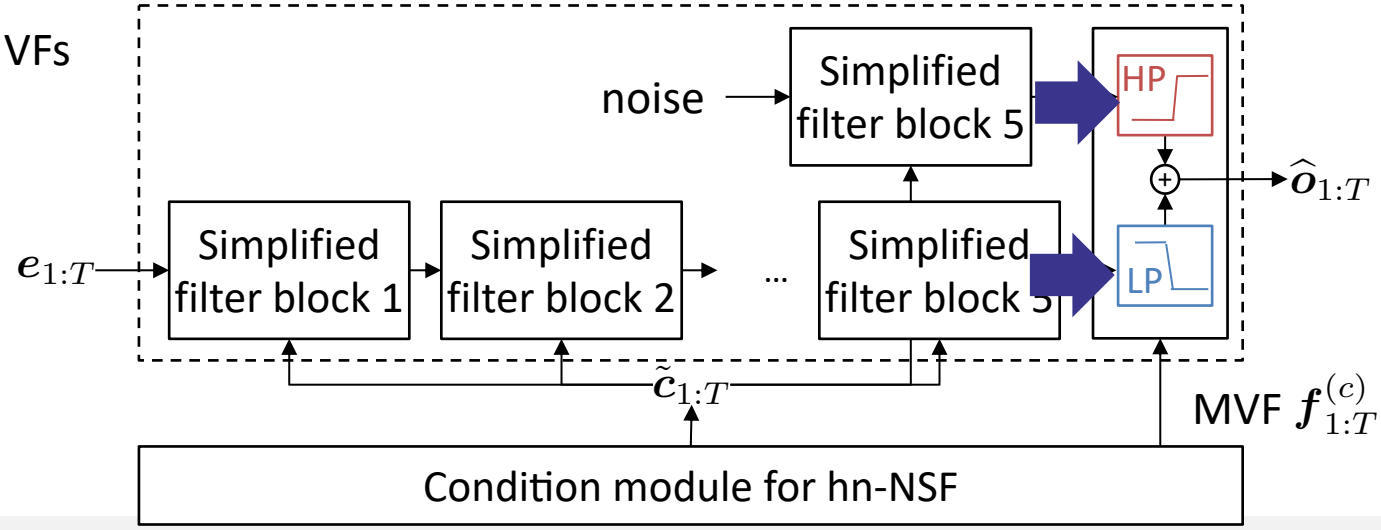
Predi



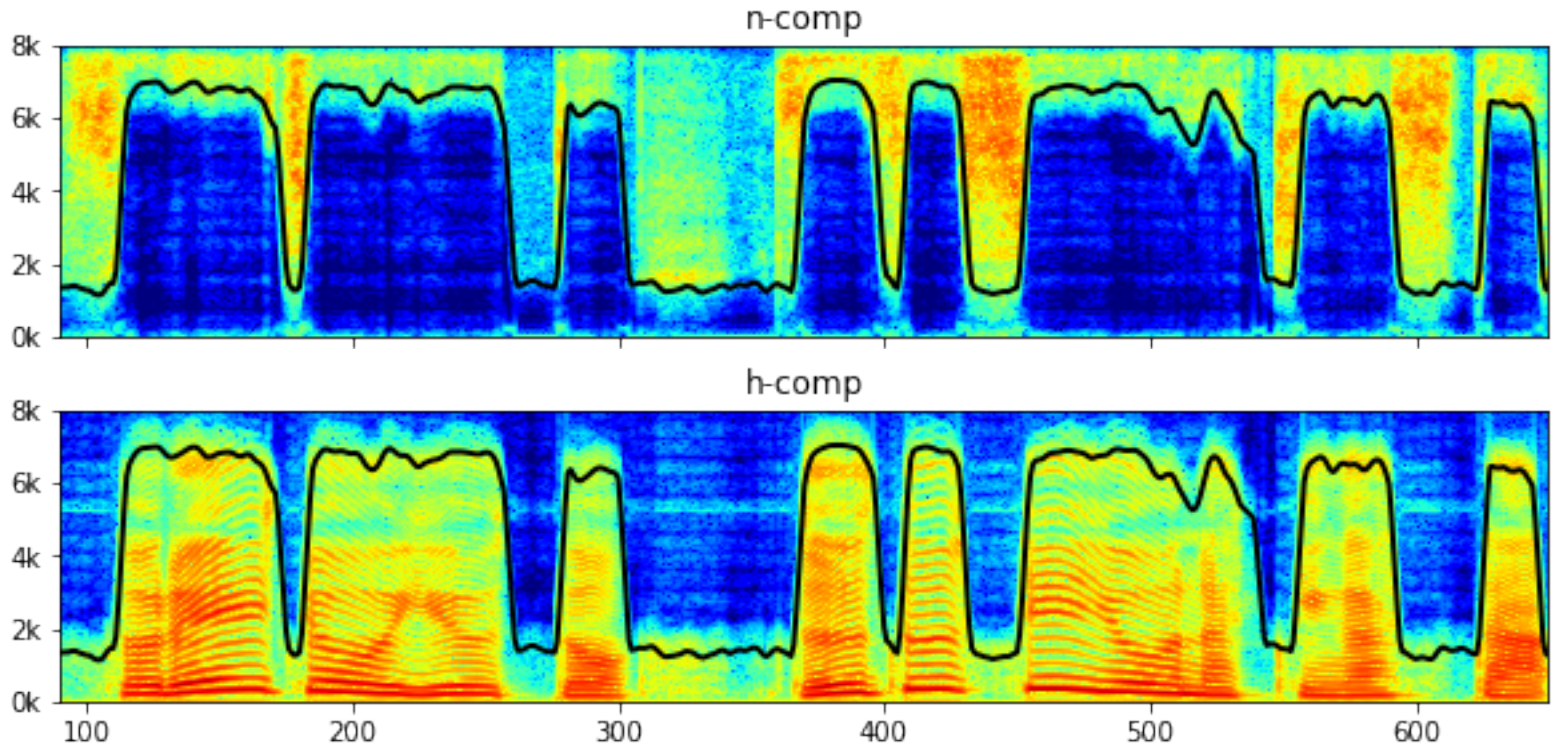


Pre-c

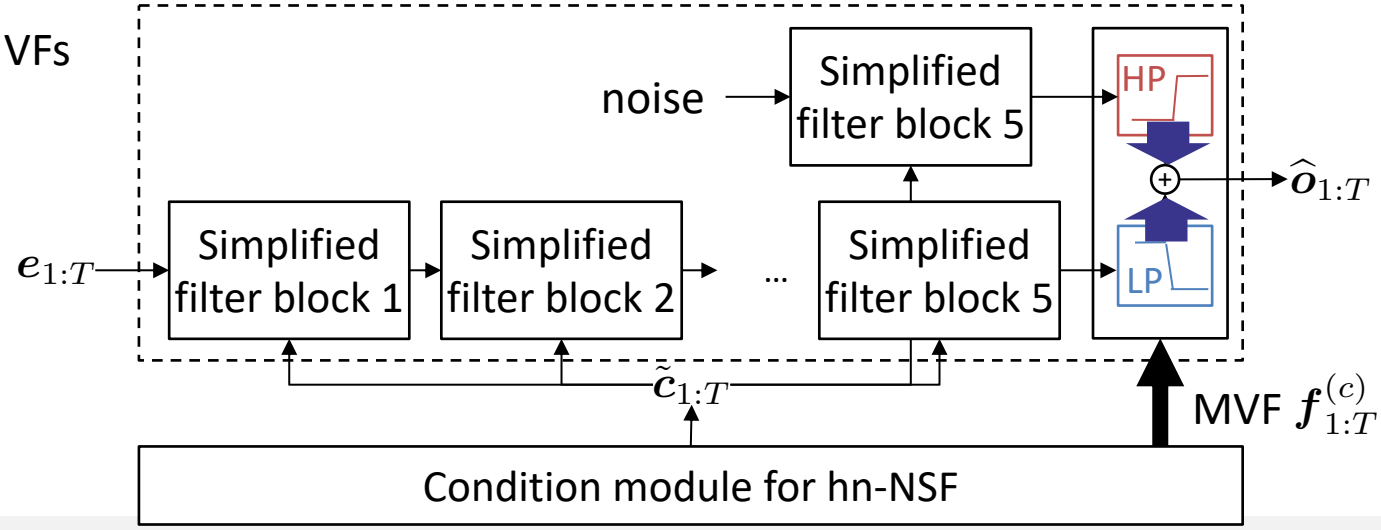
Predicted MVFs



Pre-c



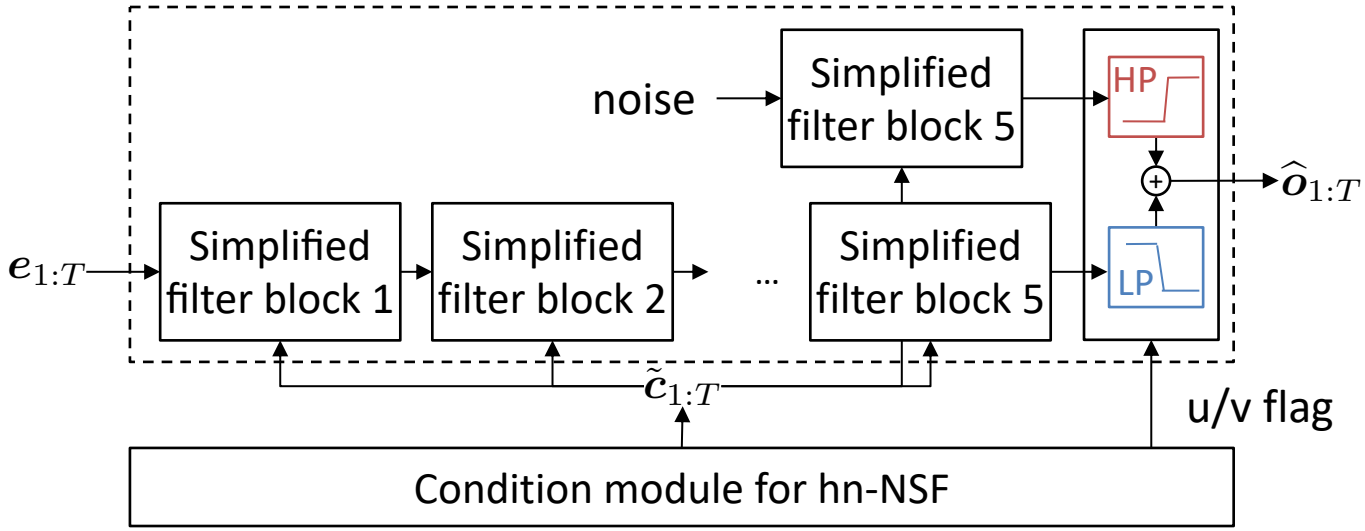
Predicted MVFs



NEURAL SOURCE-FILTER MODEL

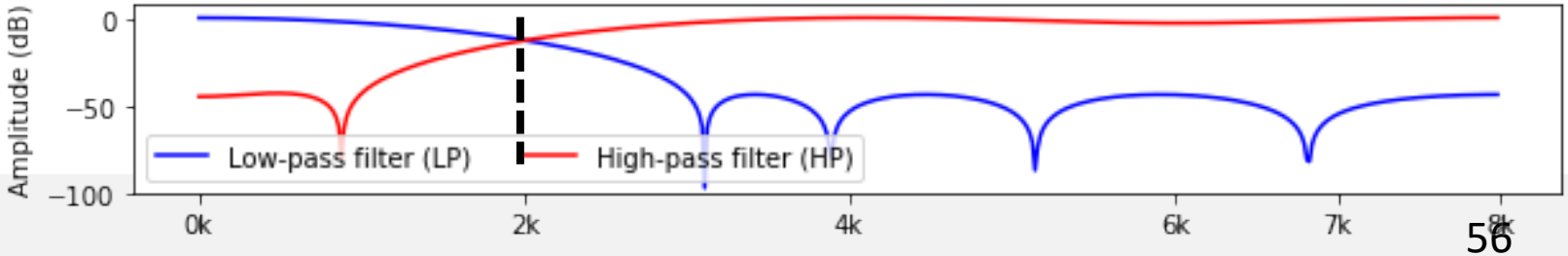
Harmonic-plus-noise NSF

- Version I: choose MVF based on u/v



Voiced sounds

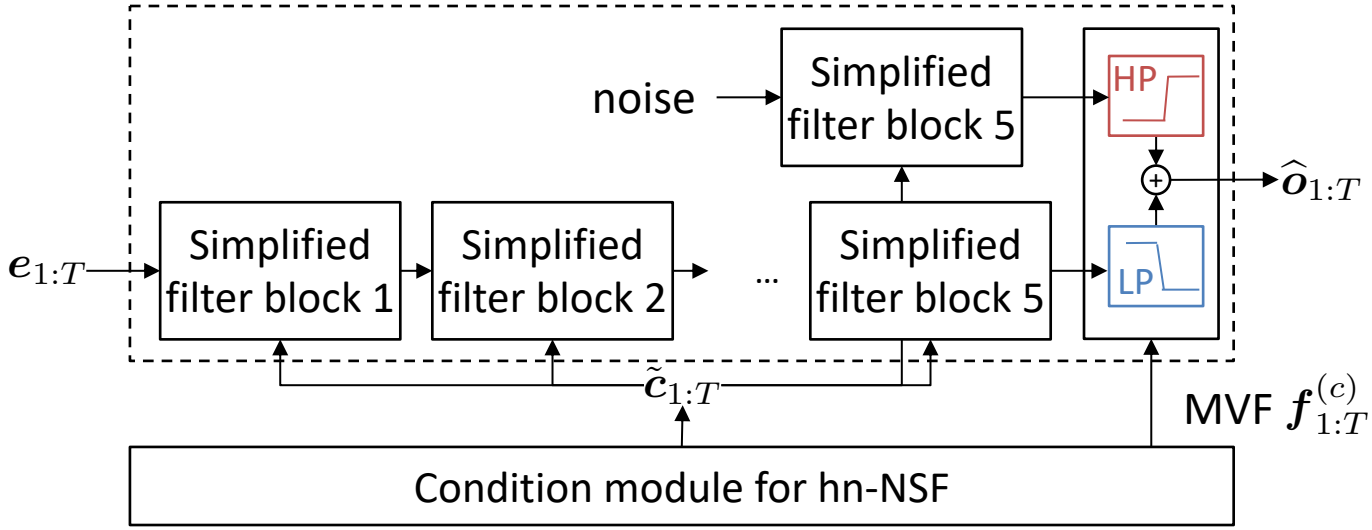
Unvoiced sounds



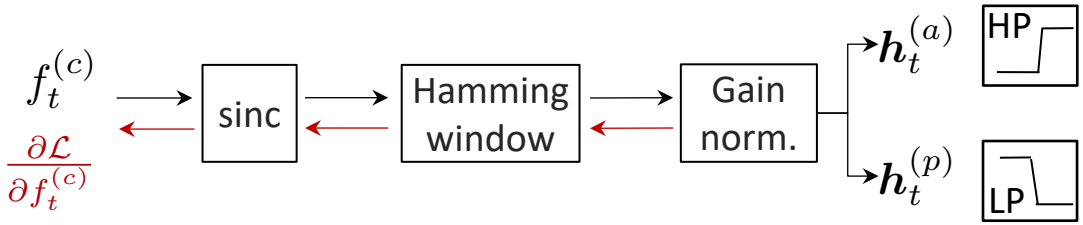
NEURAL SOURCE-FILTER MODEL

Harmonic-plus-noise NSF

- Version II: predict MVF from input features



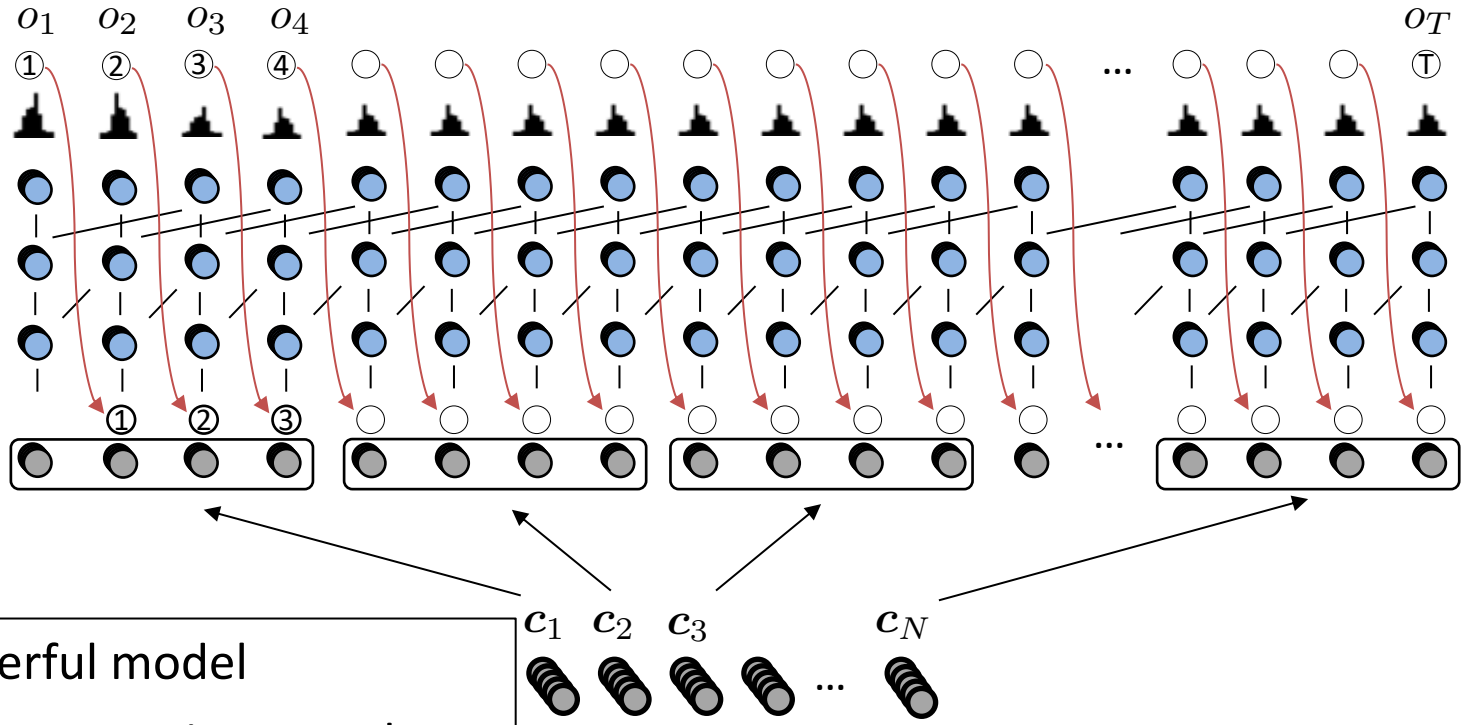
- Forward and backward propagation (SSW paper section 3)



INTRODUCTION

Autoregressive (AR) model

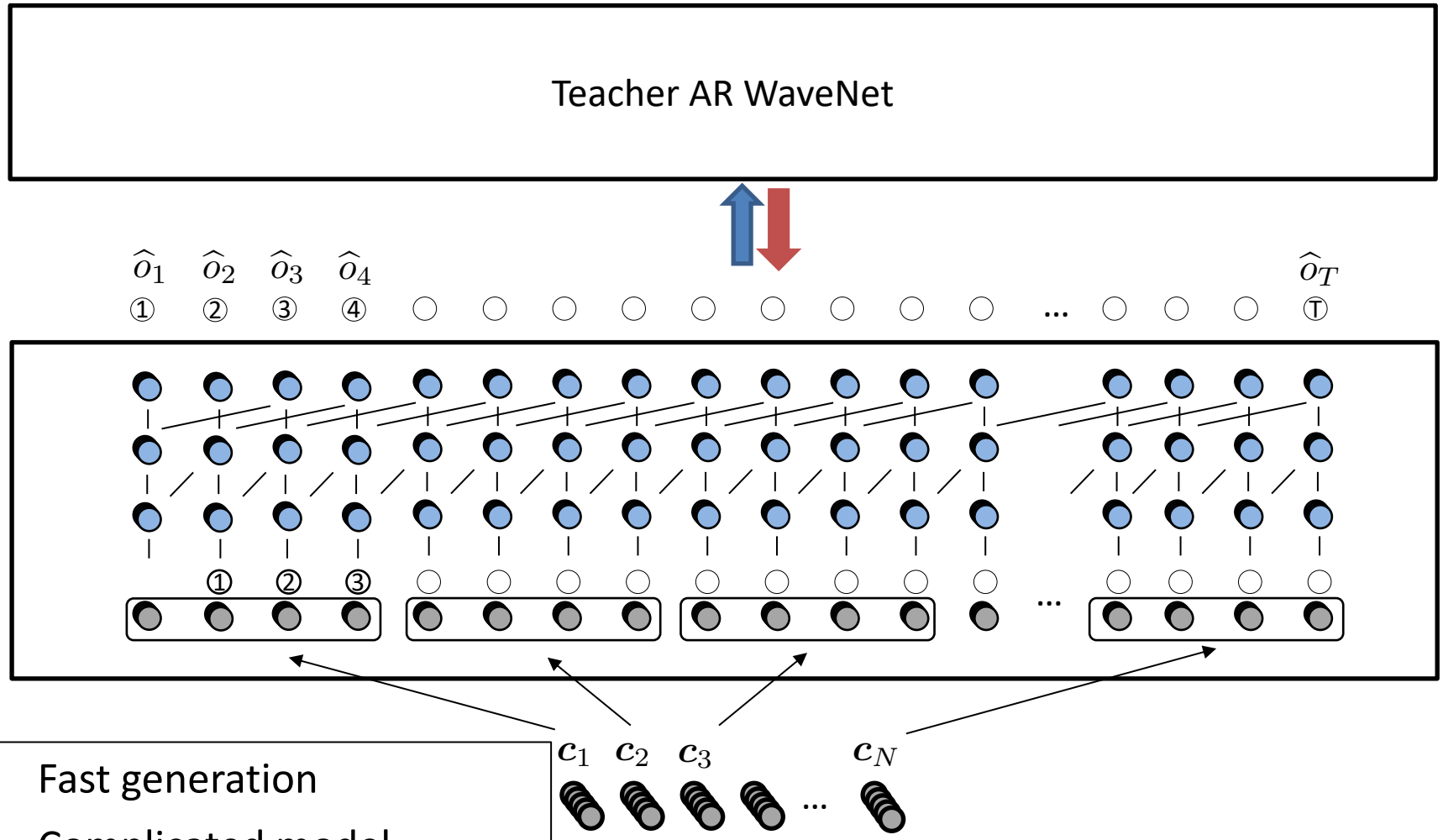
$$p(\boldsymbol{o}_{1:T}|\boldsymbol{c}_{1:N};\boldsymbol{\Theta}) = \prod_{t=1}^{T} p(\boldsymbol{o}_t|\boldsymbol{o}_{1:t-1},\boldsymbol{c}_{1:N};\boldsymbol{\Theta})$$



- Powerful model
- ! Slow generation speed

INTRODUCTION

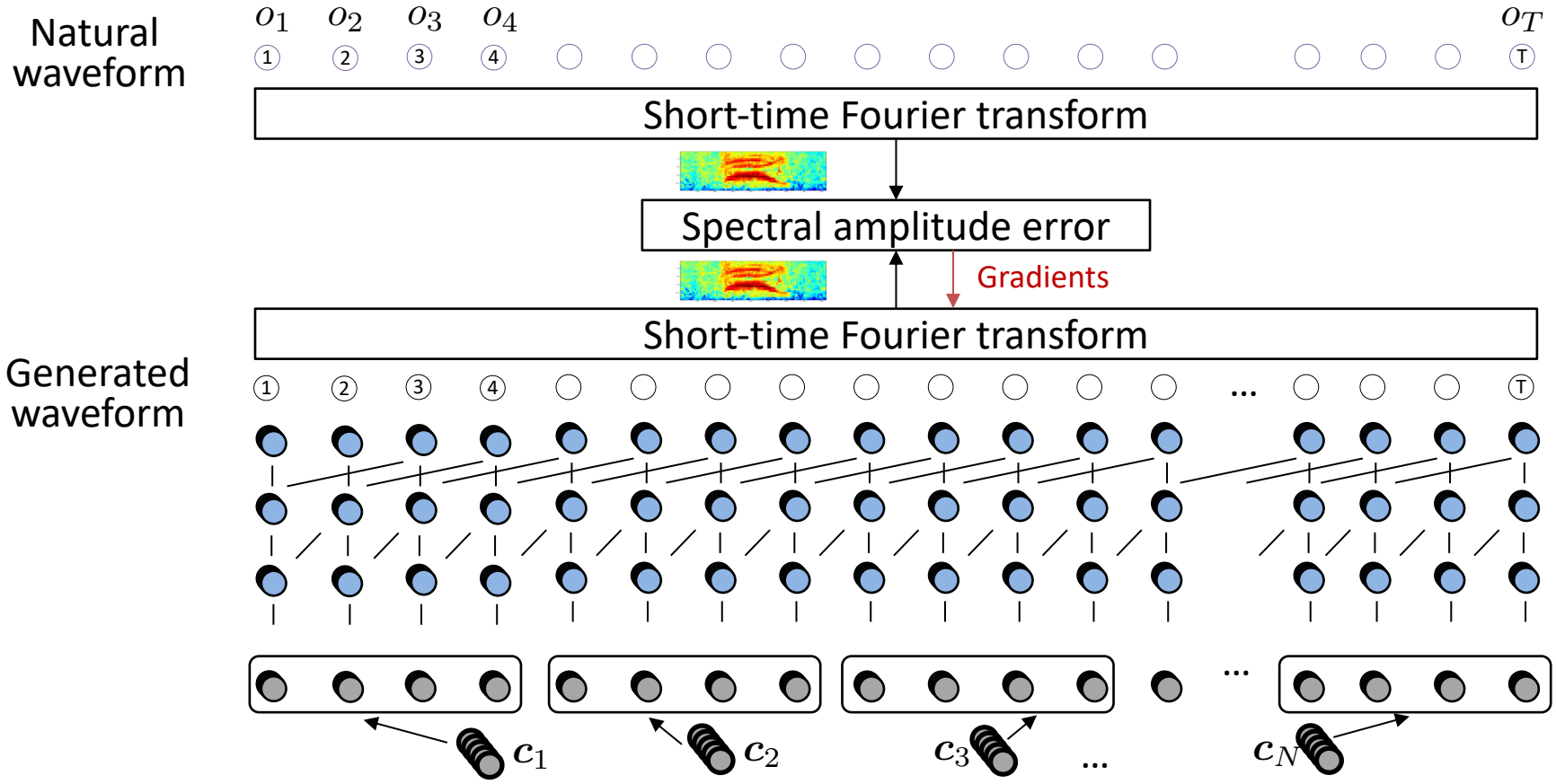
Normalizing flow + knowledge distilling



- Fast generation
- ! Complicated model

NEURAL SOURCE-FILTER MODEL

Idea 1: spectral domain criterion



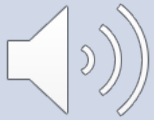
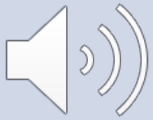
- Avoid naïve model assumption
- Differentiable & efficient computation

SUMMARY

Recent work

- ❑ Input F0?

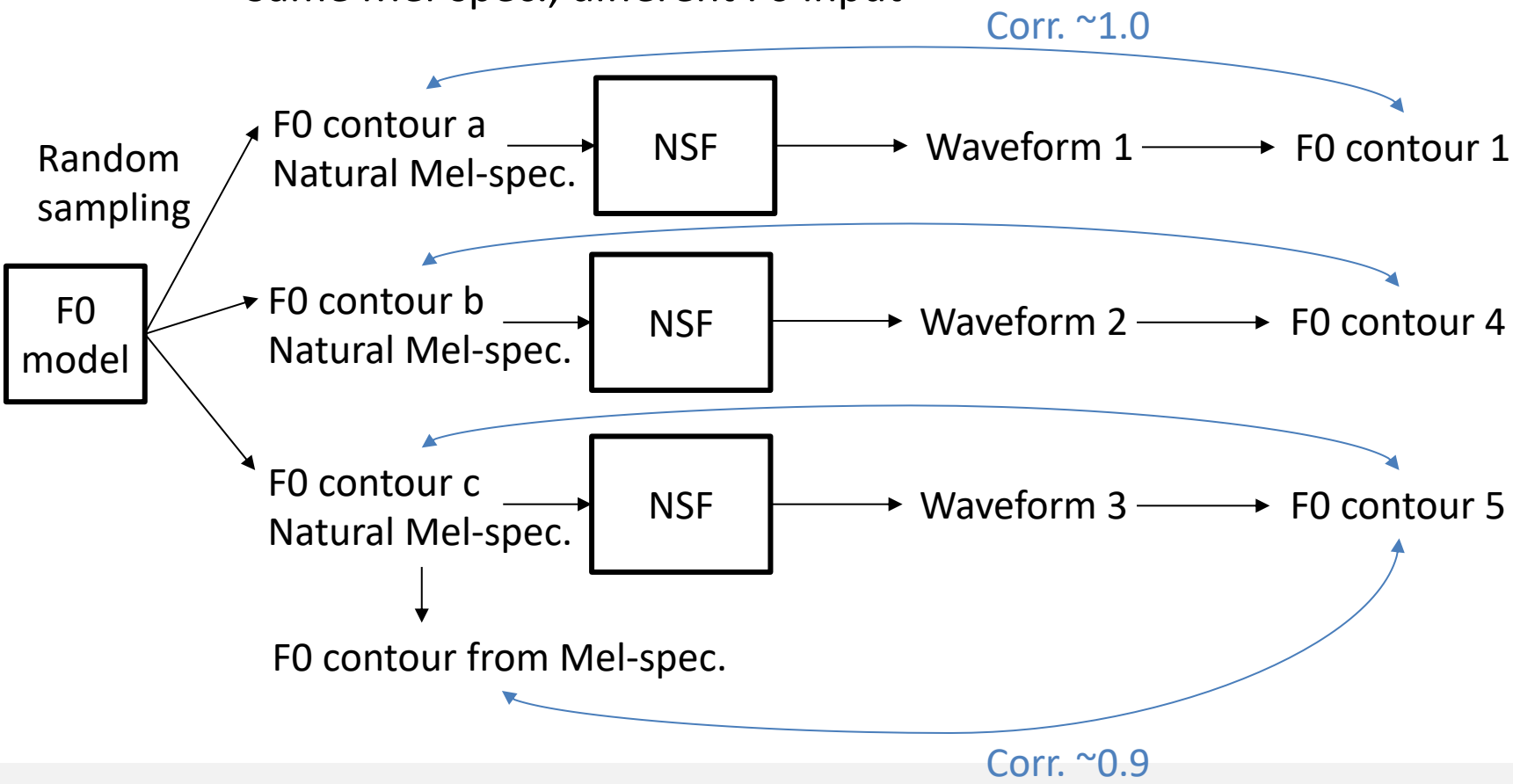
  - VCTK corpus, Mel-spectrogram + F0
  - Samples generated from NSF, by changing F0 only

F0	0.5 * F0	2.0 * F0
		
		

SUMMARY

Recent work

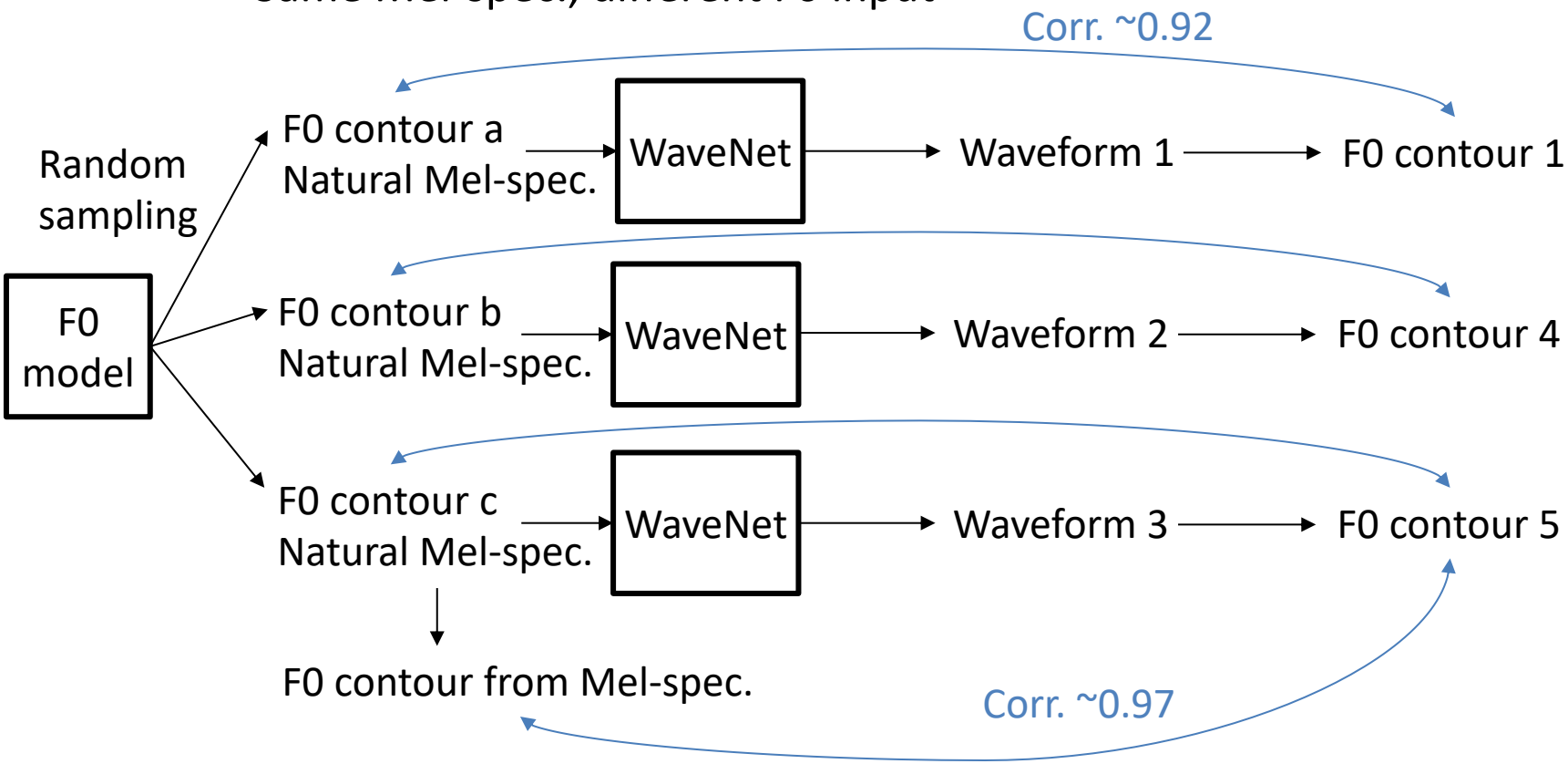
- Input F0?
  - https://arxiv.org/abs/1904.12088
  - Same Mel-spec., different F0 input



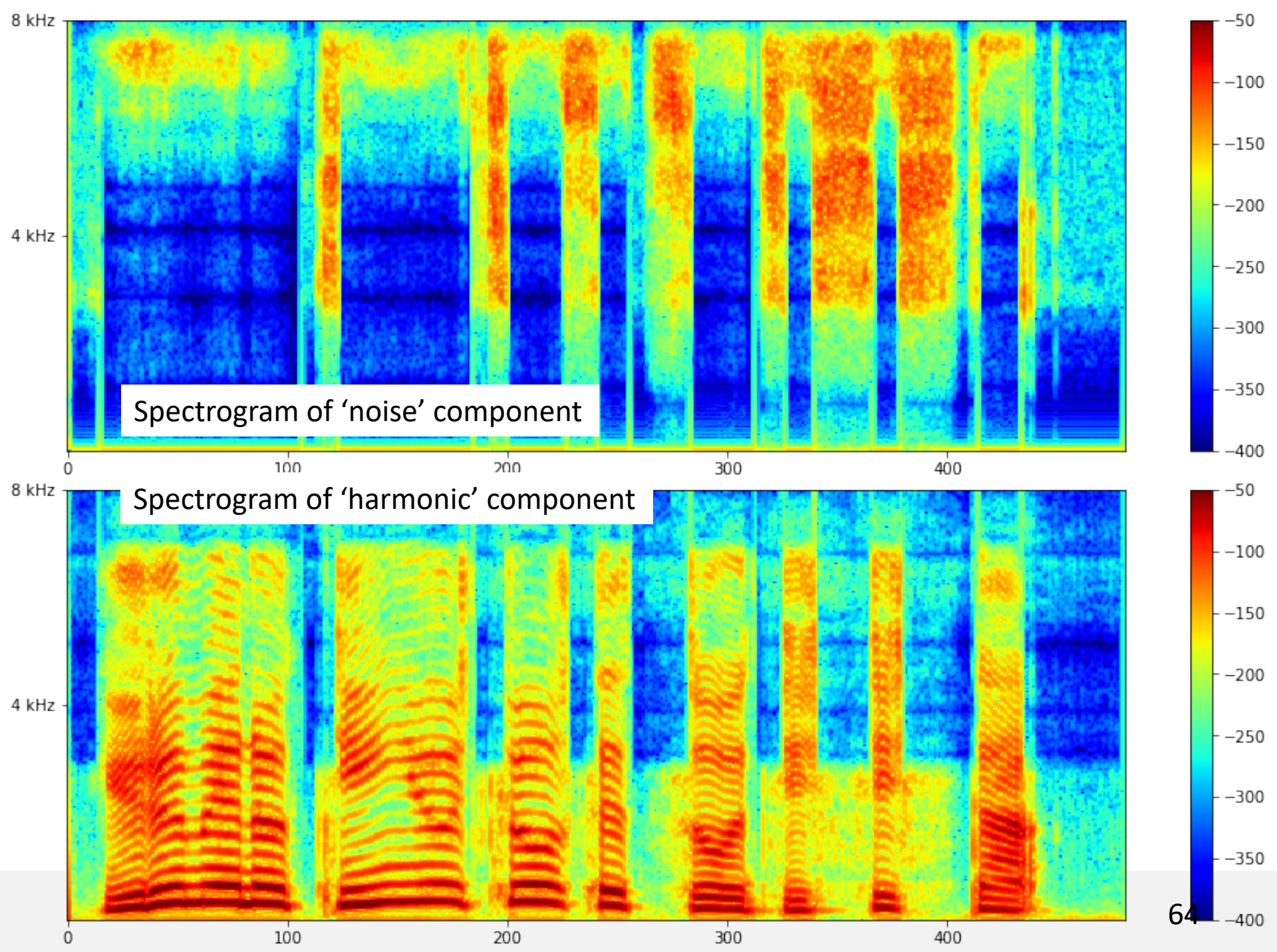
SUMMARY

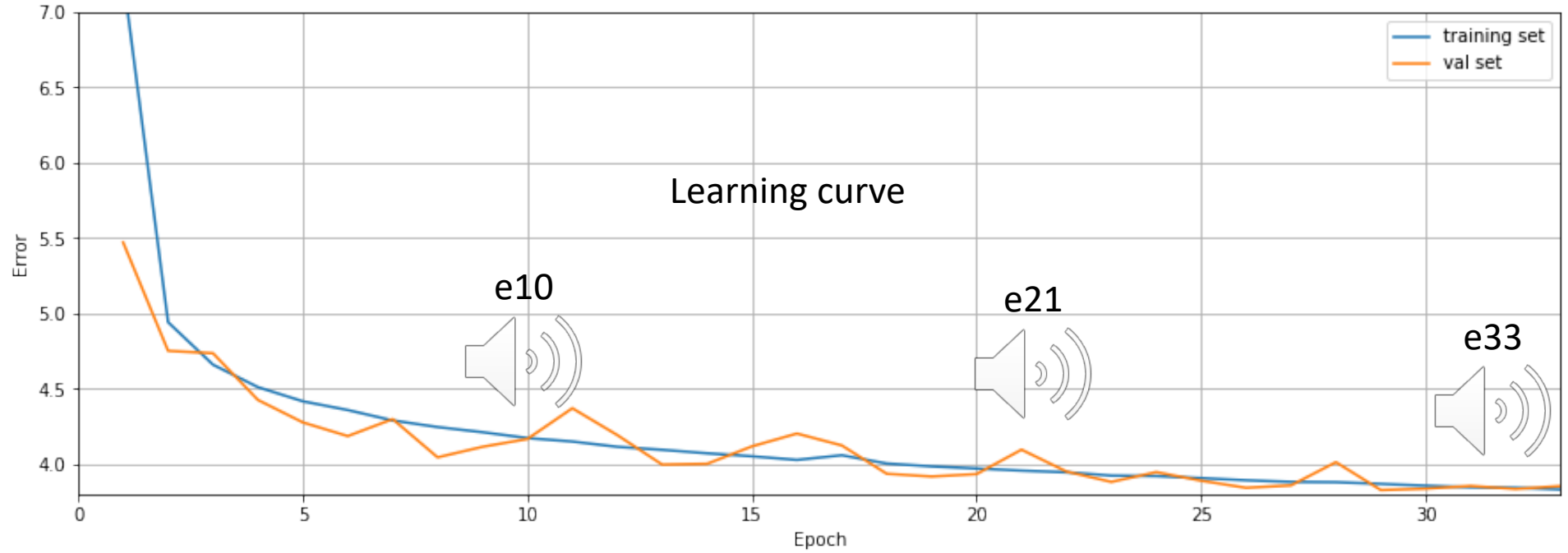
Recent work

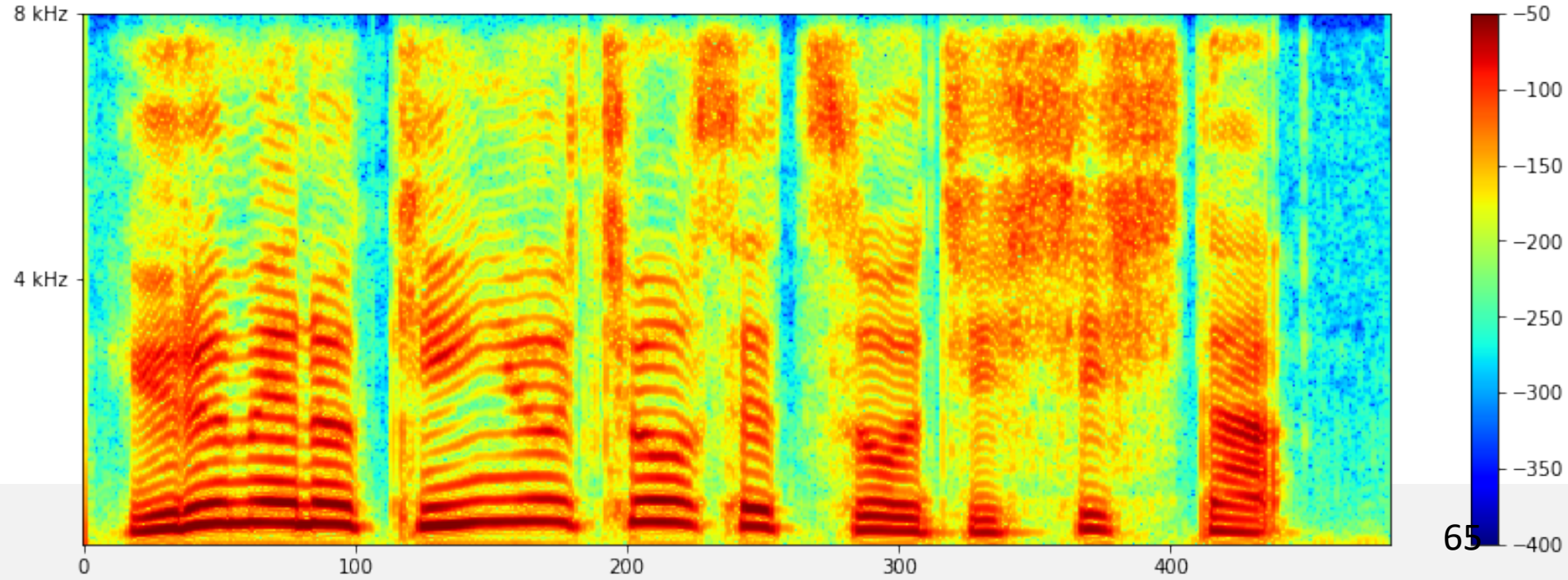
- Input F0?
  - https://arxiv.org/abs/1904.12088
  - Same Mel-spec., different F0 input



- Not easy to control F0 of generated waveforms from WaveNet

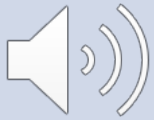

8 kHz
4 kHz
0
100
200
300
400
−50
−100
−150
−200
−250
−300
−350
−400
Spectrogram of ‘noise’ component
Spectrogram of ‘harmonic’ component


Learning curve
training set
val set
Error
Epoch
e10
e21
e33


8 kHz
4 kHz

SUMMARY

Samples (16kHz waveform, natural acoustic features)

Natural	Original NSF		Simplified NSF		Simplified & improved NSF	
	MGC + F0	Mel-spec + F0	MGC + F0	Mel-spec + F0	MGC + F0	Mel-spec + F0
						
						
						

- Natural MGC / Mel-spectrogram
- Natural F0

SUMMARY

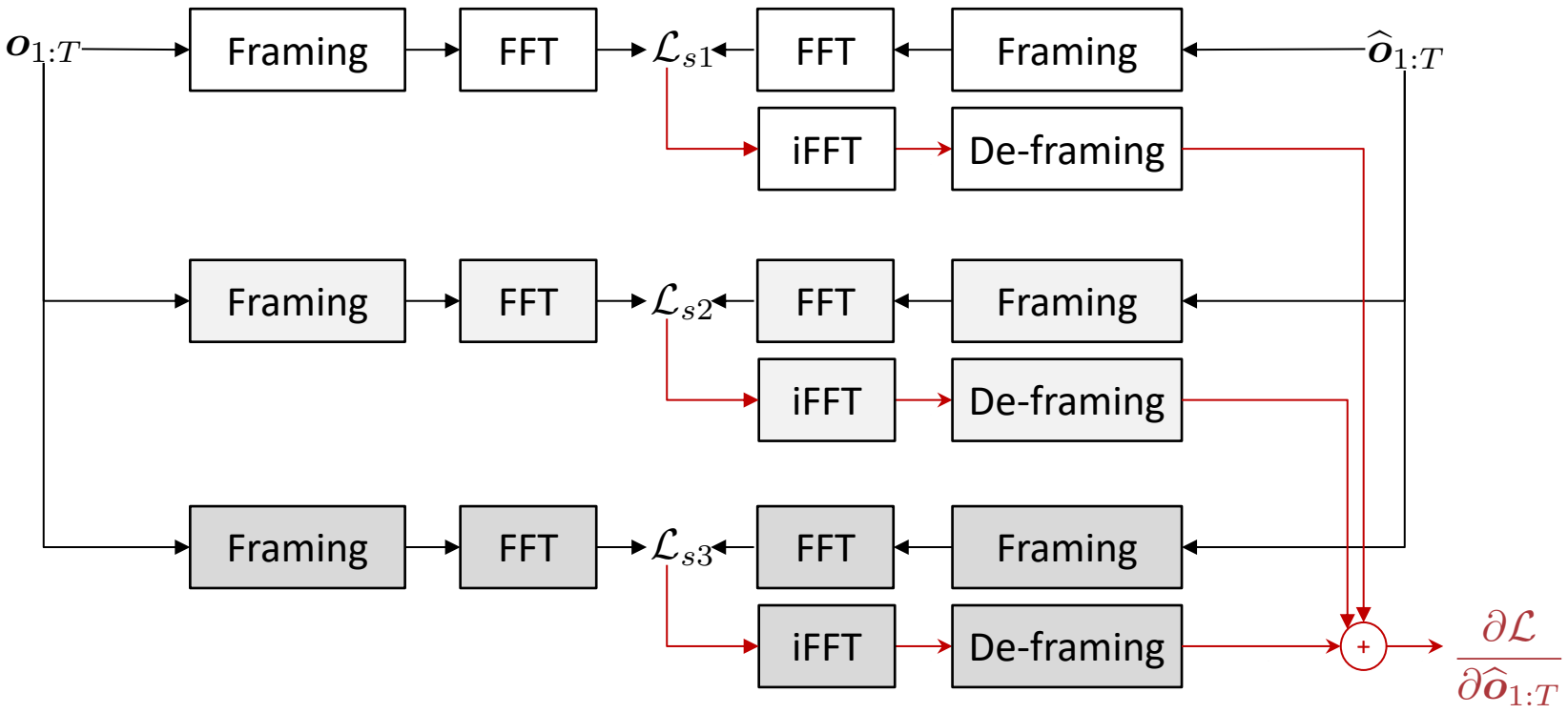
Samples (16kHz waveform, text-to-speech)

Natural	Original NSF		Simplified NSF		Simplified & improved NSF	
	MGC + F0	Mel-spec + F0	MGC + F0	Mel-spec + F0	MGC + F0	Mel-spec + F0
						
						
						

- Predicted MGC / Mel-spectrogram from text
- Natural F0

NEURAL SOURCE-FILTER MODEL

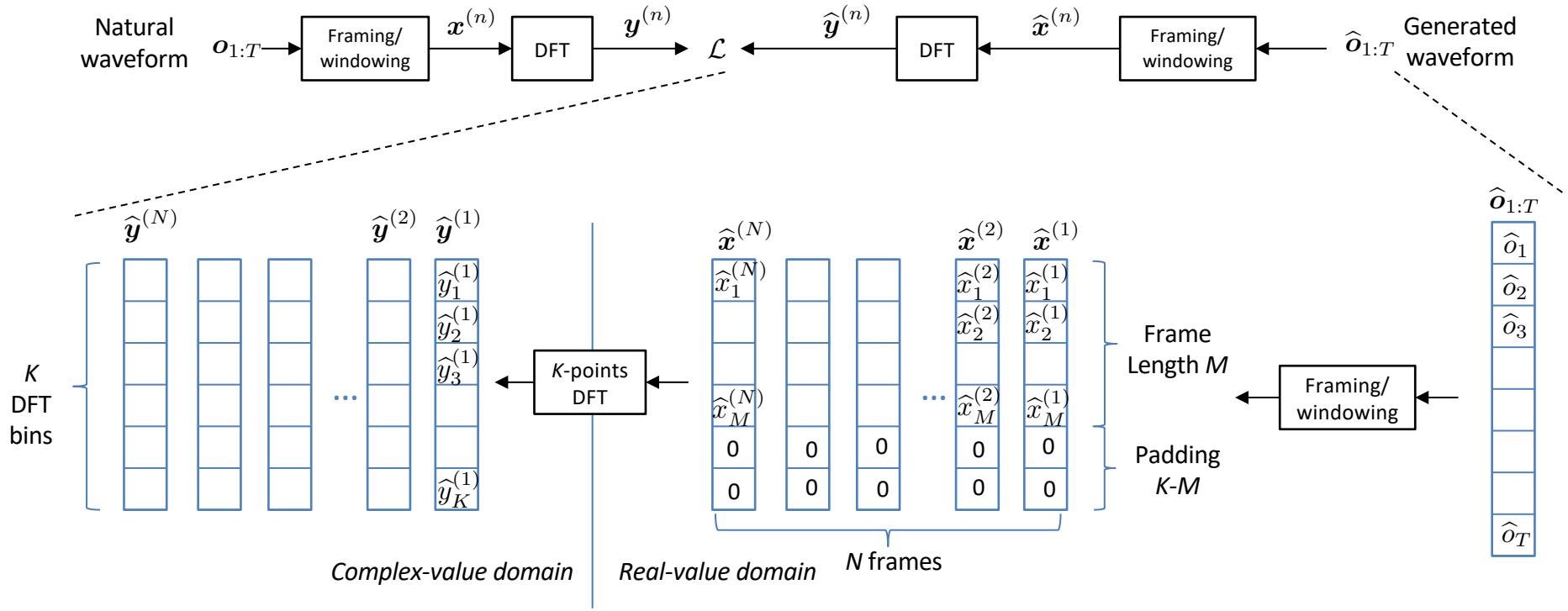
Training criterion



- Different frame shifts / window lengths / FFT points
- Homogenous distances $\mathcal{L} = \mathcal{L}_{s1} + \mathcal{L}_{s2} + \mathcal{L}_{s3}$
- $$\frac{\partial \mathcal{L}}{\partial \hat{\boldsymbol{o}}_{1:T}} = \frac{\partial \mathcal{L}_{s1}}{\partial \hat{\boldsymbol{o}}_{1:T}} + \frac{\partial \mathcal{L}_{s2}}{\partial \hat{\boldsymbol{o}}_{1:T}} + \frac{\partial \mathcal{L}_{s3}}{\partial \hat{\boldsymbol{o}}_{1:T}}$$

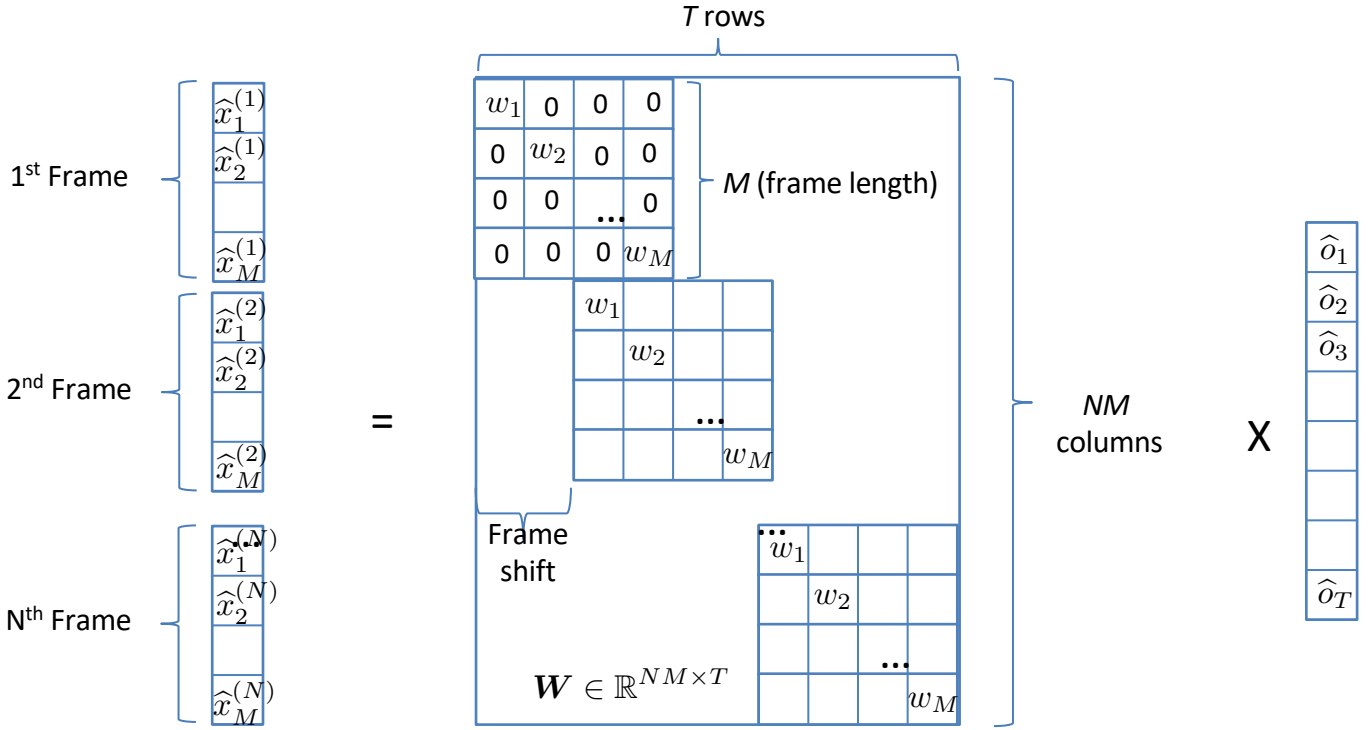
NEURAL SOURCE-FILTER MODEL

Training criterion



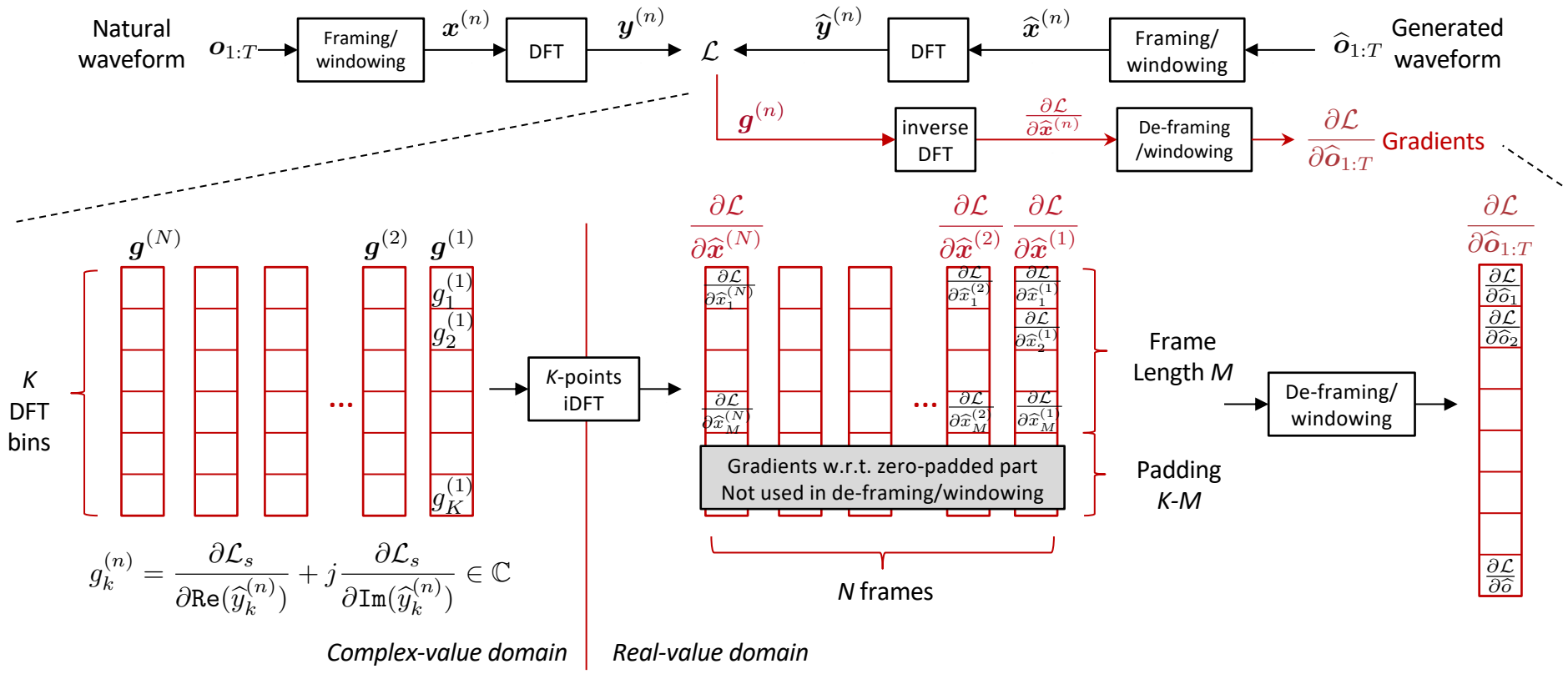
NEURAL SOURCE-FILTER MODEL

Training criterion



NEURAL SOURCE-FILTER MODEL

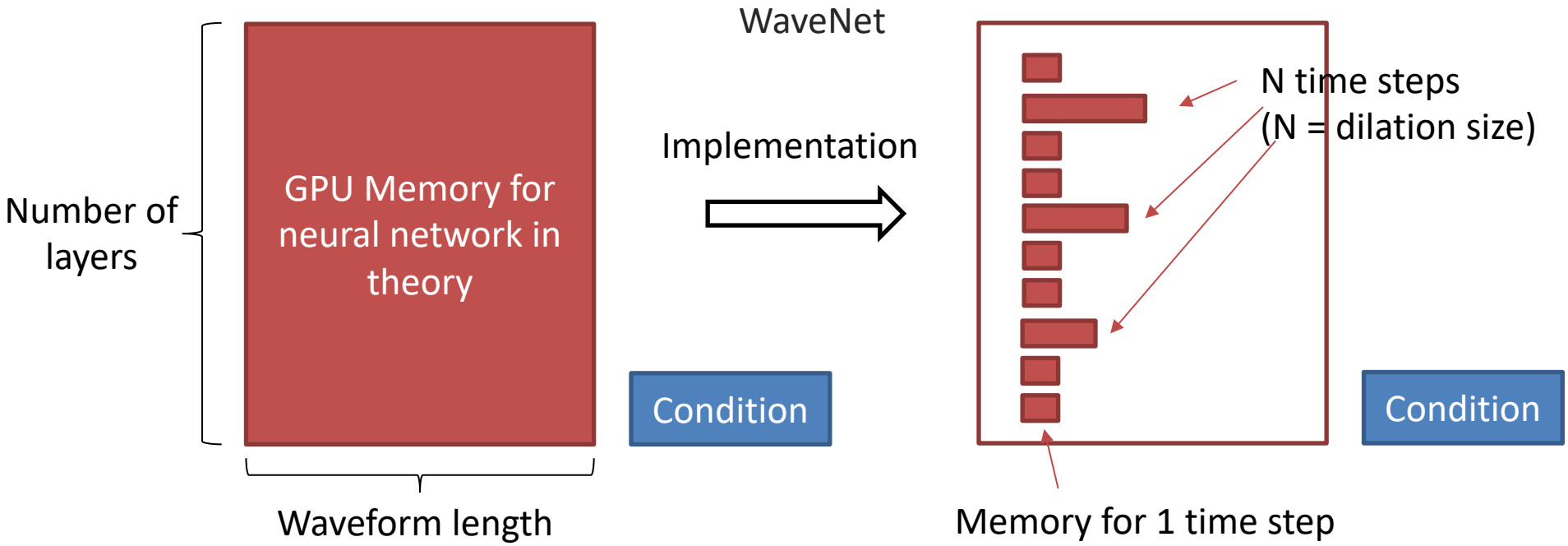
Training criterion



PART 3

Implementation issue

- WaveNet memory consumption in generation

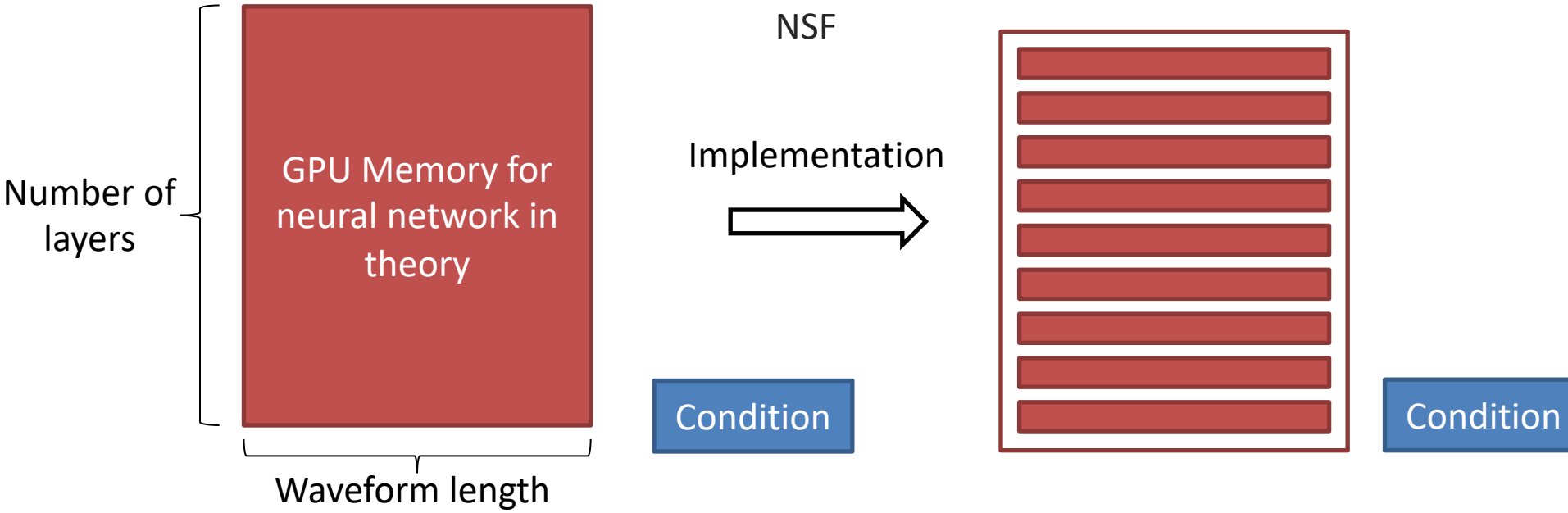


- Only allocate GPU memory for generating 1 waveform point
- For t = 1 : T, reuse the memory space
- Check: http://tonywangx.github.io/pdfs/CURRENNT_WAVENET.pdf (pp.44-55)

PART 3

Implementation issue

- NSF normal node

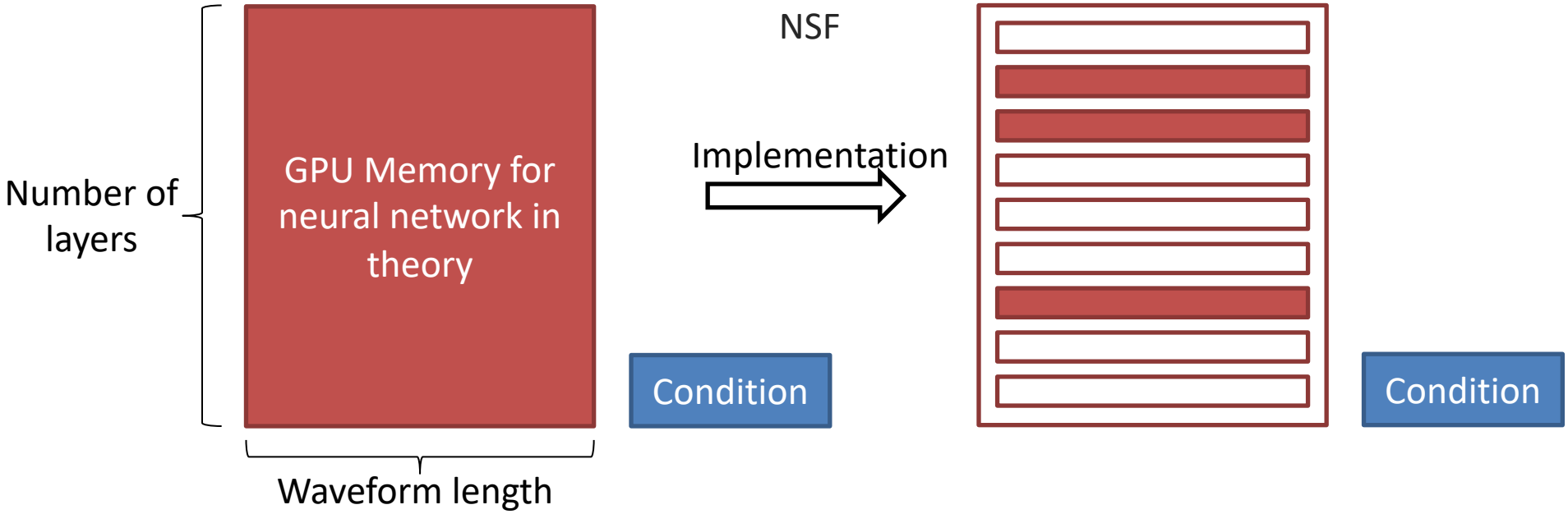


- Normal mode: allocate GPU memory for all layers and all time steps
- GPU memory consumption is large for long waveforms

PART 3

Implementation issue

- NSF memory-save node

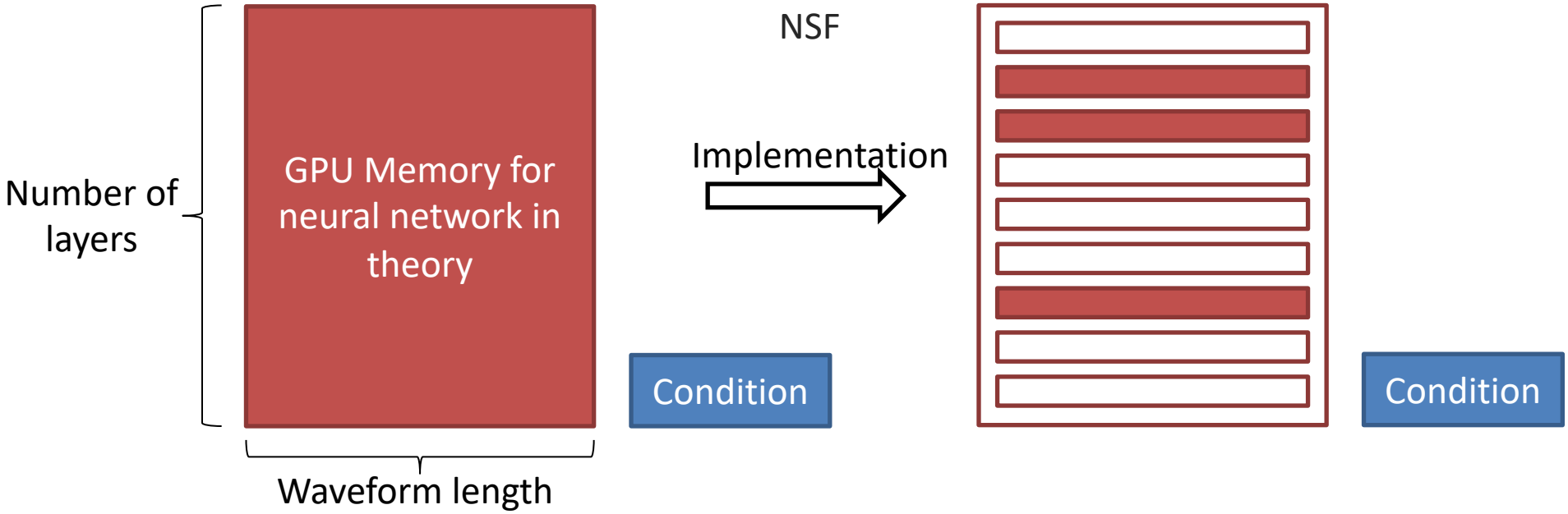


- Memory-save mode:
  - Allocate GPU memory for the current layer and all previous layers it depends on
  - Release GPU memory when one layer is no longer needed
- GPU memory consumption is small

PART 3

Implementation issue

- NSF memory-save node

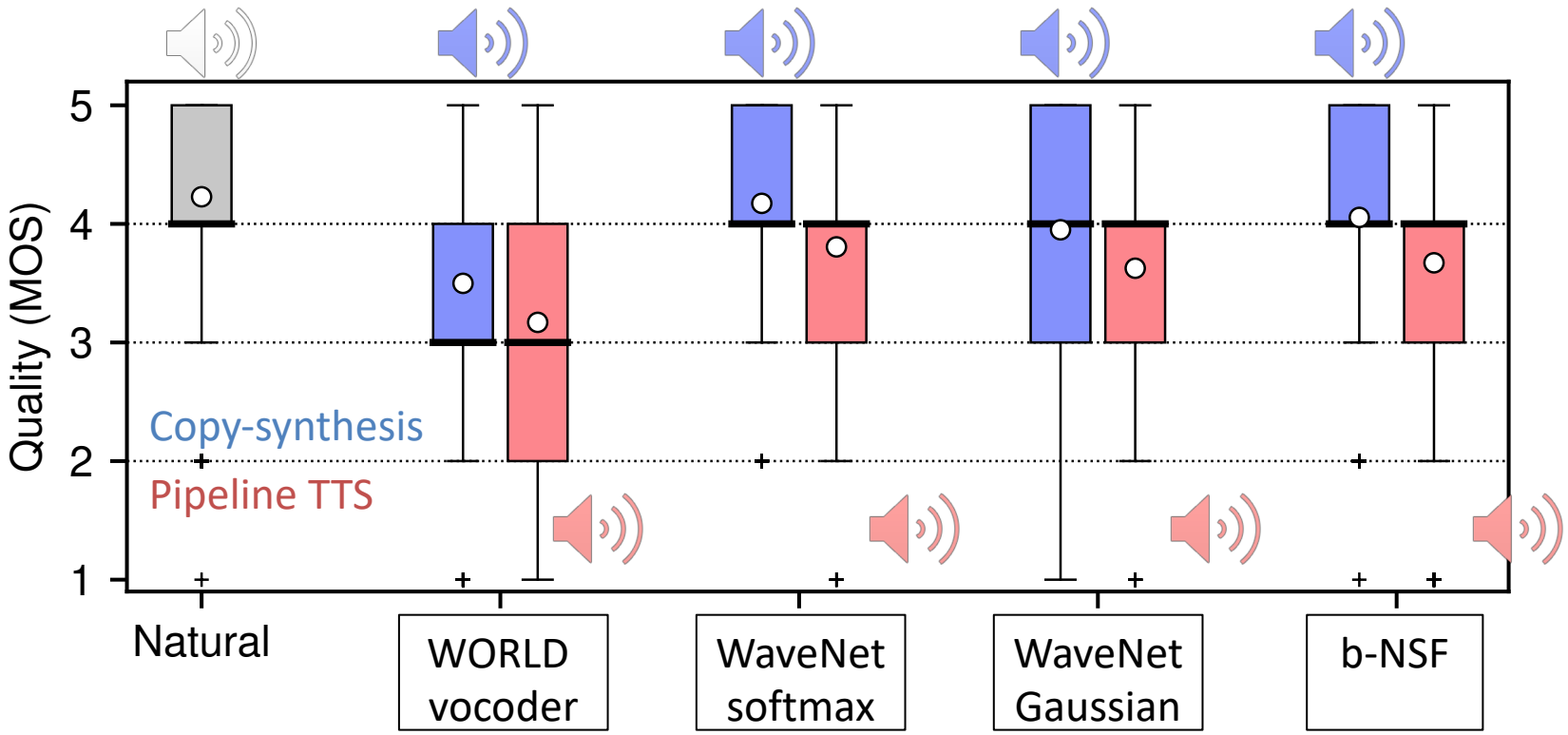


Memory-save mode:
Function __computeGenPass_LayerByLayer_mem
  Build layer-dependency graph
  For layer_n = 1 : L
    Allocate GPU memory and compute layer_n's output
    For layer_m = 1 : layer_n where layer_m's output is input to layer_n
      If layer_m is no longer needed by any layer, release memory of layer_m

PRACTICE: COMPARISON

Speech quality (ICASSP)

WORLD vocoder | WaveNet softmax | WaveNet Gaussian | b-NSF | s-NSF | hn-NSF fixed MVF | hn-NSF trainable MVF



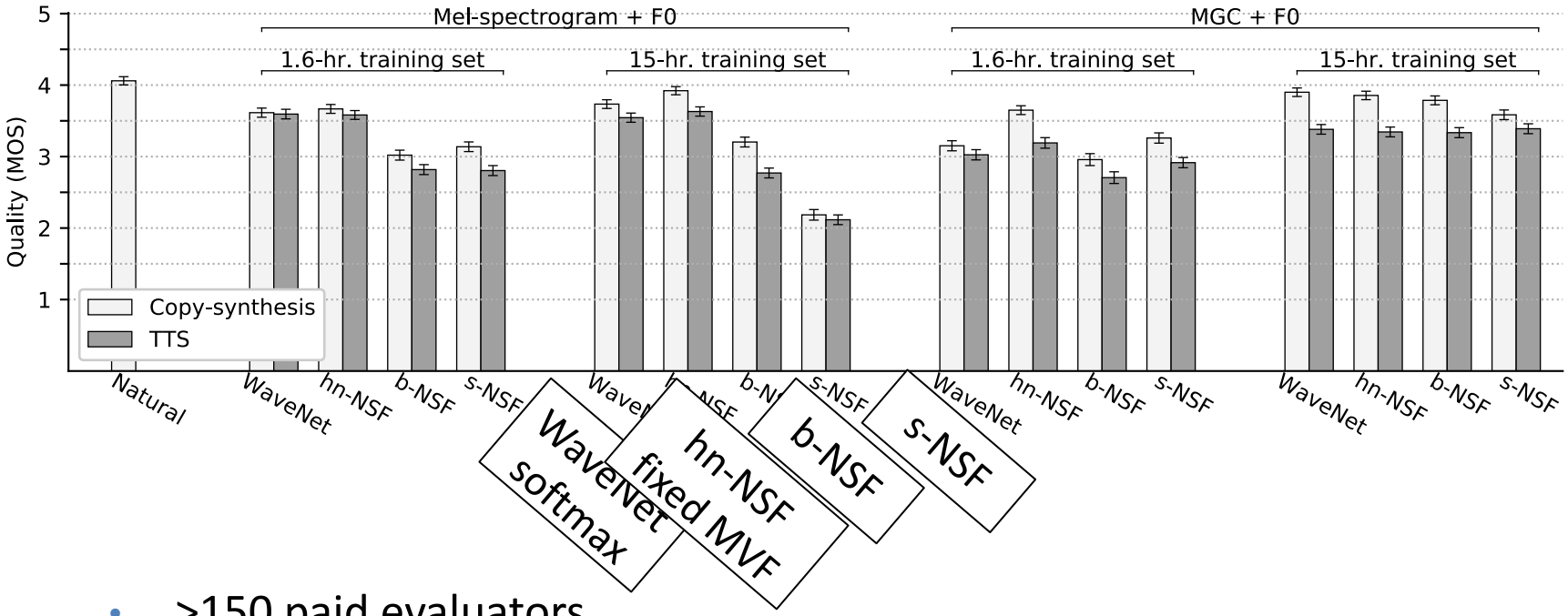
- 245 paid evaluators, 1450 evaluation sets

☛ Samples, models, codes: https://nii-yamagishilab.github.io/samples-nsf/nsf-v1.html

PRACTICE: COMPARISON

Speech quality (Journal paper submitted)

WORLD vocoder	WaveNet softmax	WaveNet Gaussian	b-NSF	s-NSF	hn-NSF fixed MVF	hn-NSF trainable MVF



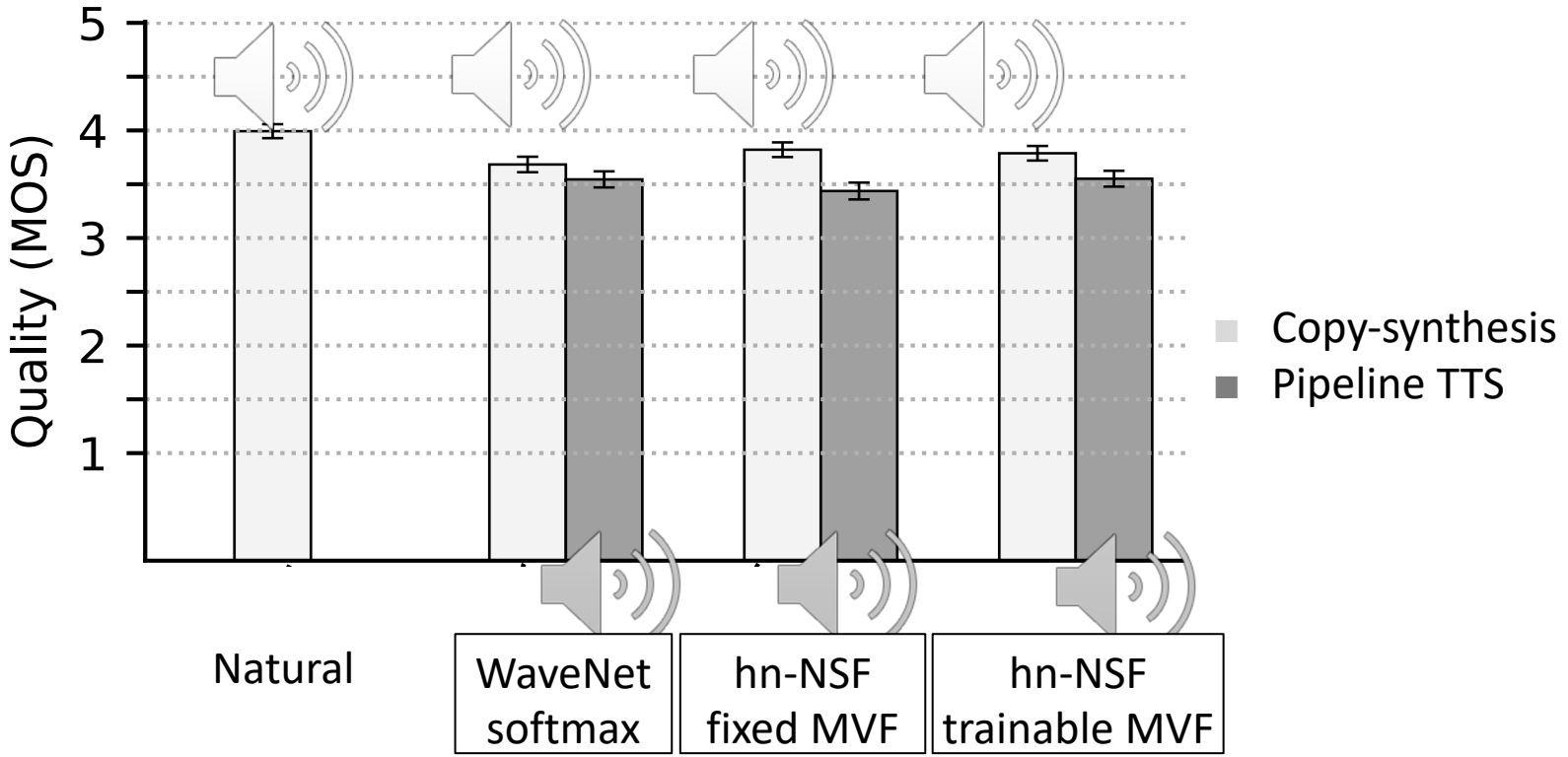
- >150 paid evaluators
- s-NSF did badly on unvoiced sounds

☛ Samples, models, codes: https://nii-yamagishilab.github.io/samples-nsf/nsf-v2.html

PRACTICE: COMPARISON

Speech quality (SSW 2019)

WORLD vocoder	WaveNet softmax	WaveNet Gaussian	b-NSF	s-NSF	hn-NSF fixed MVF	hn-NSF trainable MVF



- >150 paid evaluators

☛ Samples, models, codes: https://nii-yamagishilab.github.io/samples-nsf/nsf-v3.html

EXPERIMENTS

Test 2: Ablation test

- One more corpus

Corpus	Size	Note
CMU-arctic SLT [2]	0.83 hours	16kHz, English

	Feature	Dimension
Acoustic	Mel-spectrogram	80
	F0	1

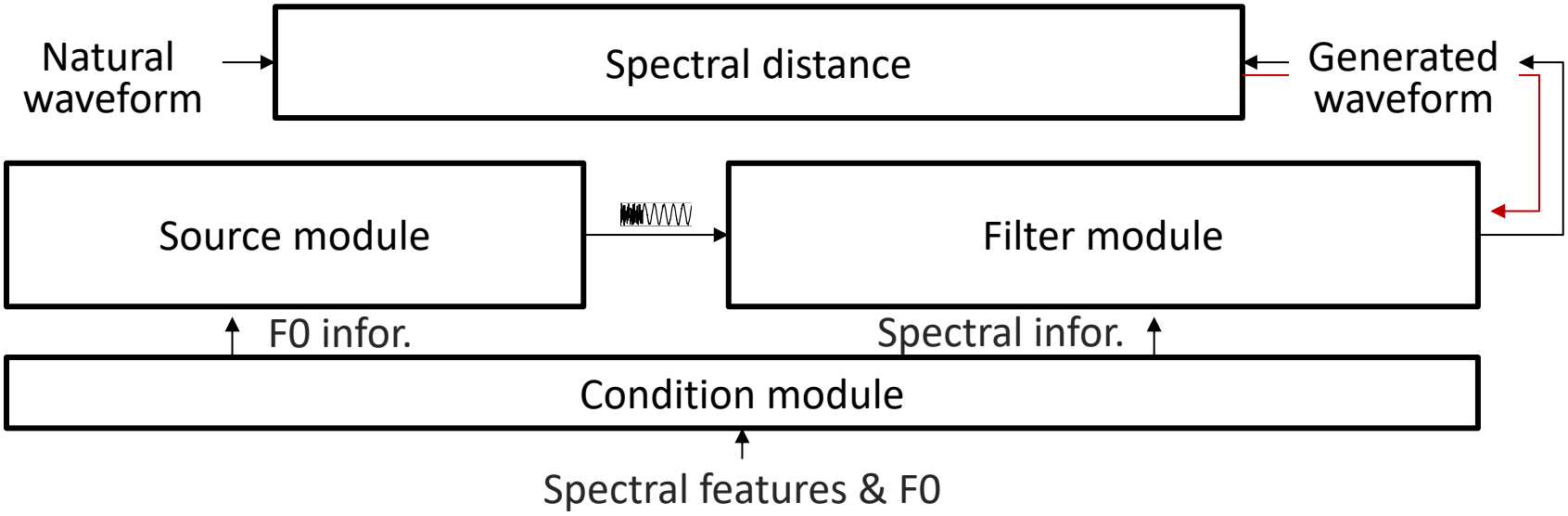
- Copy-synthesis

[2] J. Kominek and A. W. Black. The cmu arctic speech databases. In Fifth ISCA workshop on speech synthesis, 2004.

EXPERIMENTS

Test 2: Ablation test

	Natural	NSF	
ATR Ximera F009			
CMU-arctic SLT			

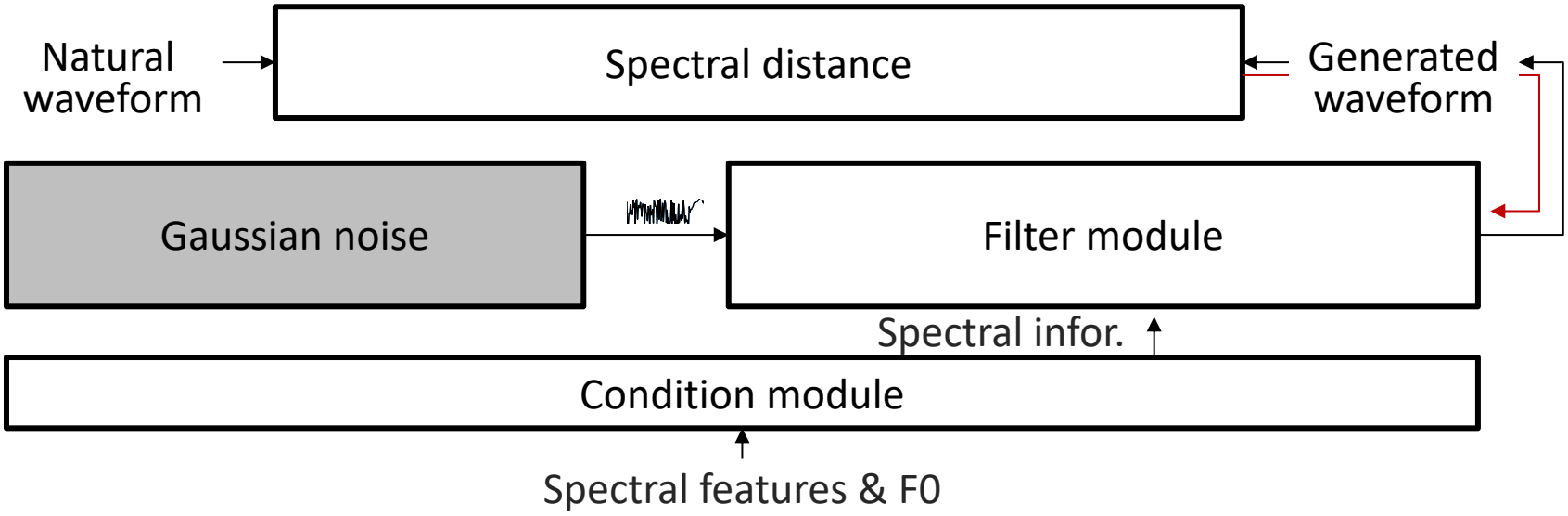


☛ Samples, models, codes: https://nii-yamagishilab.github.io/samples-nsf/index.html

EXPERIMENTS

Test 2: Ablation test

	Natural	NSF	NSF w/o sine excitation
ATR Ximera F009			
CMU-arctic SLT			

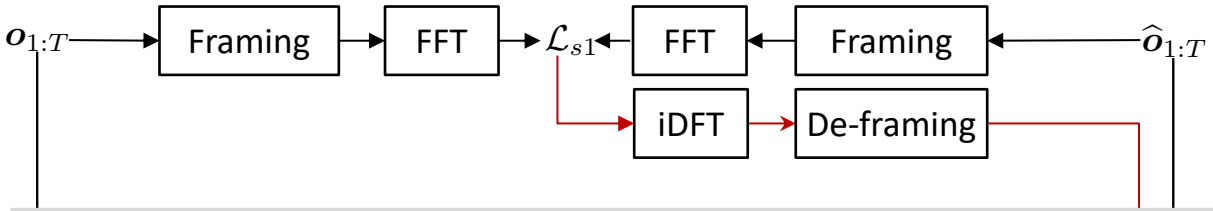


☛ Samples, models, codes: https://nii-yamagishilab.github.io/samples-nsf/index.html

EXPERIMENTS

Test 2: Ablation test

	Natural	NSF	NSF only L_{s1}	
ATR Ximera F009				
CMU-arctic SLT				



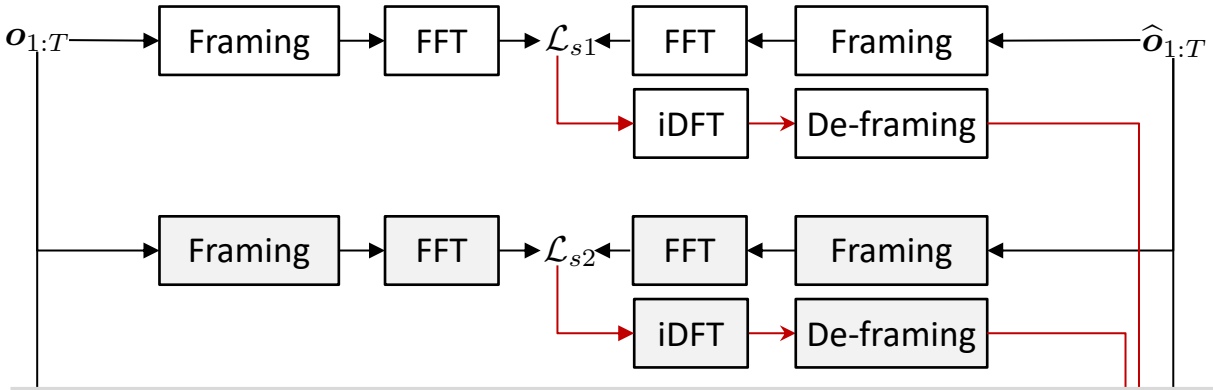
L_{s1}: frame shift 5ms, window length 20ms

☛ Samples, models, codes: https://nii-yamagishilab.github.io/samples-nsf/index.html

EXPERIMENTS

Test 2: Ablation test

	Natural	NSF	NSF only L_{s1}	NSF L_{s1}+L_{s2}
ATR Ximera F009				
CMU-arctic SLT				



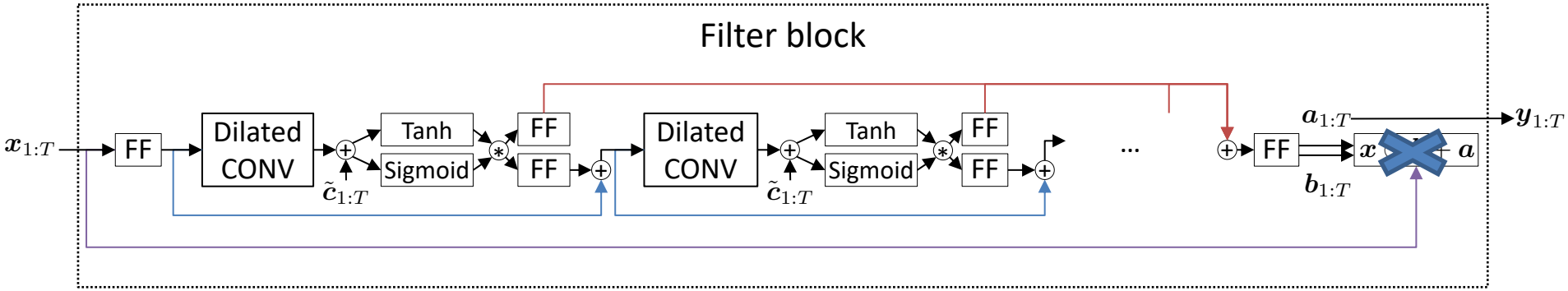
L_{s1}: frame shift 5ms, window length 20ms
L_{s2}: frame shift 2.5ms, window length 5ms

☛ Samples, models, codes: https://nii-yamagishilab.github.io/samples-nsf/index.html

EXPERIMENTS

Test 2: Ablation test

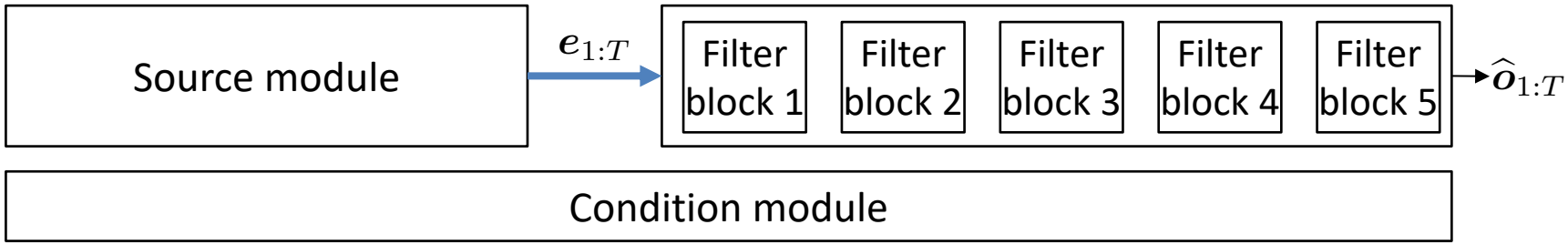
	Natural	NSF	NSF w/o 'residual' connection in filter module
F009			
CMU-arctic			

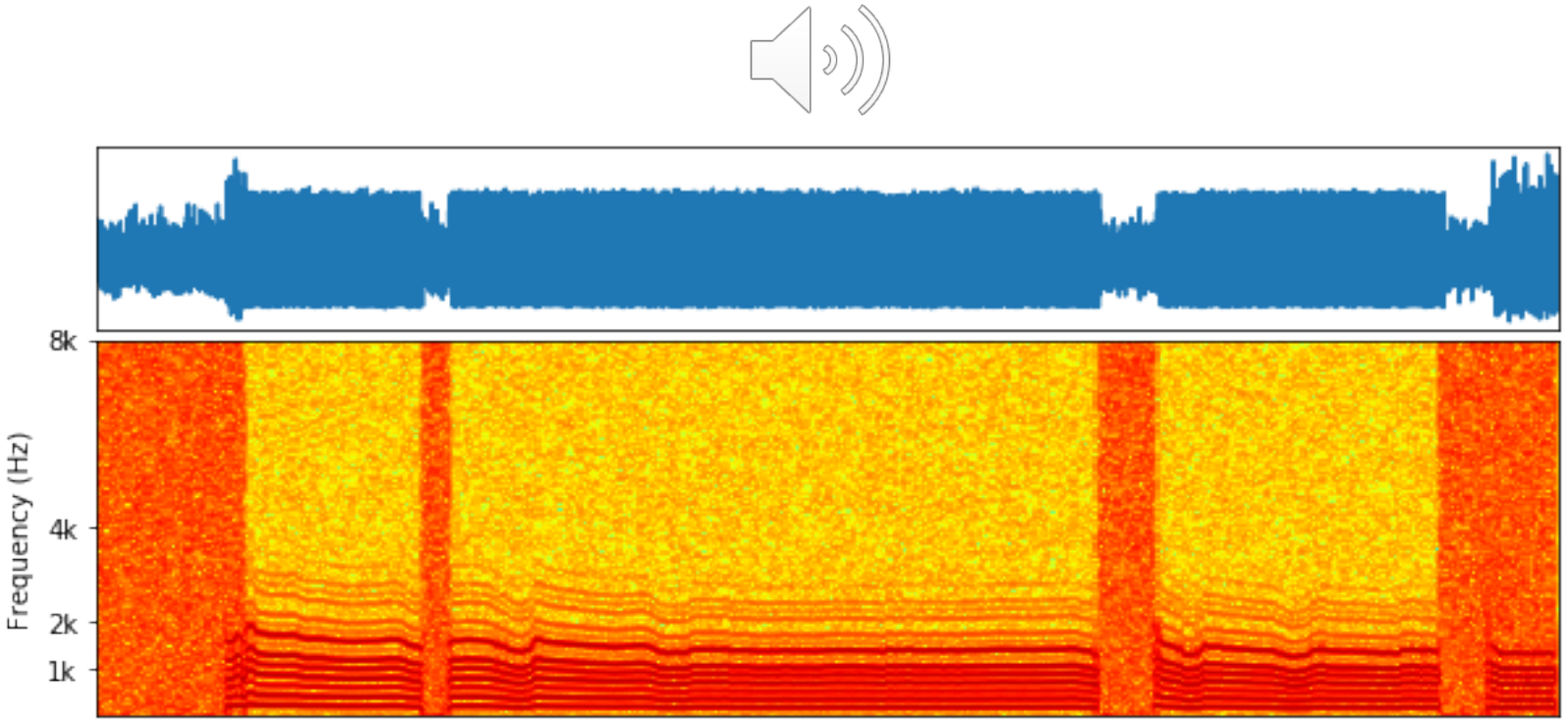


https://nii-yamagishilab.github.io/samples-nsf/index.html

EXPERIMENTS

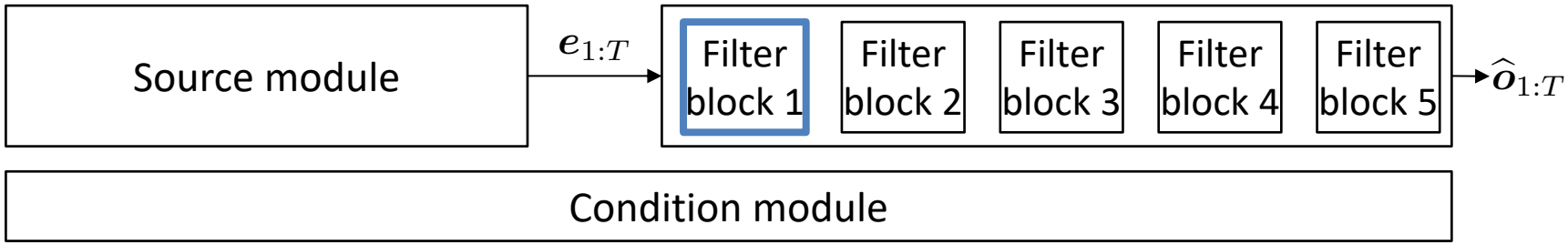
Analysis

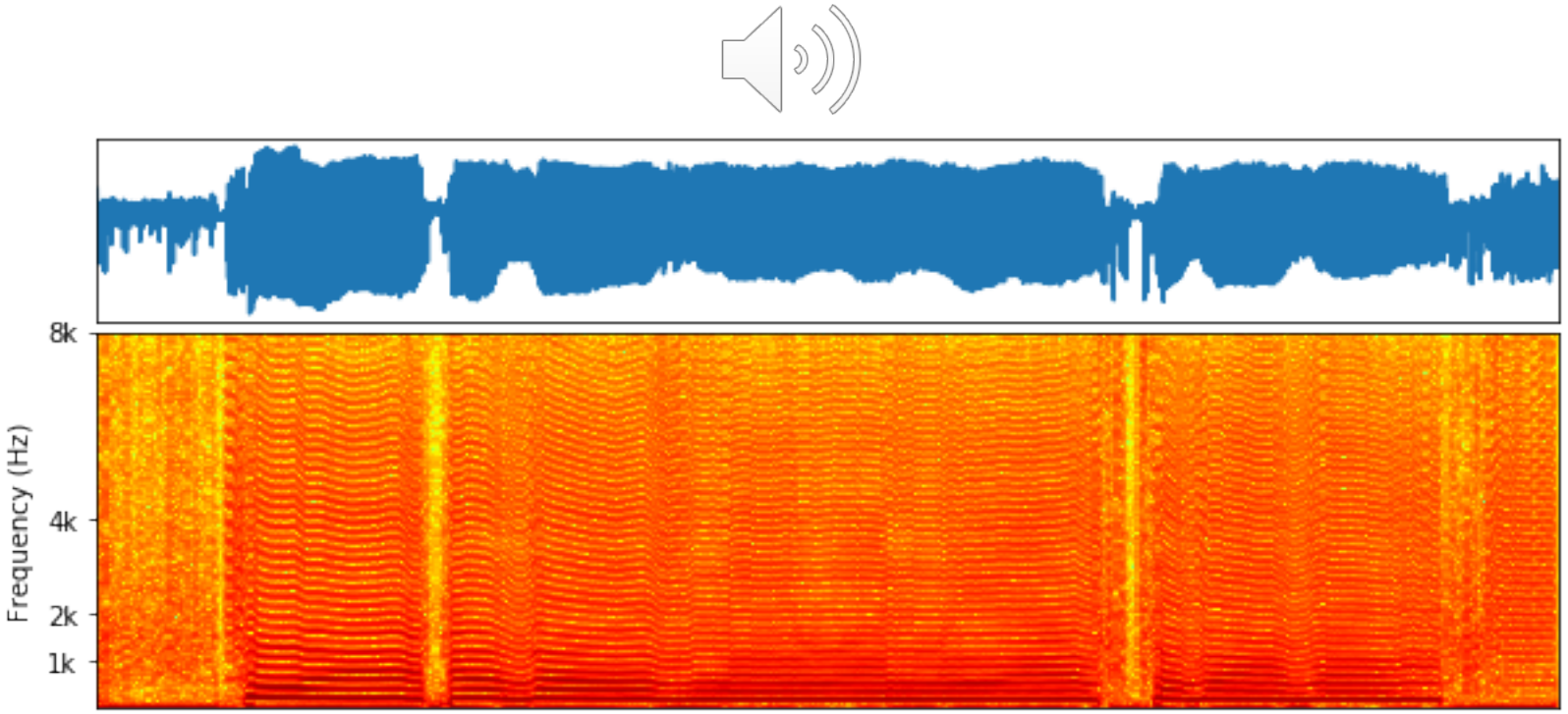




EXPERIMENTS

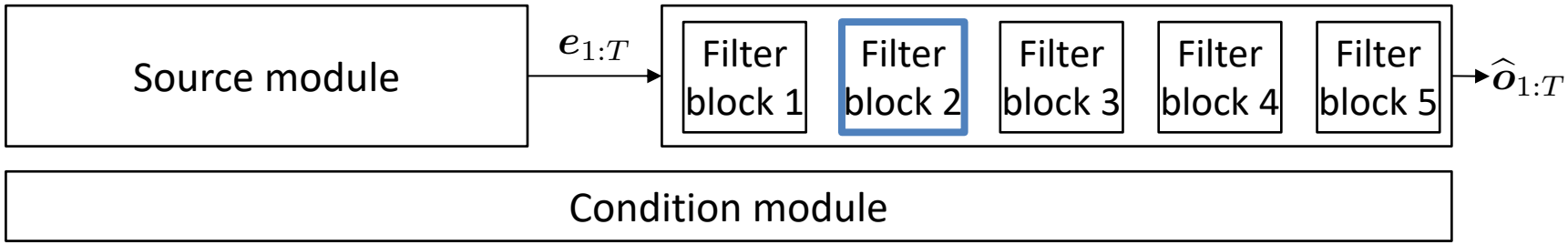
Analysis

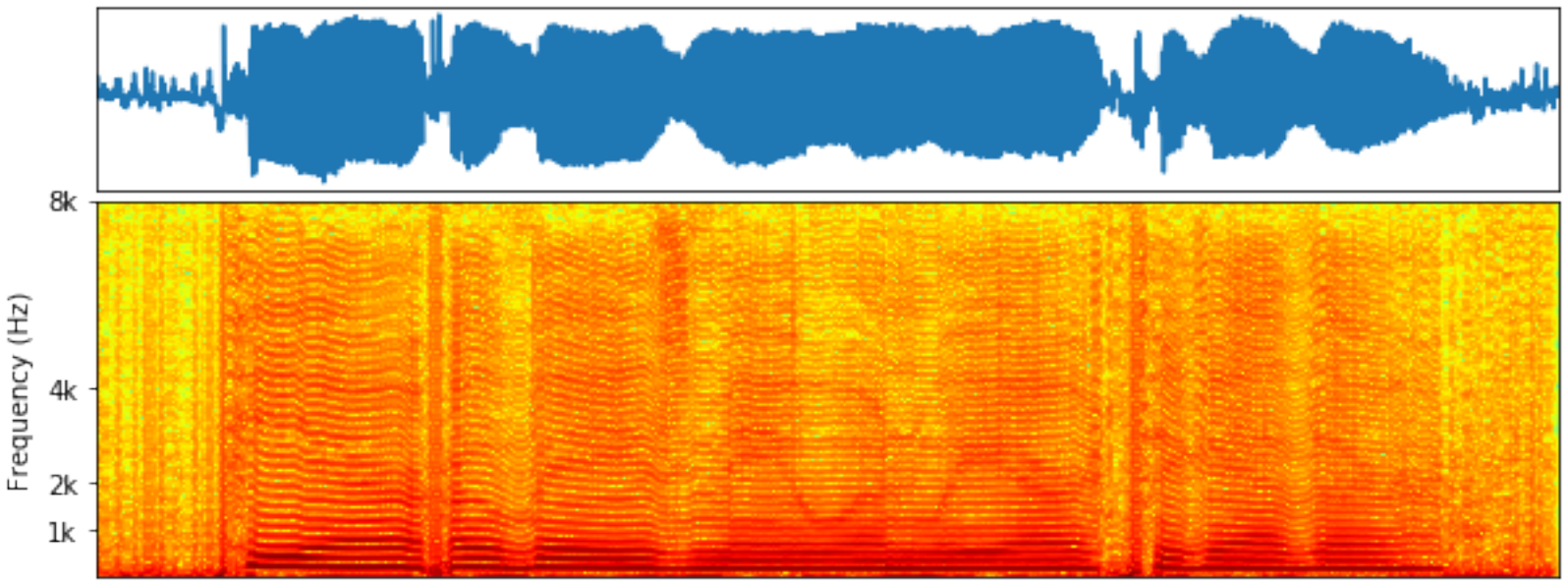




EXPERIMENTS

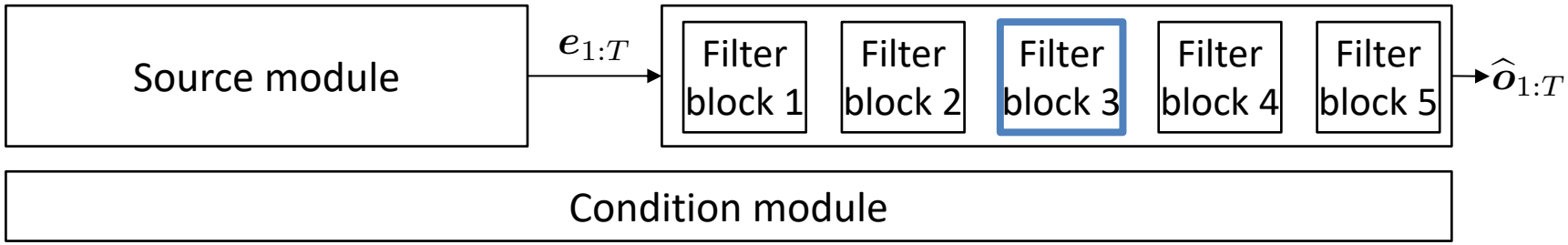
Analysis

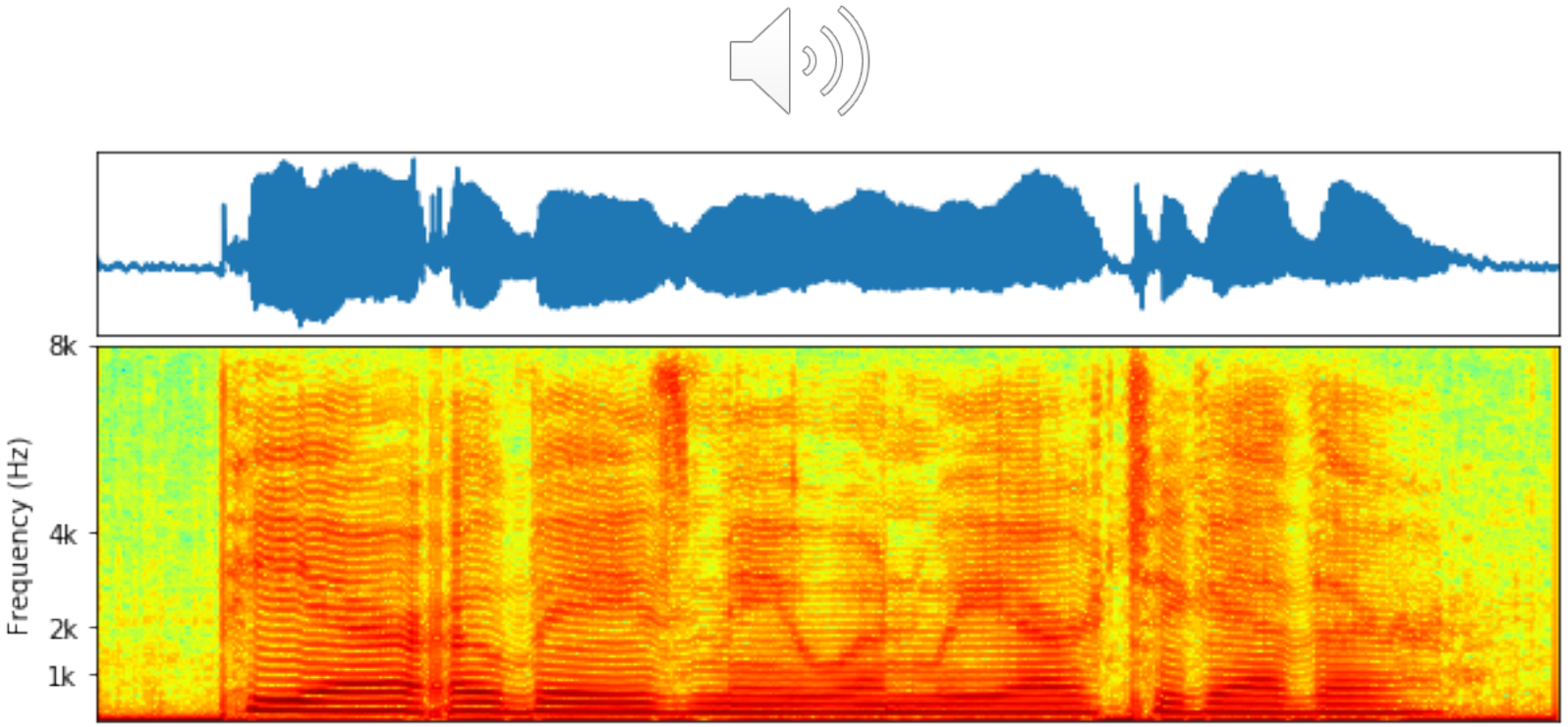




EXPERIMENTS

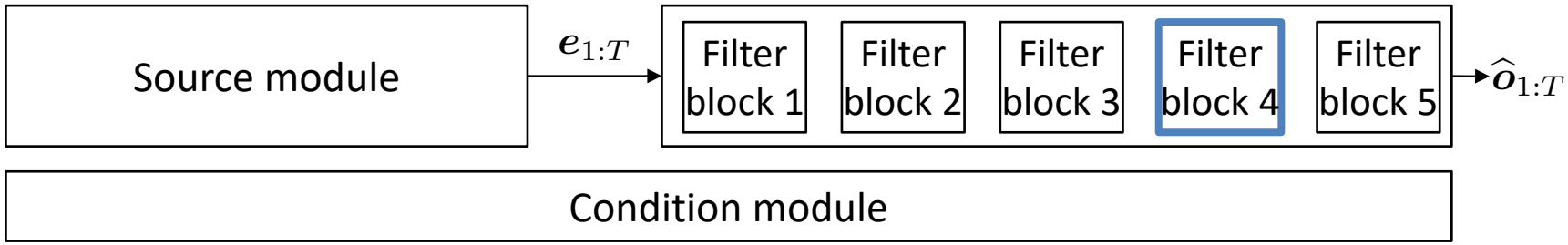
Analysis

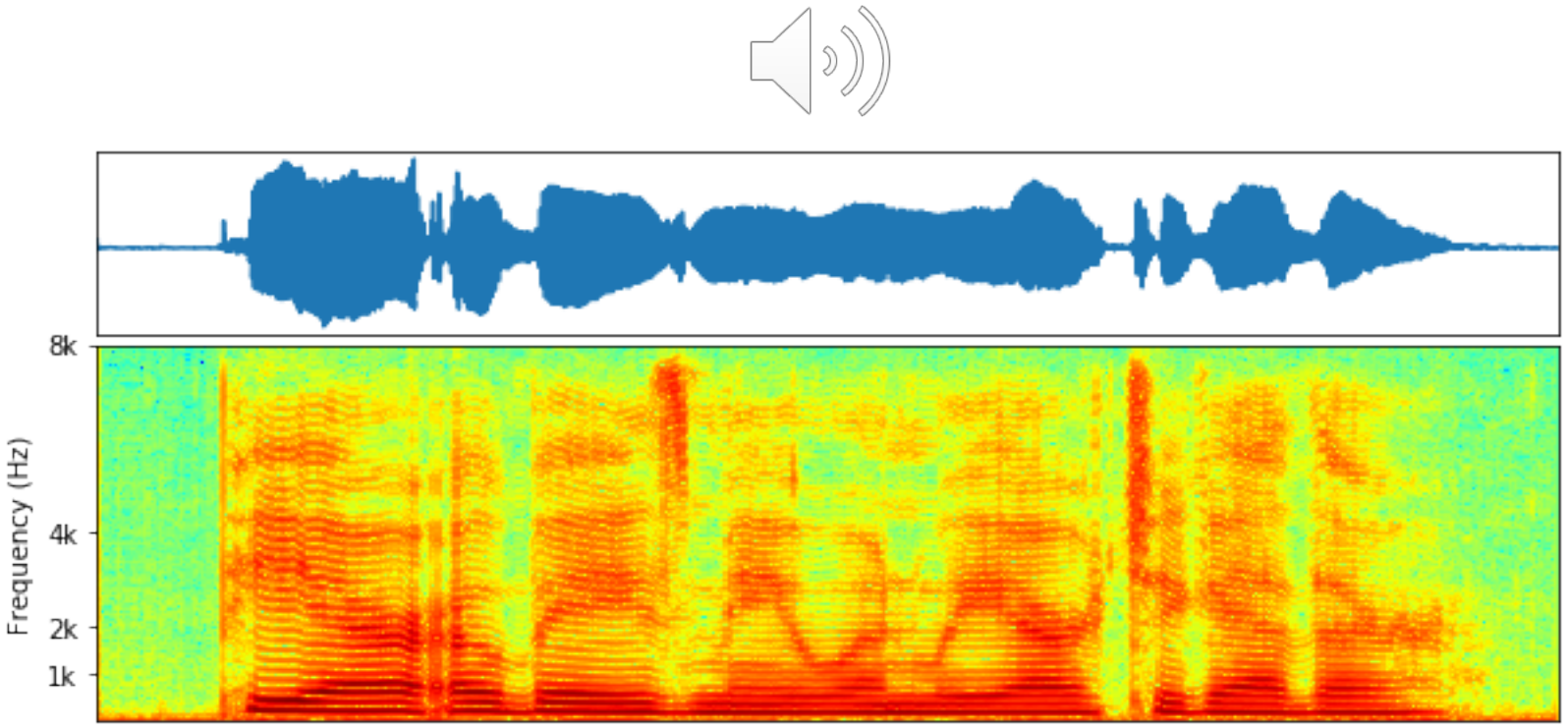




EXPERIMENTS

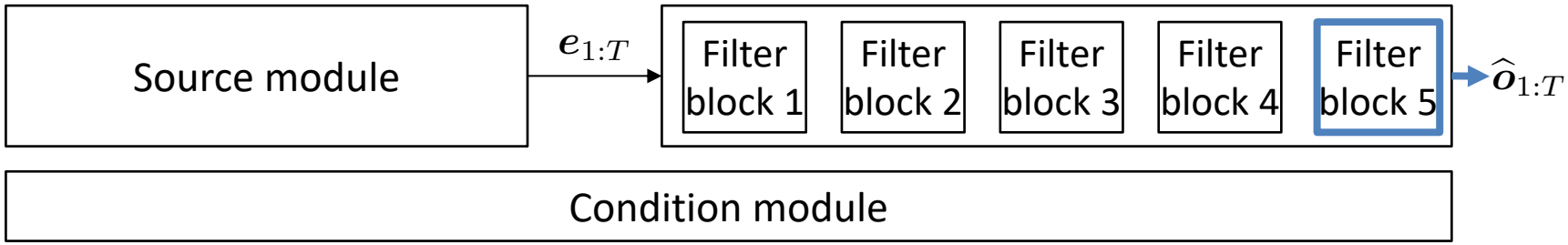
Analysis

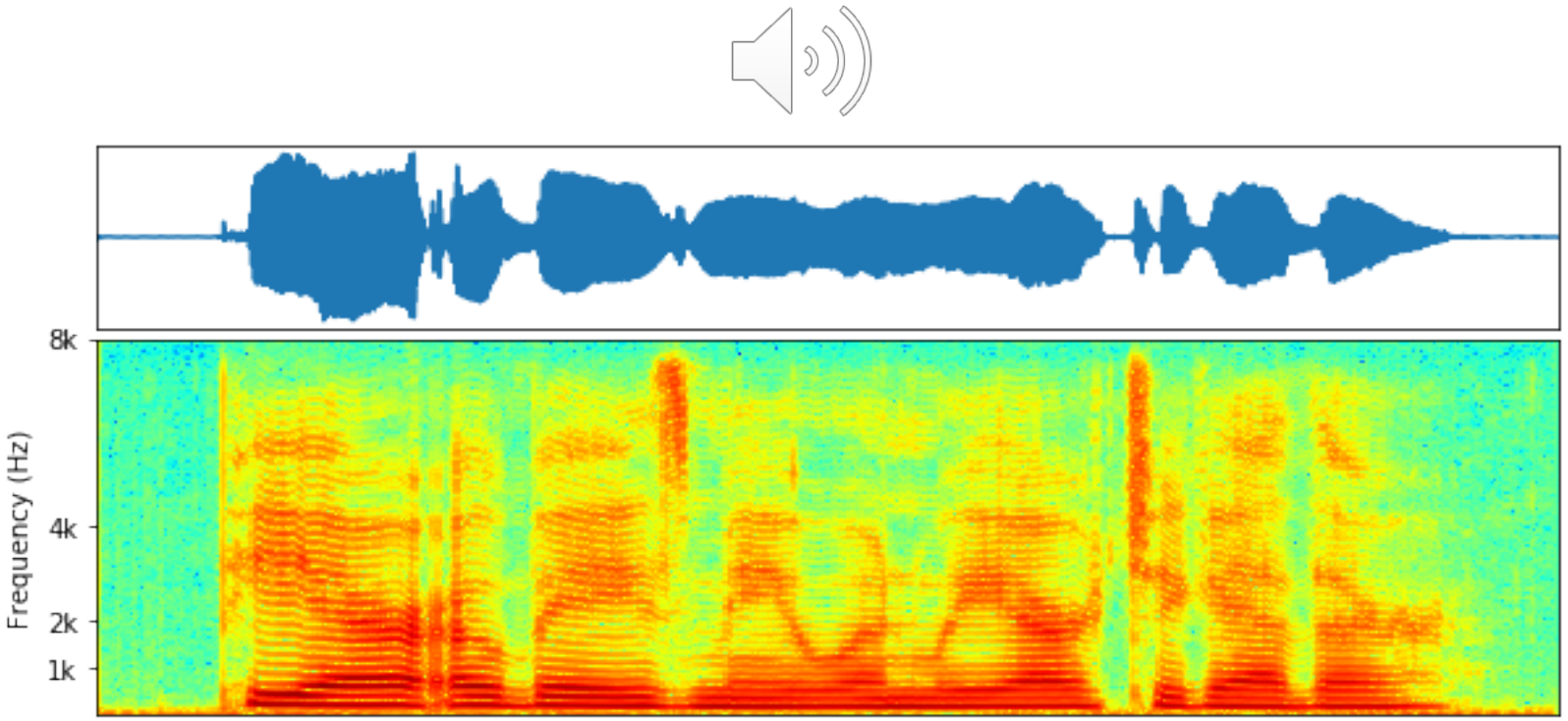




EXPERIMENTS

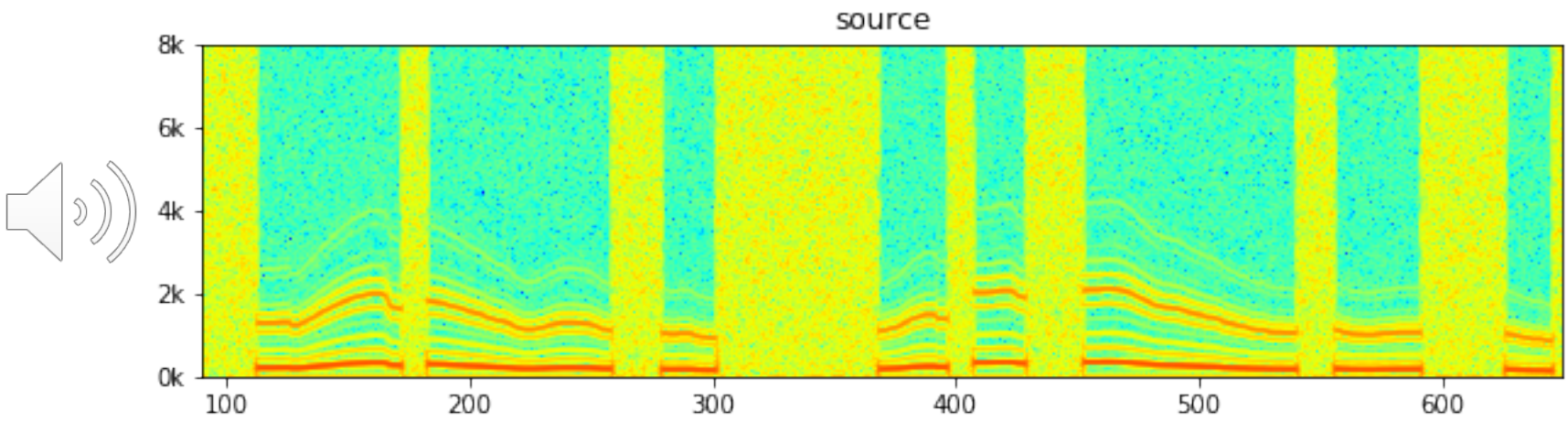
Analysis

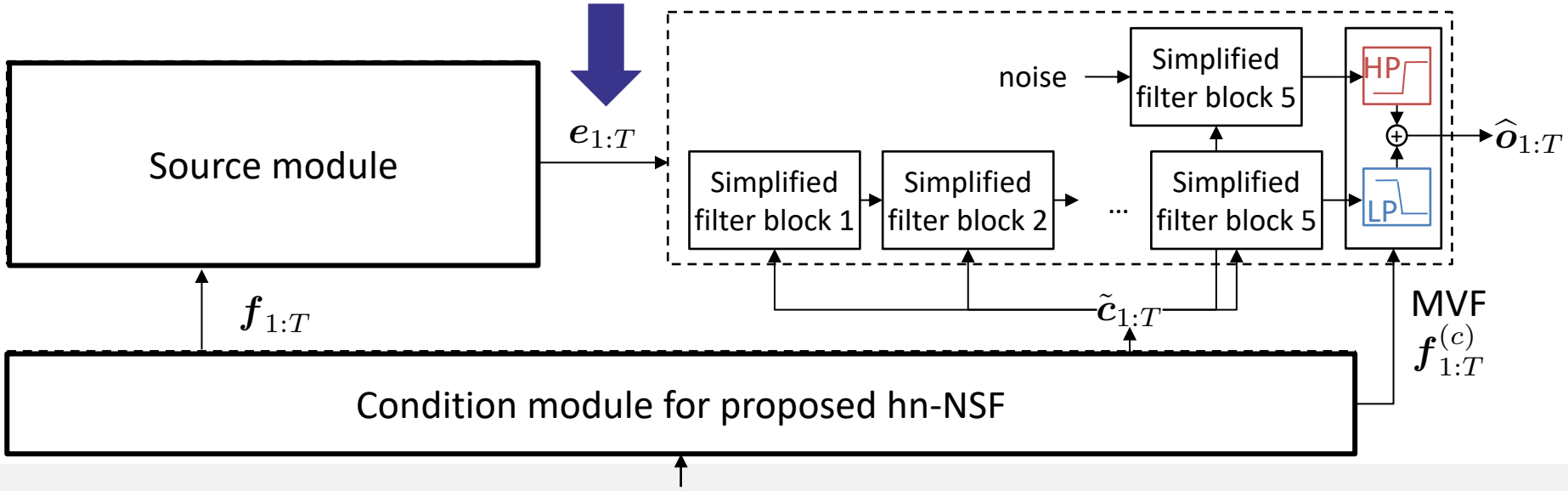




EXPERIMENTS

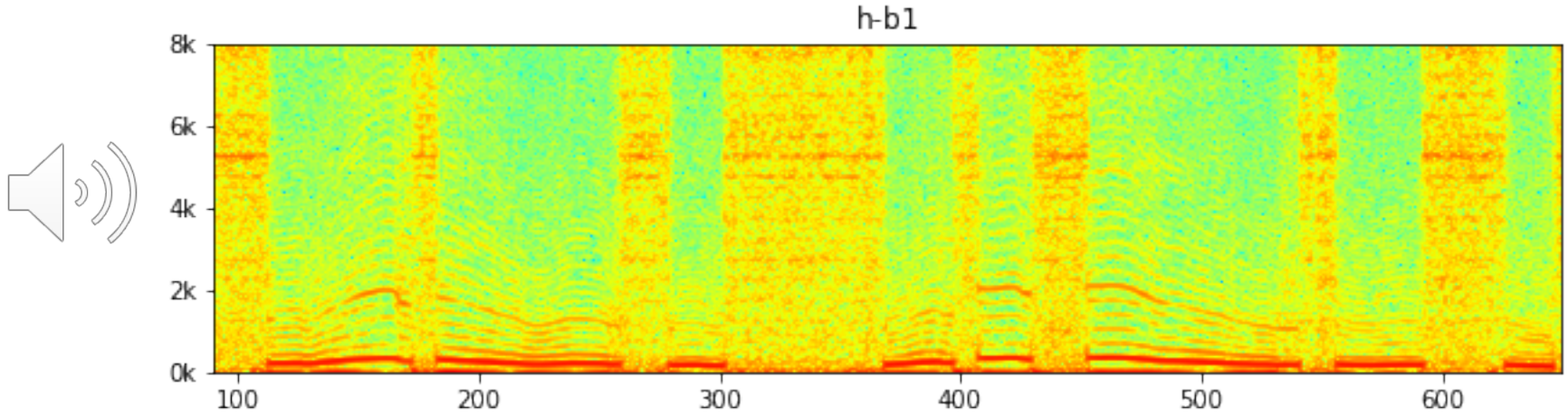
Waveform generation: step by step

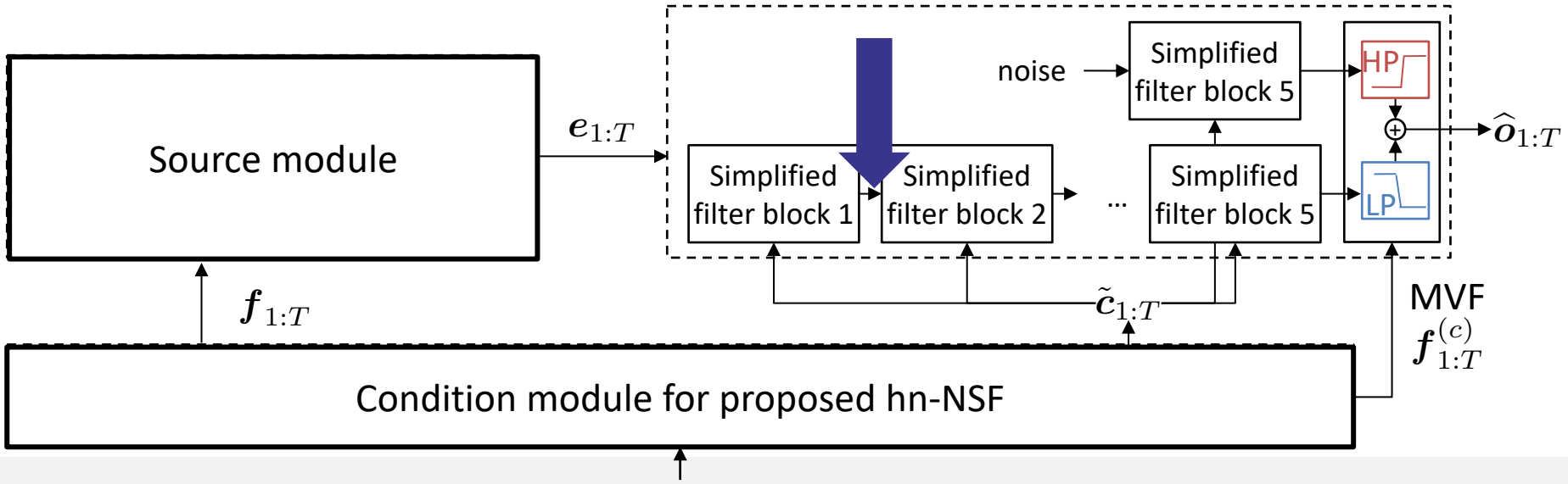




EXPERIMENTS

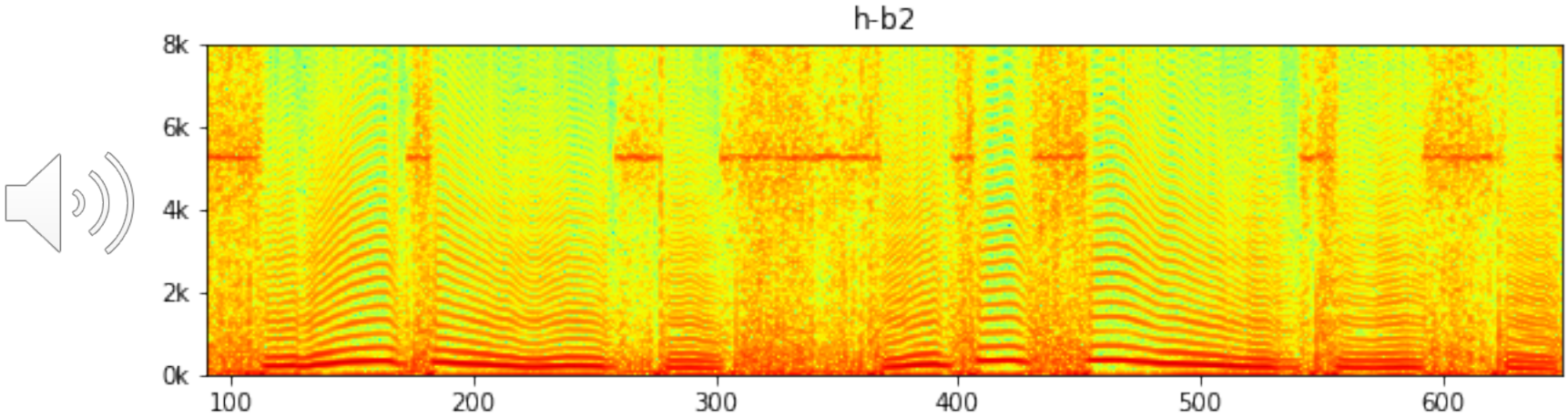
Waveform generation: step by step

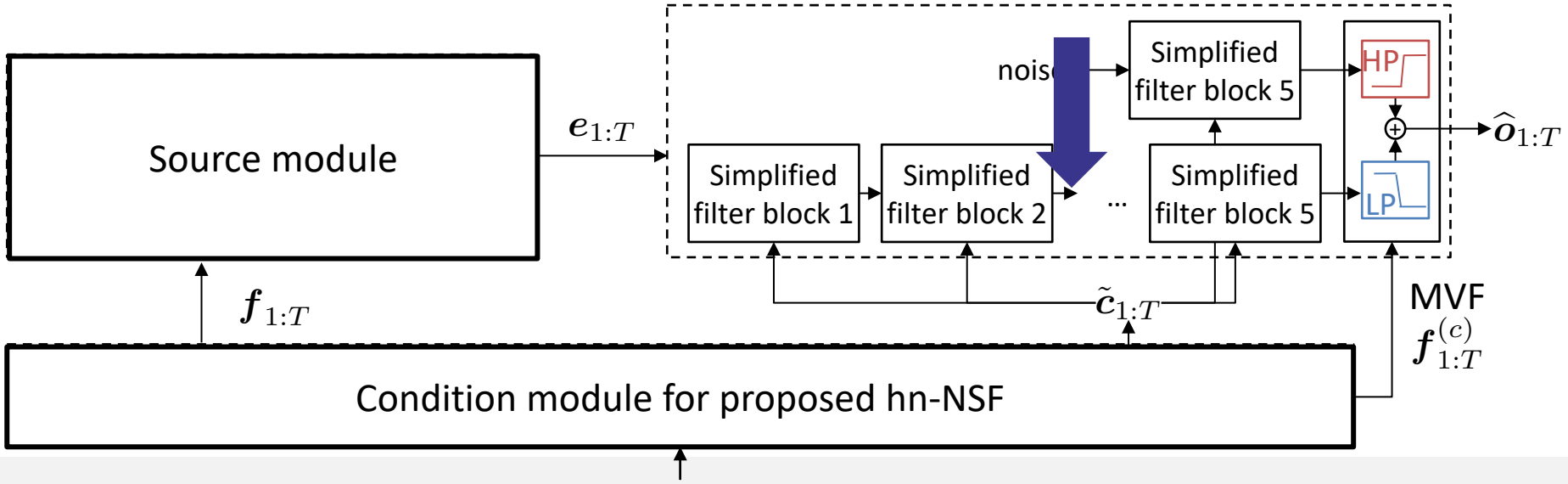




EXPERIMENTS

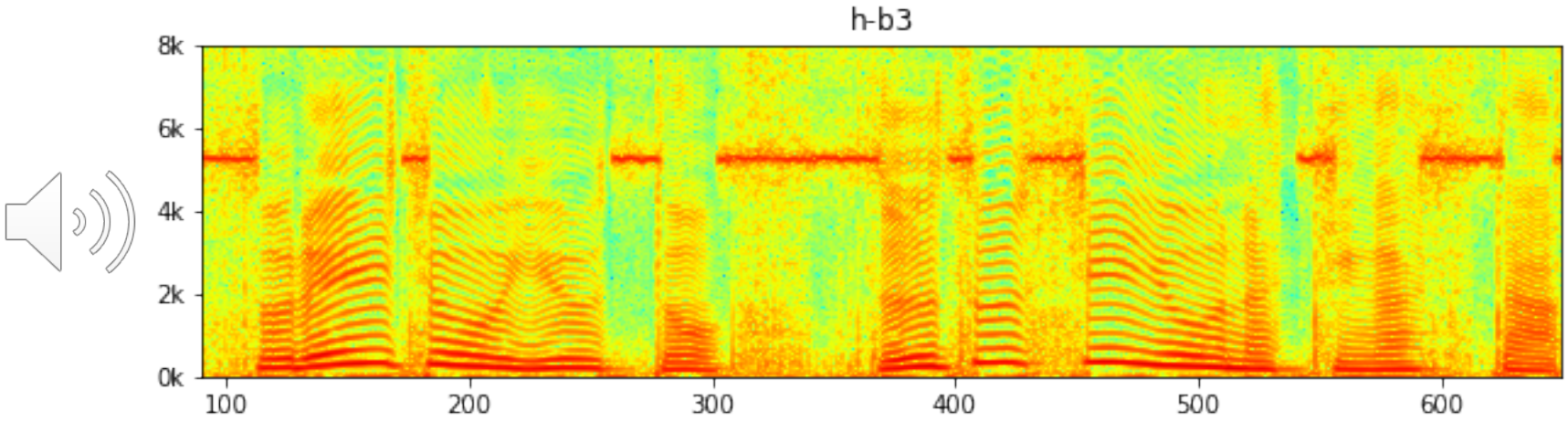
Waveform generation: step by step

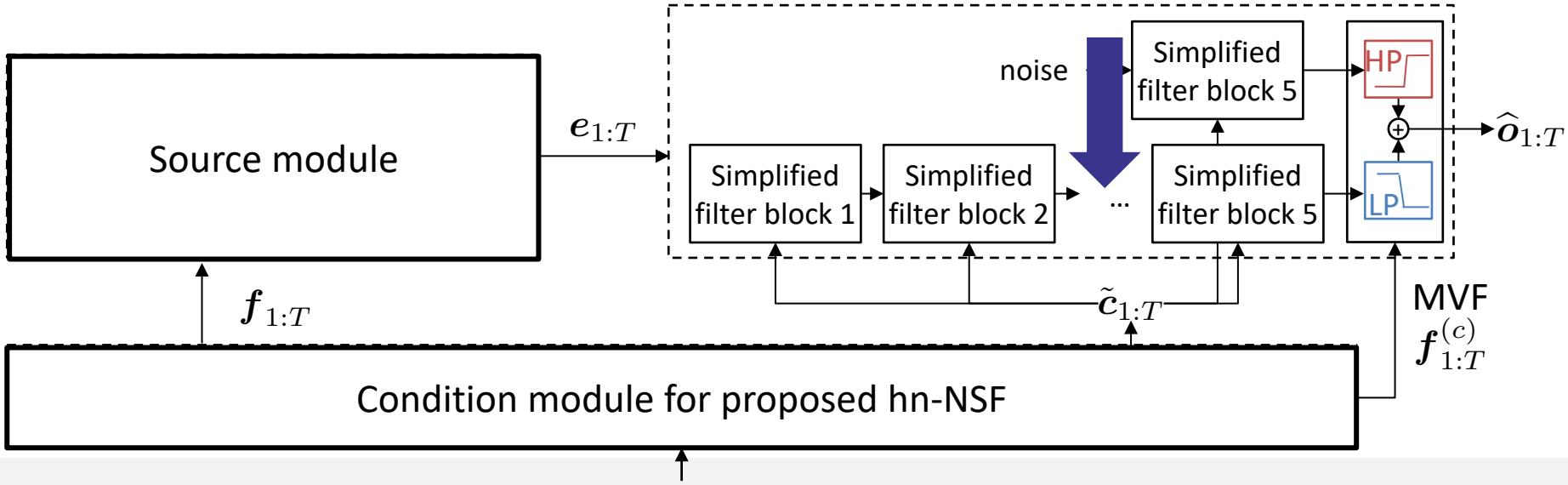




EXPERIMENTS

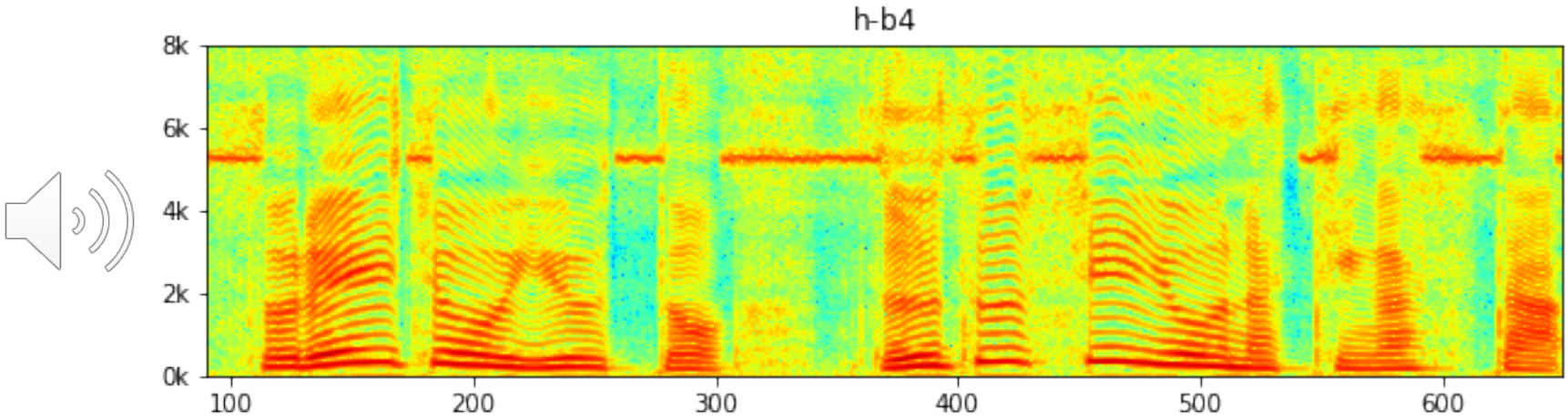
Waveform generation: step by step

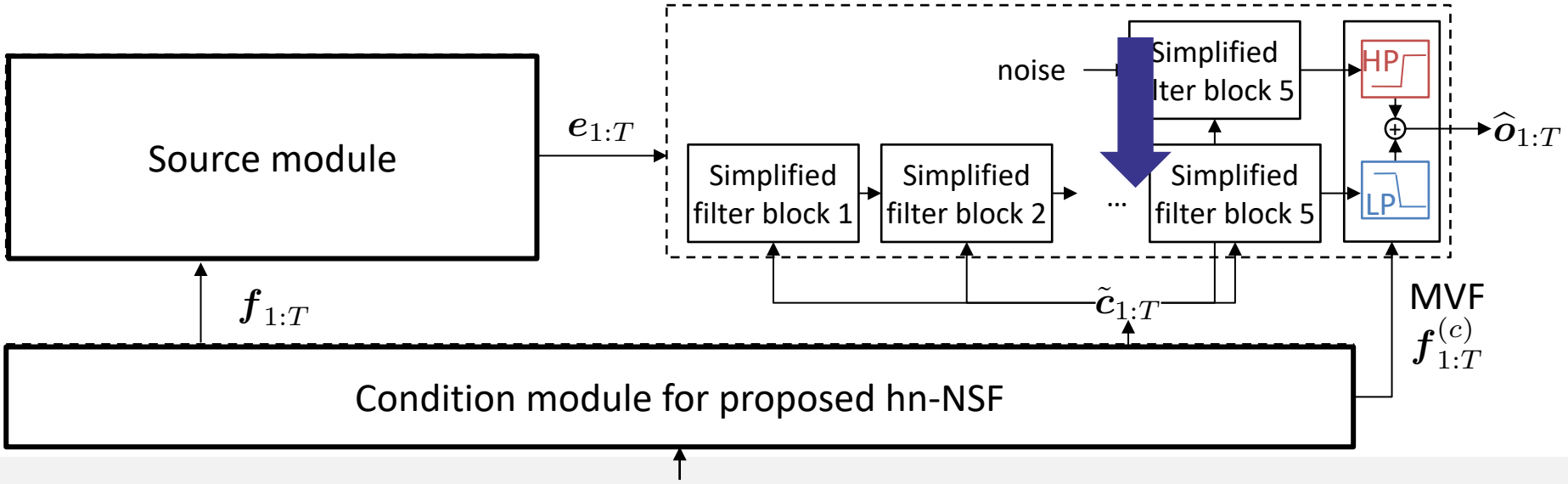




EXPERIMENTS

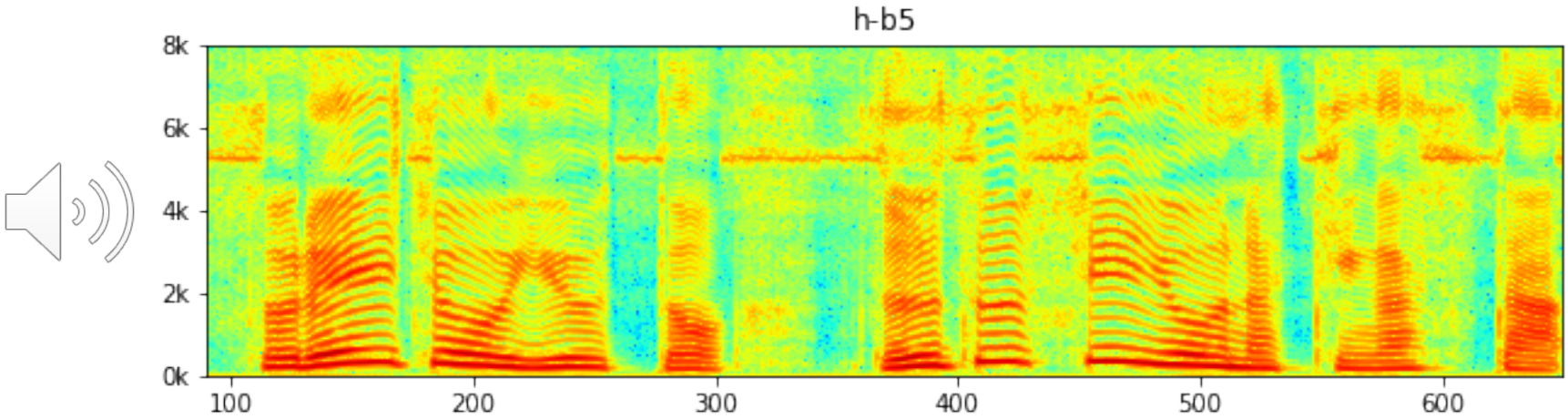
Waveform generation: step by step

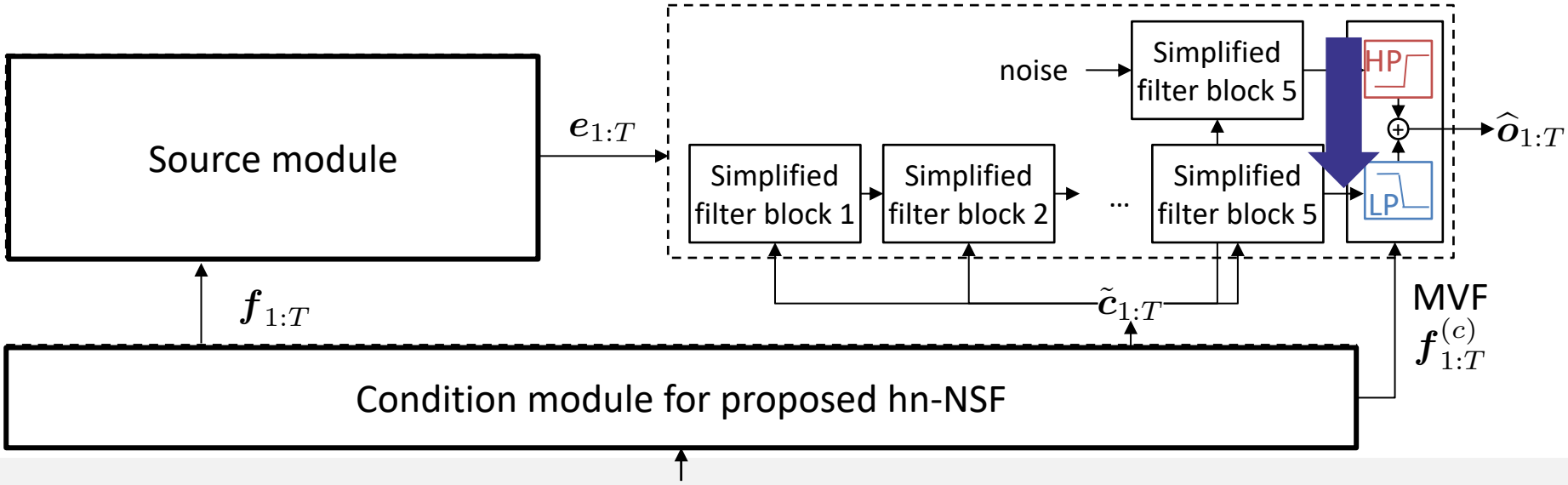




EXPERIMENTS

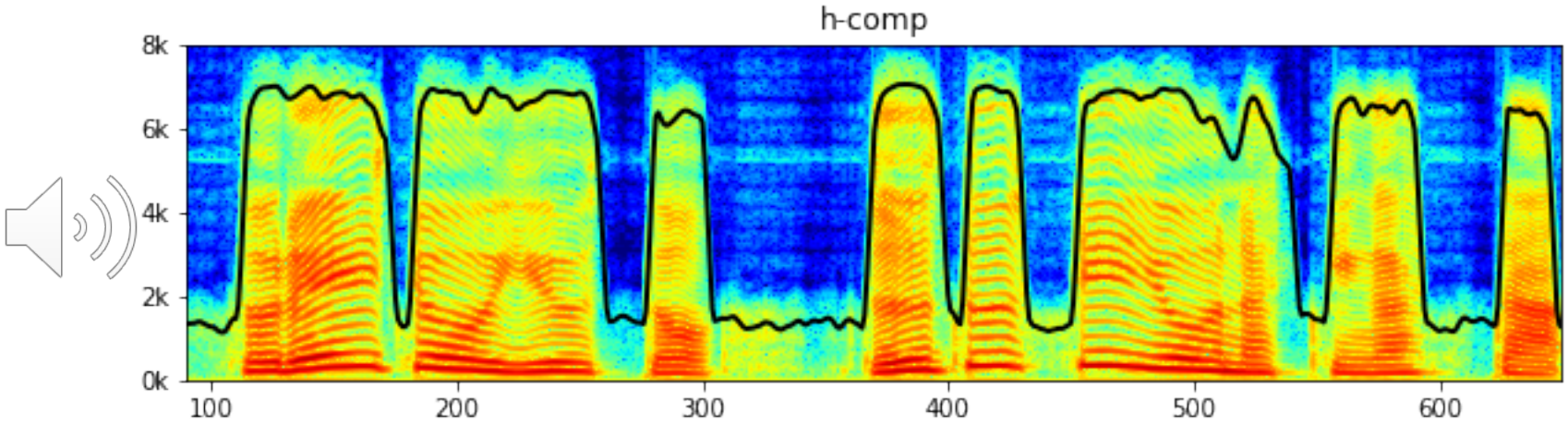
Waveform generation: step by step

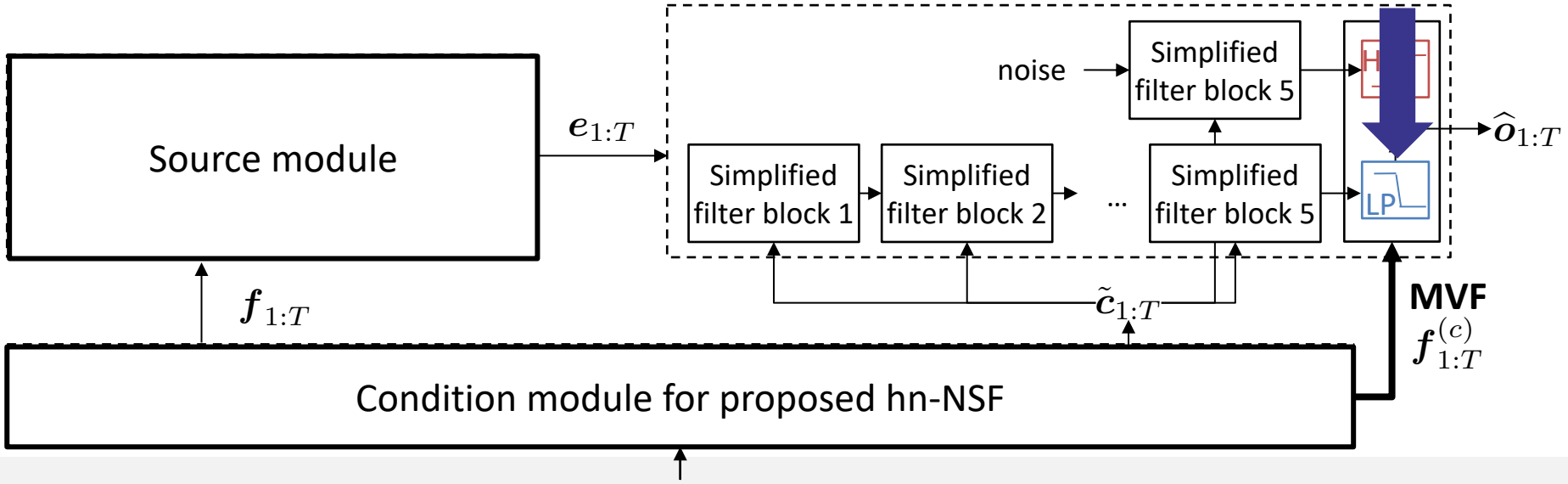




EXPERIMENTS

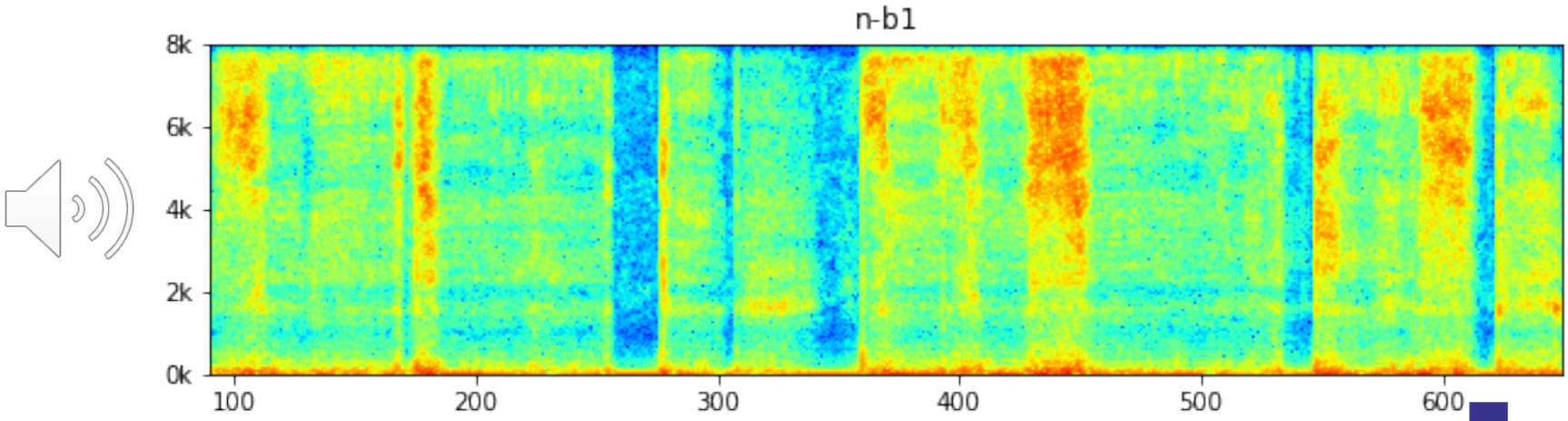
Waveform generation: step by step

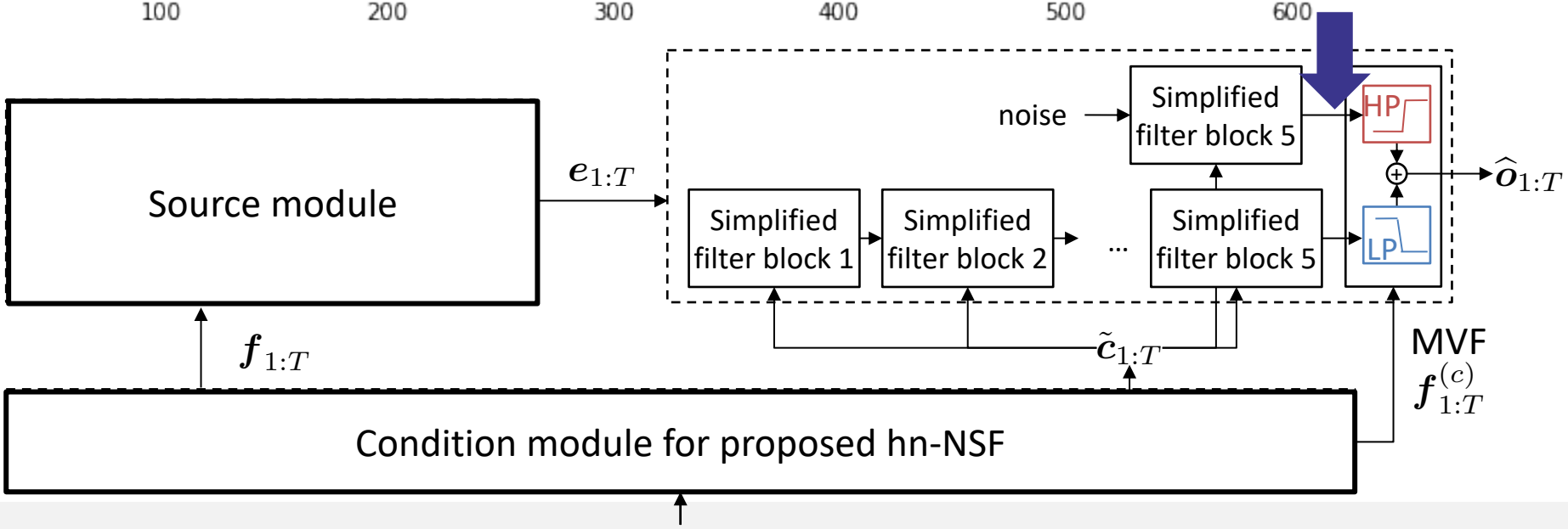




EXPERIMENTS

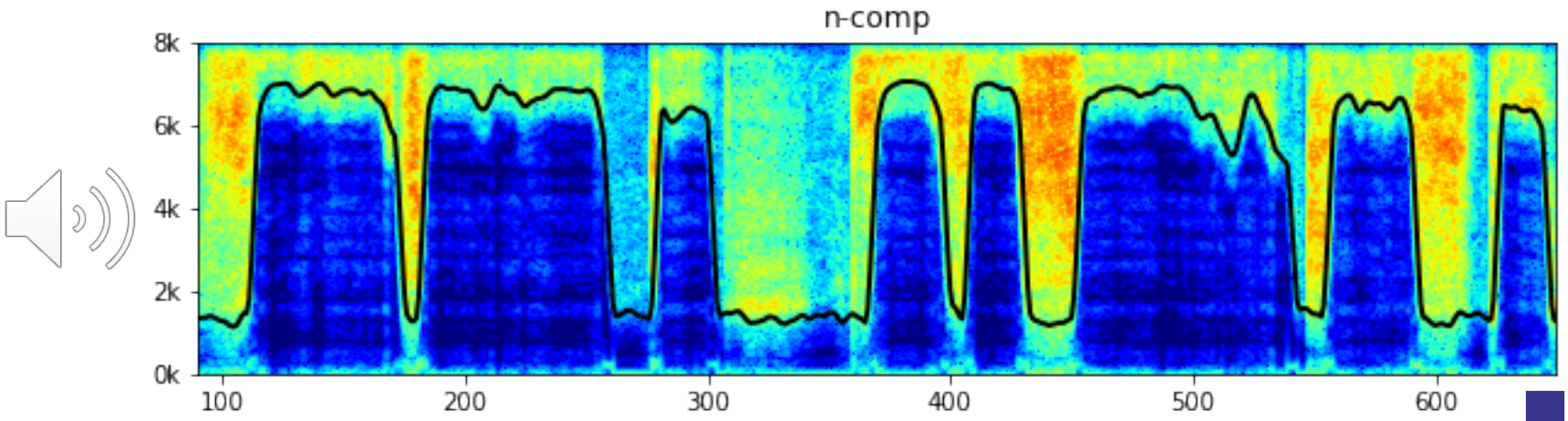
Waveform generation: step by step

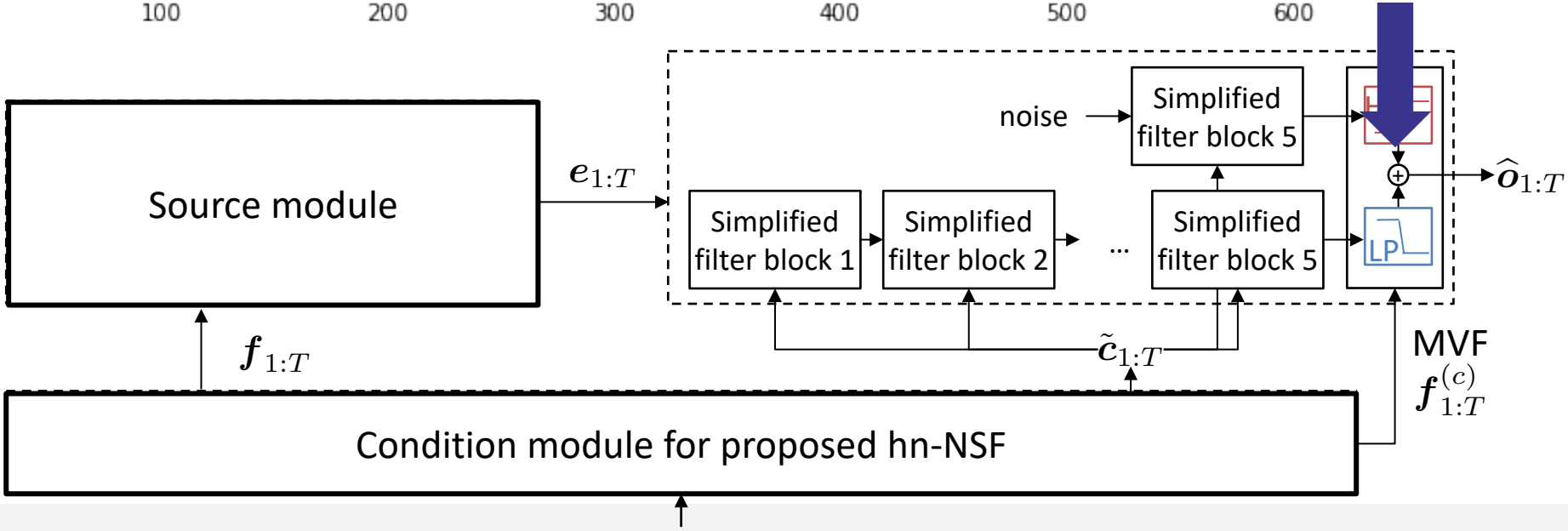




EXPERIMENTS

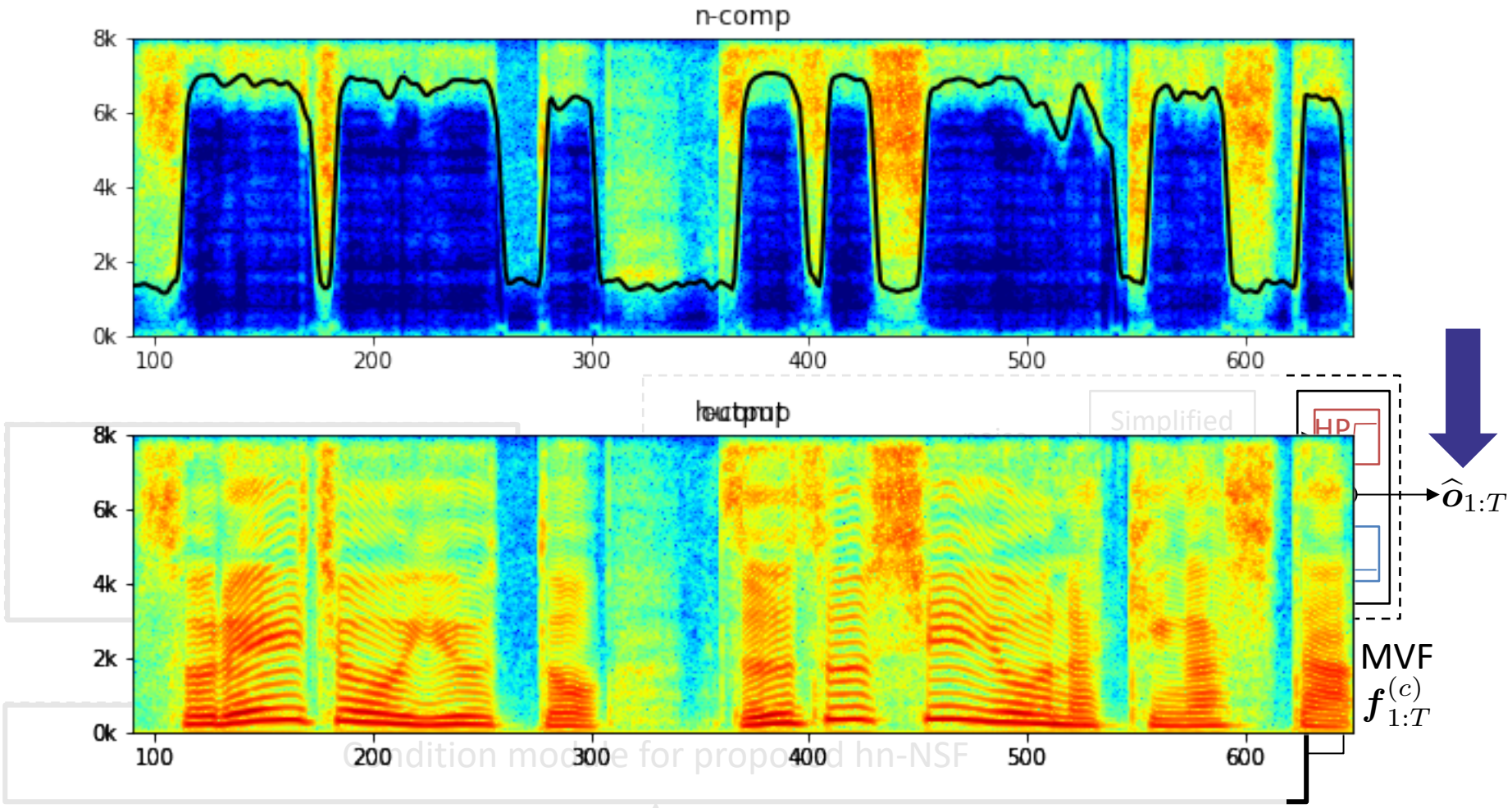
Waveform generation: step by step





EXPERIMENTS

Waveform generation: step by step



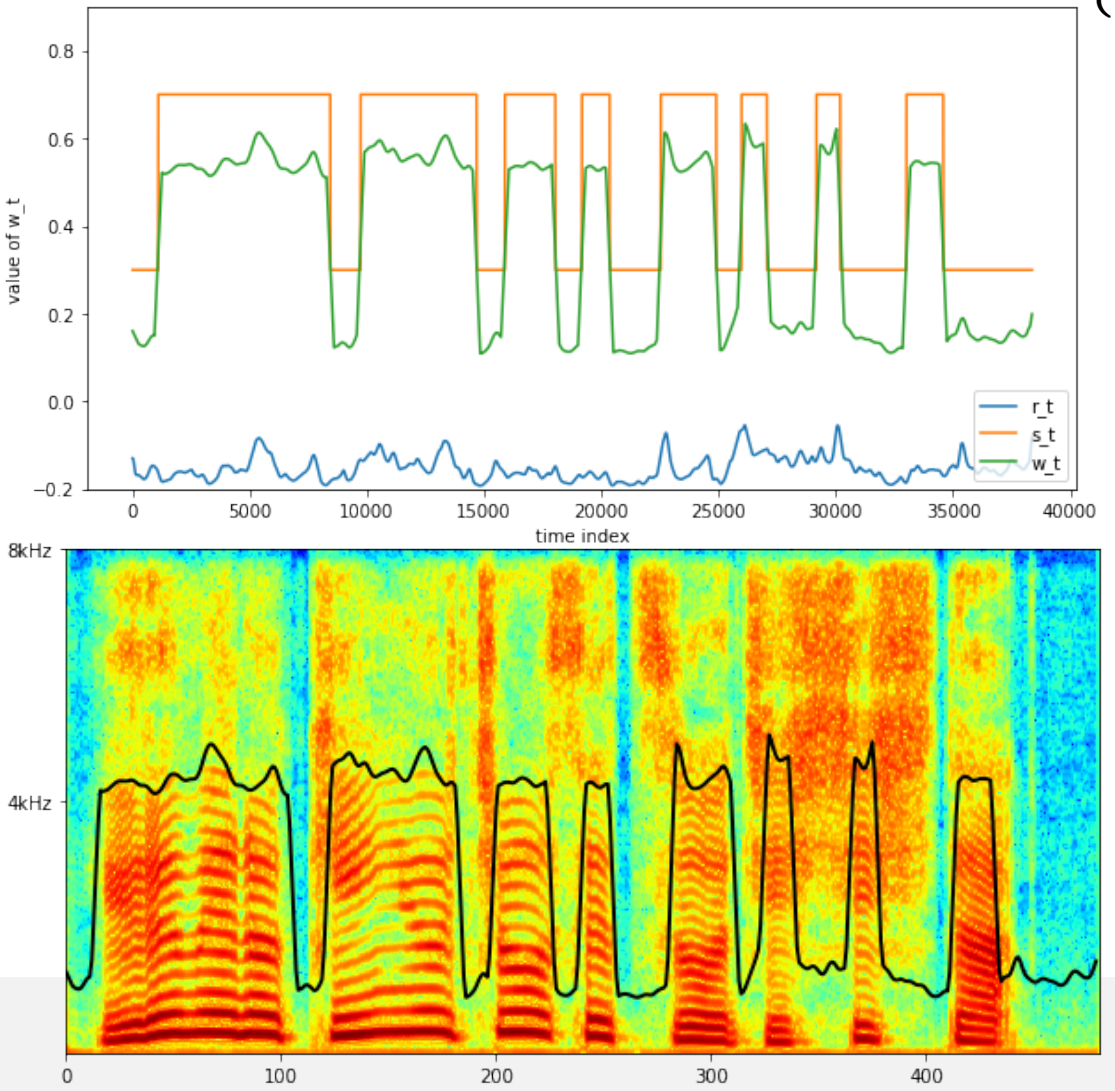
SINC-BASED H-NSF

System 1

- W_t is well trained

$$w_t = v_t + 0.2 \cdot r_t \qquad w_t \in \begin{cases} (0.5, 0.9), \text{voiced} \\ (0.1, 0.5), \text{unvoiced} \end{cases}$$

Epoch 001



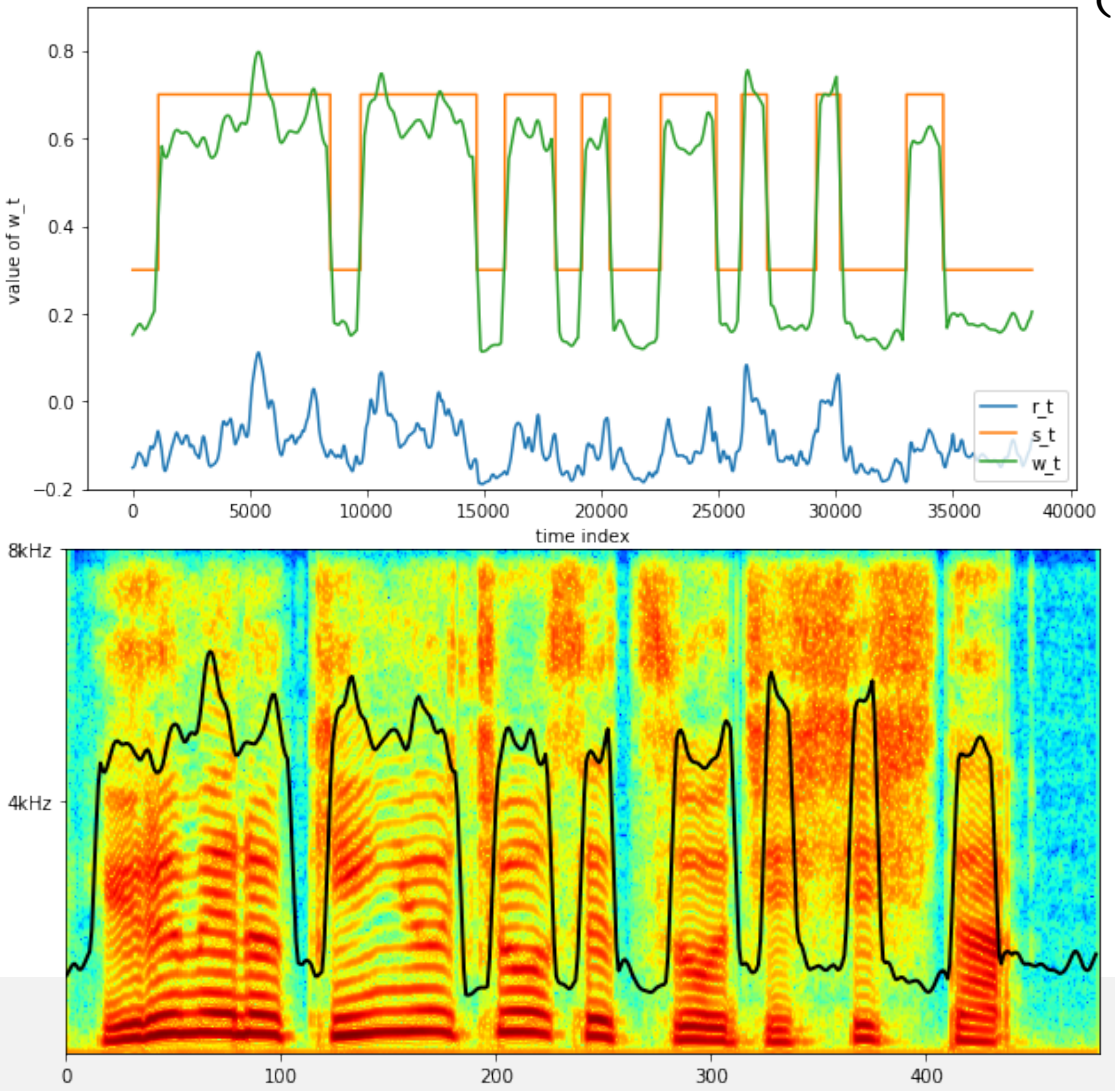
SINC-BASED H-NSF

System 1

- W_t is well trained

$$w_t = v_t + 0.2 \cdot r_t \qquad w_t \in \begin{cases} (0.5, 0.9), \text{voiced} \\ (0.1, 0.5), \text{unvoiced} \end{cases}$$

Epoch 010



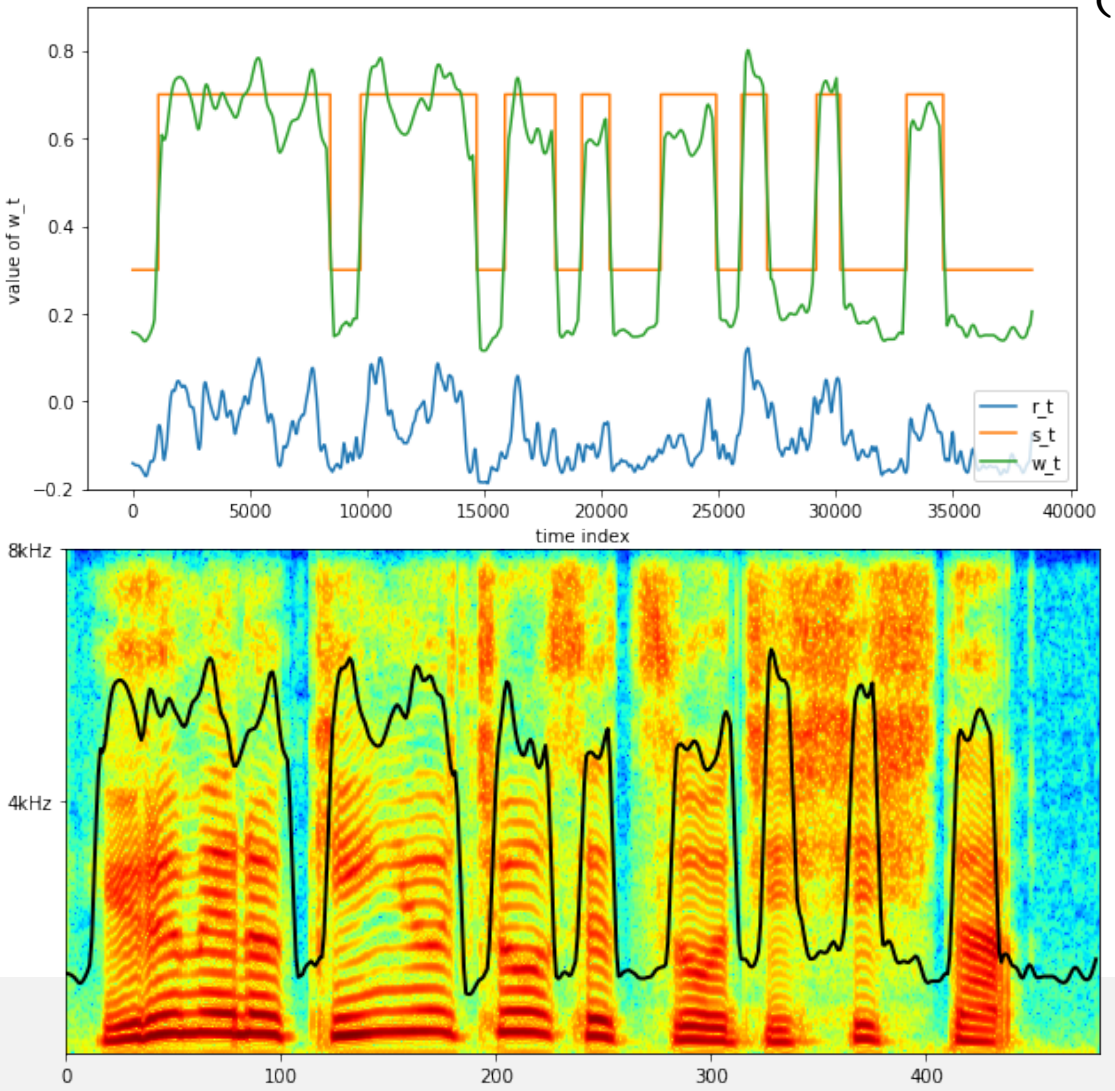
SINC-BASED H-NSF

System 1

- W_t is well trained

$$w_t = v_t + 0.2 \cdot r_t \qquad w_t \in \begin{cases} (0.5, 0.9), \text{voiced} \\ (0.1, 0.5), \text{unvoiced} \end{cases}$$

Epoch 020



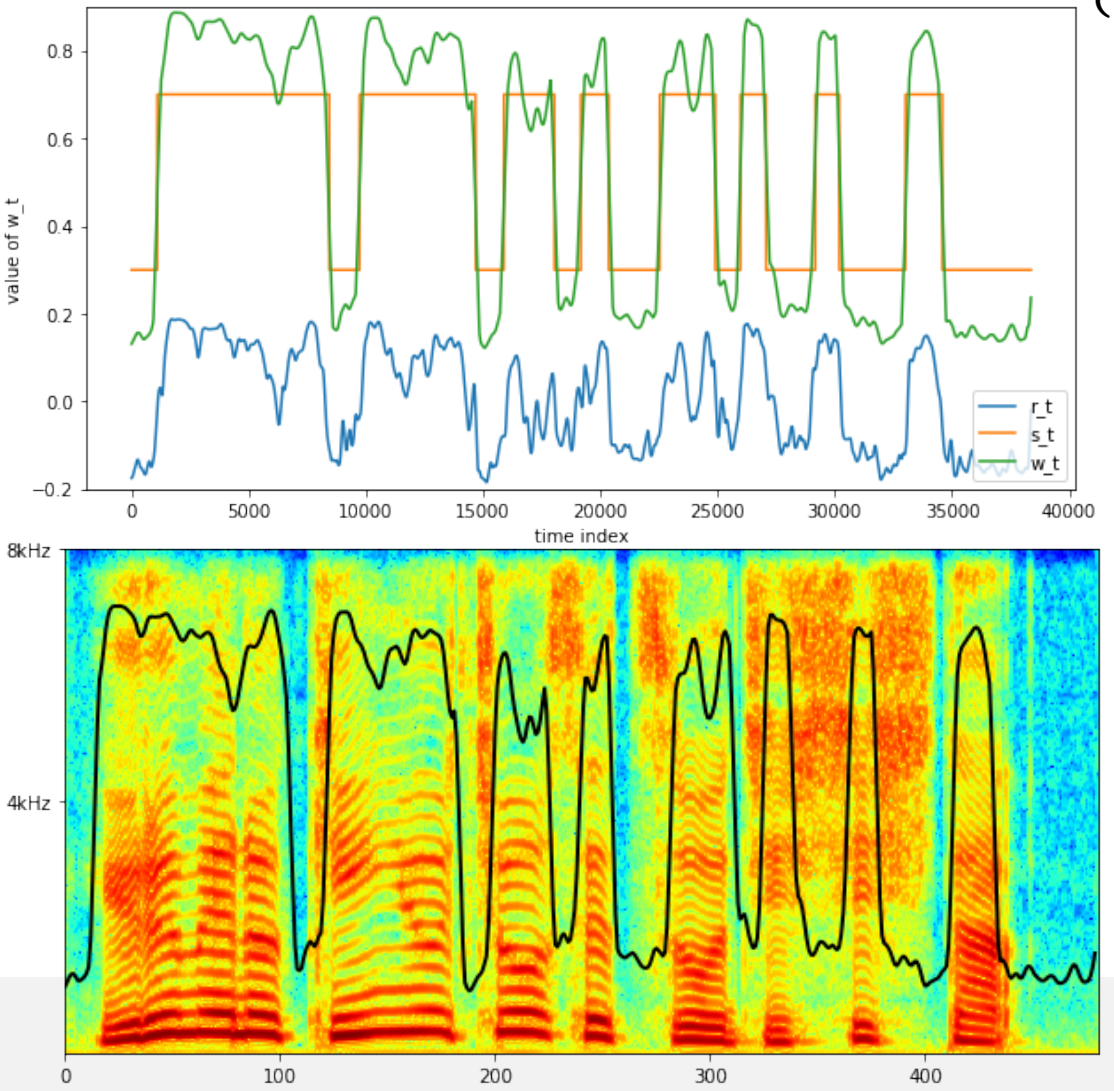
SINC-BASED H-NSF

System 1

- W_t is well trained

$$w_t = v_t + 0.2 \cdot r_t \qquad w_t \in \begin{cases} (0.5, 0.9), \text{voiced} \\ (0.1, 0.5), \text{unvoiced} \end{cases}$$

Epoch 030



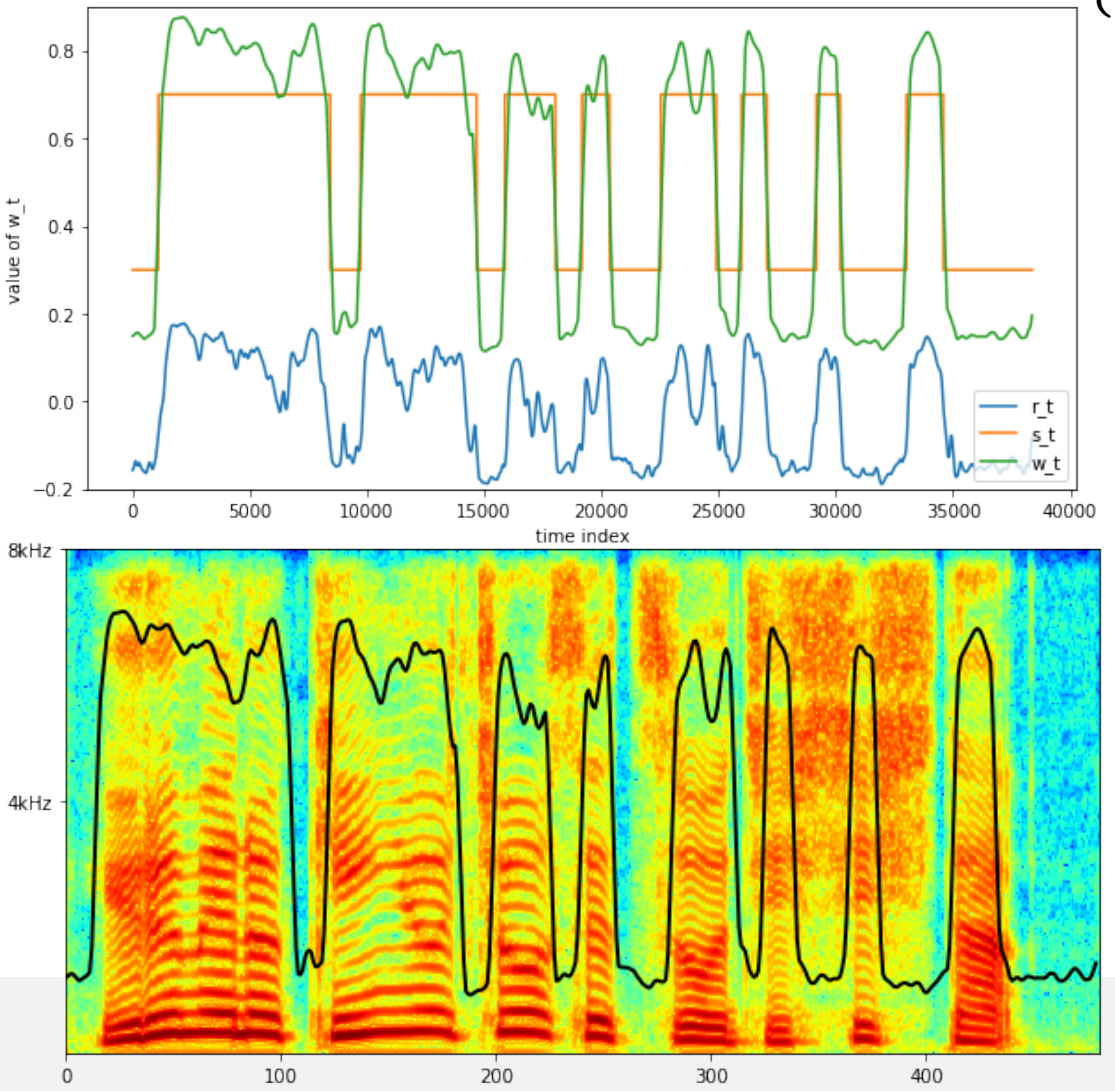
SINC-BASED H-NSF

System 1

- W_t is well trained

$$w_t = v_t + 0.2 \cdot r_t \qquad w_t \in \begin{cases} (0.5, 0.9), \text{voiced} \\ (0.1, 0.5), \text{unvoiced} \end{cases}$$

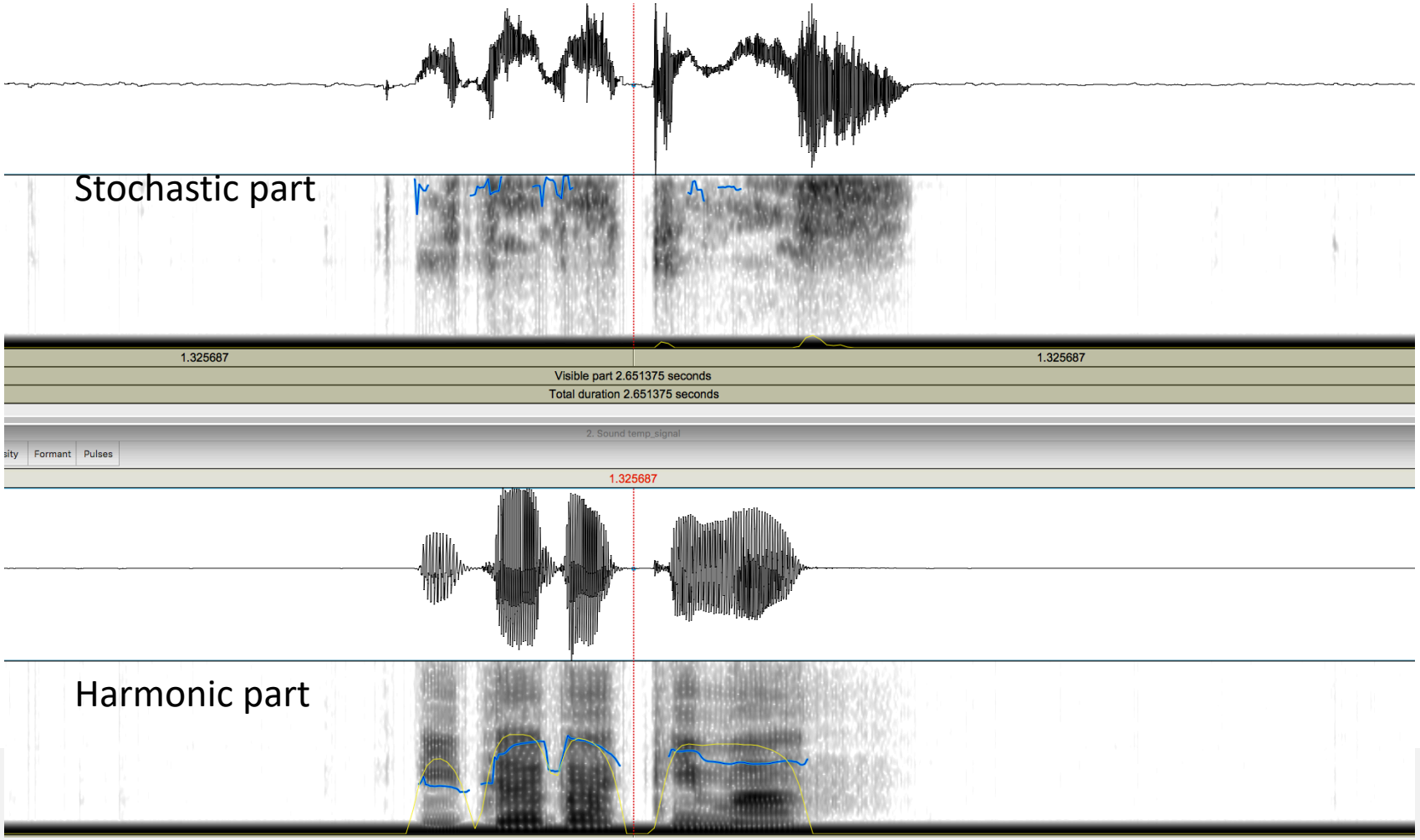
Epoch last



Waveform Modeling

Harmonic plus stochastic

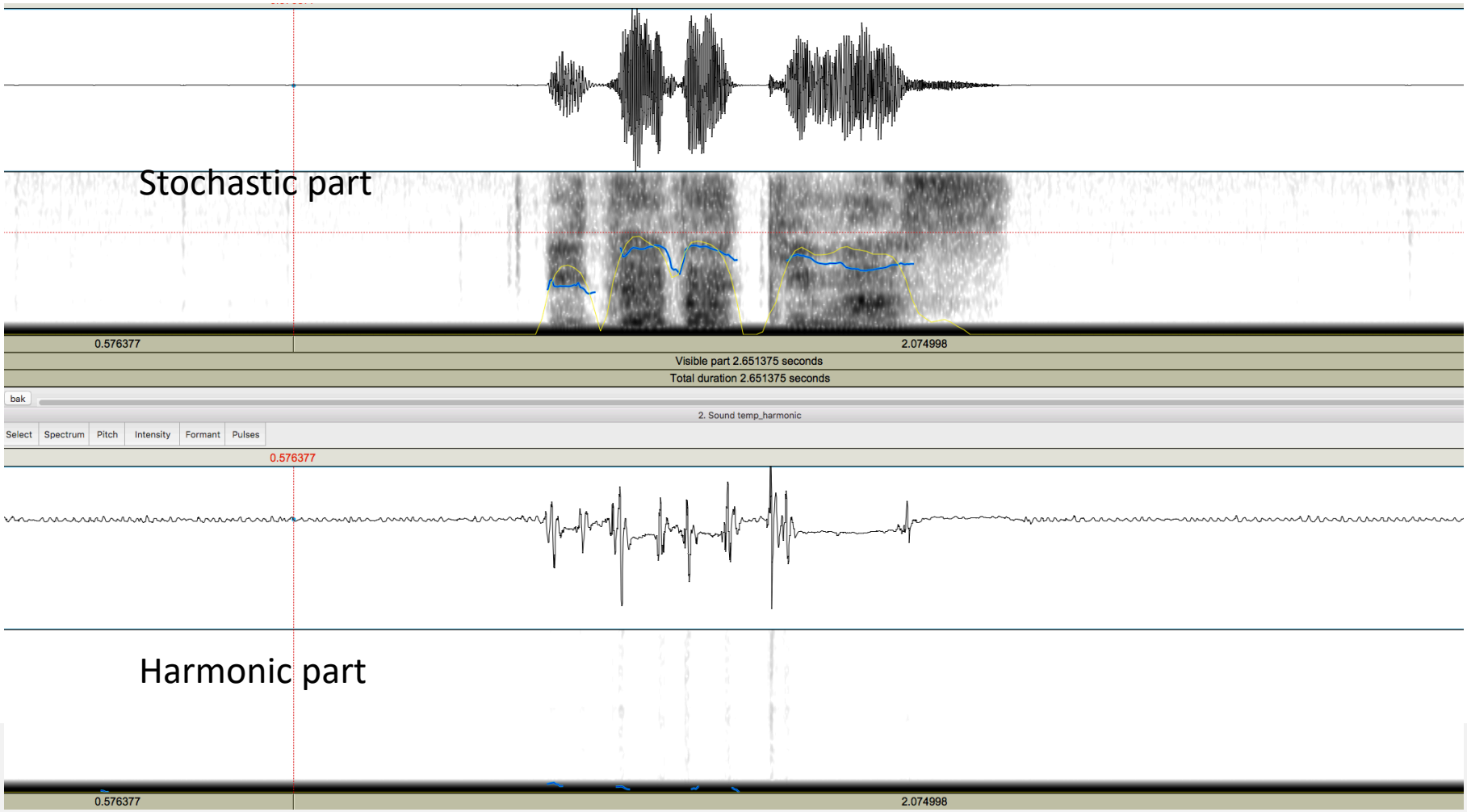
- Separate harmonics and noise excitation
  - Successful training:

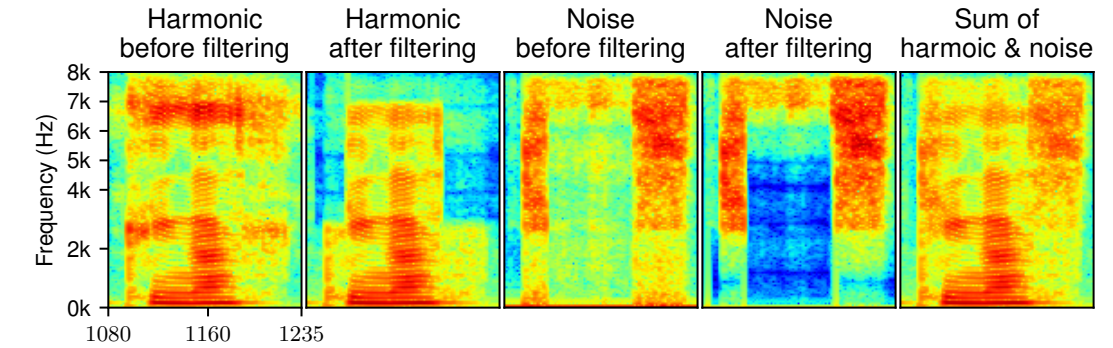


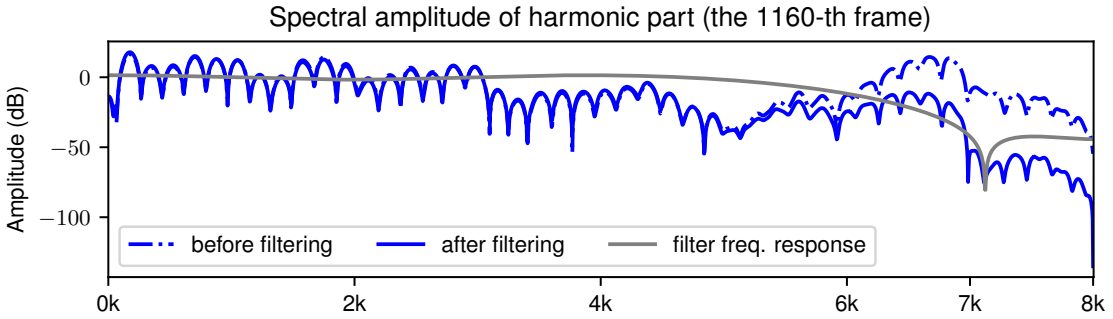
Waveform Modeling

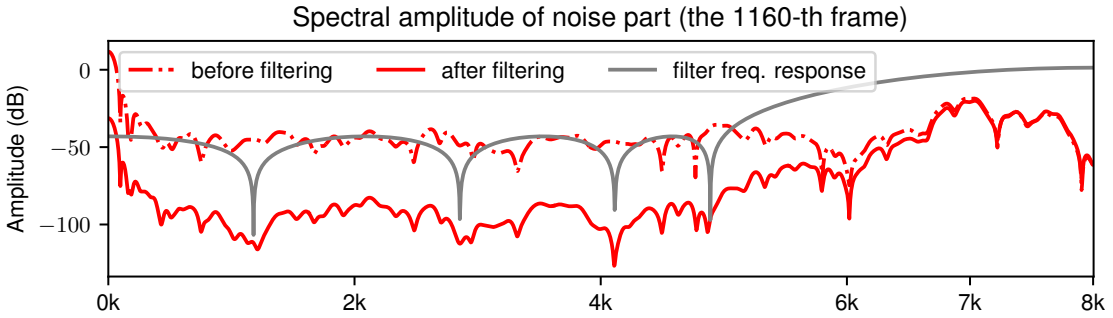
Harmonic plus stochastic

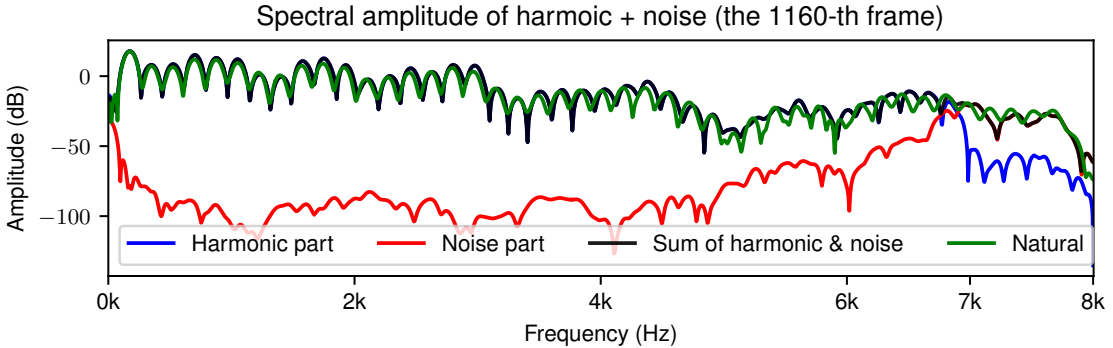
- Separate harmonics and noise excitation
  - Bad training: stochastic part plays the main role




Harmonic before filtering
Harmonic after filtering
Noise before filtering
Noise after filtering
Sum of harmoic & noise
Frequency (Hz)
8k
7k
6k
5k
4k
2k
0k
1080
1160
1235


Spectral amplitude of harmonic part (the 1160-th frame)
Amplitude (dB)
0
−50
−100
before filtering
after filtering
filter freq. response
0k
2k
4k
6k
7k
8k

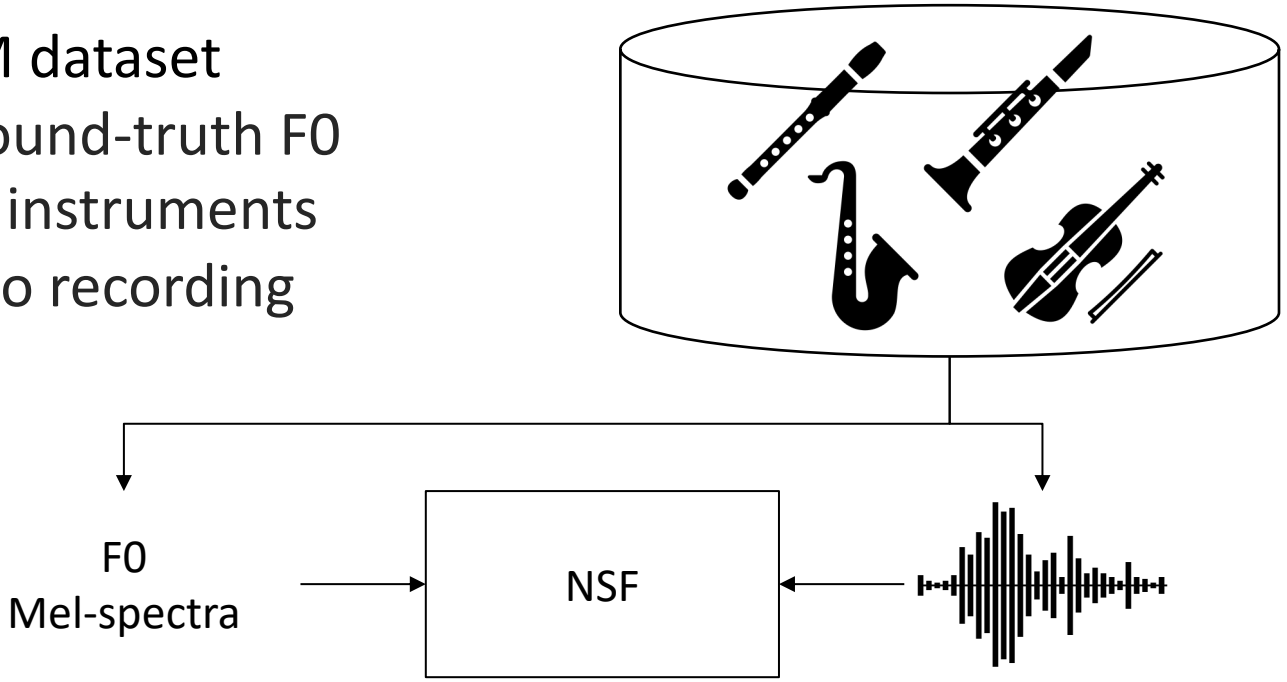

Spectral amplitude of noise part (the 1160-th frame)
Amplitude (dB)
0
−50
−100
before filtering
after filtering
filter freq. response
0k
2k
4k
6k
7k
8k


Spectral amplitude of harmoic + noise (the 1160-th frame)
Amplitude (dB)
0
−50
−100
Harmonic part
Noise part
Sum of harmonic & noise
Natural
0k
2k
4k
6k
7k
8k
Frequency (Hz)

BEYOND SPEECH

Music performance

- Training
  - URPM dataset
    - ground-truth F0
    - 13 instruments
    - solo recording



- One model for all instruments

http://www2.ece.rochester.edu/projects/air/projects/URMP.html

BEYOND SPEECH

Music performance

- Testing with natural Mel-spectra and F0 as input

	Natural	b-NSF	S-NSF	hn-NSF trainable MVF
Voilin				
Viola				
Oboe				
Trumpet				
Saxphone				

WaveNet

BEYOND SPEECH

Music performance

- Testing with natural Mel-spectra and F0 as input

	Natural	b-NSF	S-NSF	hn-NSF trainable MVF
Horn				
Trombone				
Tuba				
Clarinet				
Flute				

TTS

NSF + acoustic models

Natural	Joint train DAR + NSF from scratch	Joint fine-tuning DAR + NSF	1. Initialzie DAR and NSF 2. Train post-net	Best Pipeline

Not better