INTERSPEECH 2021
30 August – 3 September, 2021, Brno, Czechia

Towards simultaneous machine interpretation

Alejandro Pérez-González-de-Martos, Javier Iranzo-Sánchez, Adrià Giménez Pastor, Javier Jorge, Joan-Albert Silvestre-Cerdà, Jorge Civera, Albert Sanchis, Alfons Juan

Machine Learning and Language Processing (MLLP) research group
Valencian Research Institute for Artificial Intelligence (VRAIN)
Universitat Politècnica de València, Spain
{alpegon2,jairsan,adgipas,jajorca,joasilce,jorcisai,josanna2,ajuanci}@vrain.upv.es

Abstract

Automatic speech-to-speech translation (S2S) is one of the most challenging speech and language processing tasks, especially when considering its application to real-time settings. Recent advances on streaming Automatic Speech Recognition (ASR), simultaneous Machine Translation (MT) and incremental neural Text-To-Speech (TTS) make it possible to develop real-time cascade S2S systems with greatly improved accuracy. On the way to simultaneous machine interpretation, a state-of-the-art cascade streaming S2S system is described and empirically assessed in the simultaneous interpretation of European Parliament debates. We pay particular attention to the TTS component, particularly in terms of speech naturalness under a variety of response-time settings, as well as in terms of speaker similarity for its cross-lingual voice cloning capabilities.

Index Terms: speech-to-speech translation, incremental text-to-speech, cross-lingual voice cloning, simultaneous machine interpretation

1. Introduction

S2S has been one of the most challenging tasks in the field of natural language processing for many years [1, 2]. At this time, however, we are closer than ever to seeing accurate S2S systems thanks to the steady progress in deep learning for speech and language processing tasks, particularly ASR, MT and TTS. Indeed, although holistic, *end-to-end* systems are also being tried with good results, the more traditional *cascade* systems pipelining ASR, MT and TTS components, are now achieving impressive state-of-the-art results [3, 4].

Fueled by its immense applicability in real-life settings, research in *simultaneous* S2S is certainly gaining momentum. In this regard, to us, there are at least three main research challenges of great importance to build effective simultaneous cascade systems. First of all, the ASR component has to work under a strict *streaming* regime; that is, subject to the constraint that output must be delivered in nearly real time, only within a short delay or *latency* after the incoming audio stream. Then, as with its predecessor, the MT component has to deliver its output almost immediately, in a *simultaneous* fashion, even if no "complete" input sentence is available for translation. Finally, the TTS component is faced with the same constraint: speech synthesis has to start from just a prefix of the "complete" input sentence. All in all, these constraints make conventional (offline) sentence-level ASR, MT and TTS systems impractical for simultaneous S2S [5]. Instead, they have to be replaced by well-fitted systems for *streaming ASR, simultaneous MT* and *incremental TTS.*

This work presents a novel, full-fledged cascade simultaneous S2S system built from state-of-the-art streaming ASR, simultaneous MT, and incremental TTS components. The ASR component follows a hybrid HMM-DNN approach to streaming ASR we are developing from a streaming-optimized decoder built on top of advanced acoustic and language models also optimized for streaming [6, 7, 8]. The MT component is based on the recently proposed *multi-k* approach to simultaneous MT [9]. Then, at the end of the pipeline, the TTS component is designed according to the latest developments in the field, as an end-to-end deep learning architecture including a number of refinements for fast, incremental multilingual and multi-speaker TTS. Due to its novelty with respect to our preceding work, more emphasis is put on the TTS component. The assessment of the proposed system is conducted on a realistic simultaneous machine interpretation task for European Parliament debates, from the newly introduced Europarl-ST dataset [10]. For brevity, in this work we limit ourselves to machine interpretation from English to Spanish, and the other way around. Each component is first assessed in terms of quality, using appropriate, conventional criteria in each case, and then the full pipeline is evaluated in terms of latency on a single, standard PC. Although quality and speed are mutually conflicting objectives, and thus different trade-offs between them can be chosen, we show that results close to state-of-the-art quality can be achieved by the proposed S2S system running on a standard PC with an *ear-voice span* latency roughly doubling that of human interpreters (3–6 secs).

The ASR, MT and TTS components are first described in Sections 2, 3 and 4, respectively. Then, experiments are reported in Section 5, and conclusions in Section 6.

2. Streaming ASR

State-of-the-art ASR systems are based on the hybrid approach [11]. Bidirectional Long-Short Term Memory (LSTM) networks have shown to deliver highly competitive results for acoustic modelling in a wide range of ASR tasks [12, 13, 14]. Likewise, Transformer models have recently become the preferred choice for language modeling [15], though unidirectional LSTM recurrent neural networks are also widely used [16].

Our ASR component consists of a streaming-optimized one-pass decoder using acoustic and language models also optimized for streaming [6, 7]. In brief, we move from a sentence-based to a chunk-based training strategy, in which the input signal is processed by a sliding window over the audio stream. The acoustic score of each audio frame is computed within the limited past and future contexts imposed by the sliding window. On the other hand, decoder hypotheses are also scored using a Transformer language model whose computational cost is kept under control by pruning and variance regularization techniques [17].

http://dx.doi.org/10.21437/Interspeech.2021-201

3. Simultaneous Machine Translation

Mainstream MT architectures [18] require the entire source sentence to be available before generating its corresponding translation. On the contrary, simultaneous MT systems are characterized by a translation policy that dictates, at each point, whether enough context is available, or if we must wait for additional input words. These policies can be either fixed [19], if they used a set of simple deterministic rules, or adaptative [20] if they also depend on the specific input words which are available.

In this work, the fixed wait-k policy [19] is used. This policy waits for k words to be processed in the source text before starting the translation process. Then, every word processed in the source text generates a word in the target text. Additionally, we use the multi-k approach [9] to train systems that seamlessly may switch between different k at inference time, by sampling a random value for k for each training batch. This approach has been shown to obtain competitive results compared with the latest adaptative policies. An additional advantage of choosing a fixed policy is that the latency between words is fixed and consistent. In contrast, it has been observed that adaptative policies sometimes have spurious, longer than usual delays when translating between some input words. This fact greatly hinders the naturalness of the downstream TTS system. Thus, the prevention of these harmful long-tail events is another important reason for choosing a fixed policy over an adaptative one.

4. Incremental Multilingual Text-To-Speech

4.1. Adapted prefix-to-prefix framework

Conventional TTS models are designed to generate the speech signal corresponding to a given sentence or full semantical unit. When considering the tight response time constraints of a simultaneous S2S pipeline, it is unpractical to wait until the translated sentence is completed to start the synthesis process. This would incur in significant delays for the speech synthesis, particularly for long utterances.

Inspired on the prefix-to-prefix framework adopted for simultaneous MT [19], [21] proposes an adaptation for the incremental TTS task. Under this framework adapted to our case, the spectrogram and waveform are incrementally generated as

$$y_t = \Phi(x_{\leq g(t)}, y_{<t}) \quad (1)$$

$$w_t = \Psi(y_{\leq h(t)}, w_{<t}) \quad (2)$$

where x, y and w represent the input text, the speech spectrogram and the audio waveform, respectively; and $g(t)$ and $h(t)$ are monotonic functions that define the number of words in Eq. 1 or frames in Eq. 2, being conditioned on when generating the outputs for the t^{th} word.

For the spectrogram generation $g(t)$, we follow the same lookahead-k policy as in [21]

$$g(t) = \min(t + k, |x|). \quad (3)$$

We also introduce a maximum history context size δ_y to limit the computational complexity of the incremental text-to-spectrogram inference process, so that Eq. 1 becomes

$$y_t = \Phi(x_{t-\delta_y}^{g(t)}, y_{t-\delta_y}^{t-1}). \quad (4)$$

For the waveform generation, we condition the neural vocoder model on the generated spectrograms for word t plus an additional small context from word $t-1$. In particular, we set $h(t) = t$ and we define a maximum history context size δ_w corresponding to the number of trailing frames from word $t-1$ that will be used as context. We do not use future context for the vocoding step as the waveform reconstruction does not rely on long term context.

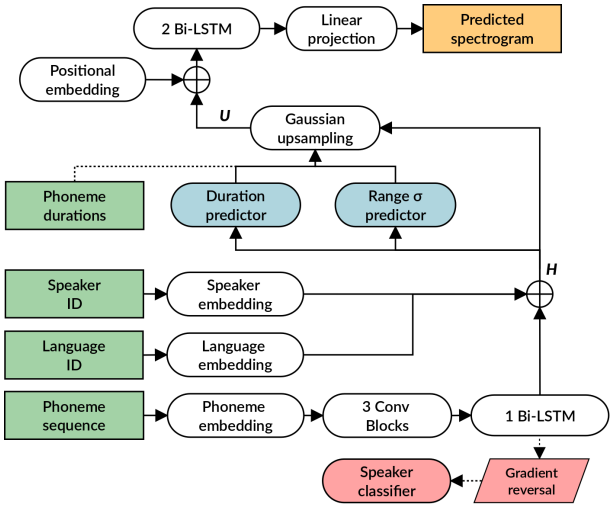

Figure 1: *Proposed multilingual multi-speaker TTS model architecture. Dashed lines are training-only connections.*

4.2. Model architecture

Our model is based on ForwardTacotron[1], which is an open-source non-autoregressive variant of Tacotron [22] inspired on duration informed neural TTS models like FastSpeech [23] or DurIAN [24]. Fig. 1 depicts our modifications introduced in the original model architecture mentioned above.

First, we replace the encoder module with the simplified architecture proposed in Tacotron-2 [25] shown at the bottom of Fig. 1. Second, we introduce speaker and language embeddings as in [26]. Input characters are converted into a phoneme sequence, where equivalent phonemes are shared across languages. The language embedding helps disambiguate phoneme pronunciations between languages. In addition, we introduce an adversarial speaker classifier as in [26] to prevent encoder hidden states from containing speaker-related information. Finally, we replace the vanilla upsampling through repetition (also known as length regulator module) with the Gaussian upsampling approach recently proposed in [27], which has been shown to improve speech naturalness. After upsampling, we add Transformer-style sinusoidal positional embeddings of a frame position with respect to the current phoneme as in [28].

The text-to-spectrogram models are trained using a combination of the $\mathcal{L}1$ loss and the *structural similarity index measure* (SSIM) between the predicted and the target spectrograms, and *Hubber loss* for logarithmic duration prediction [29]. Cross-entropy loss is used for training the adversarial speaker classifier.

The public Multi-band MelGAN implementation[2] is used for reconstructing the audio waveform conditioned on the generated spectrograms, which is able to generate high quality speech with a real-time factor of 0.03 on CPU [30].

[1] https://github.com/as-ideas/ForwardTacotron

[2] https://github.com/kan-bayashi/ParallelWaveGAN

Table 1: *English and Spanish Europarl-ST subsets considered to train TTS models.*

Source language	Hours	Words	Speakers
English	72.5	723K	282
Spanish	27.5	276K	85

Table 2: *ASR results in terms of WER [%] on the English and Spanish Europarl-ST test sets.*

System	English	Spanish
Offline	12.7	9.8
Streaming	13.4	10.7

5. Experiments

In this section, we first describe the Europarl-ST dataset, then individual components are evaluated with special emphasis on the TTS component, and finally latency figures of the complete S2S system are reported.

5.1. Corpus description

Europarl-ST [10] is a multilingual spoken language translation dataset built from the European Parliament debates archive in the period between 2008 and 2012. In its more recent version (v1.1), it contains paired <*audio, transcription, translation*> samples covering 9 different official European languages. The transcription in the original source language of the MEP intervention and their corresponding translations are available for different source-target language pairs. In this work, the *train-noisy* subsets of this dataset are discarded.

To train TTS models, English and Spanish *train*, *dev* and *test* subsets Europarl-ST were jointly considered. Conflictive samples containing OOV symbols, dates, acronyms, etc. were filtered out. The resulting number of hours, words and speakers per language are given in Table 1.

5.2. ASR and MT evaluation

General-purpose English and Spanish ASR systems were trained on a collection of thousands of speech transcribed hours from public and private datasets according to the recipe described in [8]. A sliding-window segmenter [31] is applied to the ASR output in order to define chunks to be fed to the MT component. The performance of these systems was evaluated in terms of Word Error Rate (WER). Table 2 provides WER figures in percentage for offline and streaming ASR systems, showing only a small degradation as a result of the limited speech context available to the latter.

Similarly, we trained English and Spanish simultaneous MT systems on millions of sentence pairs from OPUS parallel public datasets [32]. Table 3 reports comparative BLEU [33] scores on the Europarl-ST corpus between the conventional offline system and a range of simultaneous wait-k MT systems as a function of k. As shown in Table 3, there is a relevant gap in performance between the offline and simultaneous MT systems, which decreases as the value of k increases. Values of k higher than 8 bring negligible quality improvements. The BLEU difference between both systems when the simultaneous wait-k system works in offline mode ($k = 100$) is due to the use of an unidirectional encoder instead of a bidirectional one.

Table 3: *BLEU scores of the offline and simultaneous wait-k MT systems on the Europarl test set.*

System	k	English-Spanish	Spanish-English
Offline	-	47.4	41.3
Wait-k	1	32.2	31.0
	2	37.8	33.9
	4	42.6	35.1
	8	44.1	36.0
	32	44.3	36.1
	100	44.3	36.2

In streaming settings, translation quality and response-time should be balanced. For lower response-time setups, such as k=2 and k=4, the incurred gap ranges from 5 to 9 BLEU points. All in all, the BLEU scores achieved by simultaneous MT systems ($\simeq$ 34-43) indicate that the quality of these systems is accurate enough for a TTS component downstream.

5.3. TTS training details

We trained a reference full-sentence model, and lookahead models with values $k \in \{0, 1, 2\}$. The latter were trained by randomly selecting the first n words from each training sample for each epoch, and limiting the lookahead context to a maximum of k subsequent words.

The models were trained on the English and Spanish Europarl-ST subsets detailed in Table 1. We set the sample rate to 22kHz and extracted 100 bin log magnitude Mel-scale spectrograms with Hann windowing, 50ms window length, 12.5ms hop size and 1024 point Fourier transform. Phoneme durations were extracted by training an autoregressive attention-based Tacotron-2 model. Models were trained for 160K steps with batch size 48 and an initial learning rate of 0.0005, which was decayed by half every 40K steps. We used the Adam optimizer [34] with default parameters and gradient clipping at 1.

To reduce the harmful effects of noisy samples (i.e. containing significant background noise or inaccurate transcriptions), after 5K steps we masked the loss from those samples within the batch for which the $\mathcal{L}1$ loss was above a certain threshold. This threshold was computed dynamically as the median plus λ times the median absolute deviation of the $\mathcal{L}1$ loss from the last 5 batches, where λ was manually adjusted empirically.

On inference time, we fixed the maximum history context size parameter δ_y introduced in Section 4.1 to 6 words in all experiments. Ma et al. [21] define a chunk-level inference procedure so that the minimum number of words to synthesize at each step is such that the current chunk contains at least l phonemes (where l is a manually defined hyperparameter). We set l to 6.

5.4. TTS evaluation

To evaluate the naturalness and quality of the synthetic speech, we used 20 particularly long utterances as test set, 10 for each language, where each sample corresponds to a different speaker. We evaluated the speech naturalness of both regular and cross-lingual samples through a 5-scale Mean Opinion Score (MOS) subjective listening test. We also assessed the cross-lingual voice cloning capabilities of the proposed model by analyzing the speaker similarity between the synthetic and ground-truth utterances from the same speaker. Ten native Spanish speakers with proficiency of the English language participated in the sub-

Table 4: *Speech naturalness MOS with 95% CI.*

Regular	English	Spanish
k=0	3.03 ± 0.22	2.67 ± 0.23
k=1	3.17 ± 0.22	3.44 ± 0.21
k=2	3.33 ± 0.22	3.44 ± 0.24
Full-sentence	4.15 ± 0.20	4.17 ± 0.19
Cross-lingual	English	Spanish
k=0	2.85 ± 0.22	2.26 ± 0.22
k=1	3.23 ± 0.20	2.77 ± 0.23
k=2	3.18 ± 0.20	2.97 ± 0.21
Full-sentence	4.04 ± 0.19	3.99 ± 0.20

Table 5: *Speaker similarity MOS with 95% CI of cross-lingual samples compared with a reference utterance.*

Model	English	Spanish
Full-sentence	3.84 ± 0.22	3.67 ± 0.21

jective listening test. The reader is encouraged to listen to the cross-lingual synthetic samples [3].

Table 4 shows the 5-scale naturalness MOS with 95% confidence intervals (CI) for the different model configurations, both for English and Spanish, for regular and cross-lingual samples. We can appreciate a small degradation on the speech naturalness for the cross-lingual synthesis (e.g. 4.0 compared to 4.2 for the full-sentence case). Nevertheless, this gap is small and endorses the satisfactory performance of the proposed cross-lingual approach. It can be also noted how speech naturalness is degraded for incremental TTS configurations. We hypothesize this is due to two main factors. First, the impact of limiting the context both to past and future words. Second, the unnatural duration of pauses introduced between consecutive chunks due to the proposed incremental prefix-to-prefix approach.

Table 5 shows 5-scale speaker similarity MOS with 95% CI measuring how the cross-lingual synthesized voice resembles that of the original speaker. We limited this evaluation to the full-sentence scenario as the incremental settings should have little impact regarding voice cloning capabilities.

5.5. S2S latency evaluation

The accumulated latency of the English $\rightleftarrows$ Spanish S2S systems are measured similarly to [35, 31]. We define accumulative chunk-level latencies at five successive points in the S2S pipeline, as the time elapsed between the last word of a chunk being spoken and: 1) The consolidated hypothesis for that chunk is provided by the ASR system; 2) The segmenter defines that chunk on the ASR consolidated hypothesis; 3) The MT system translates the chunk defined by the segmenter; 4) The TTS finishes synthesizing the translated chunk; 5) The synthesized audio playback is finished. The latter is closely related to the Ear-Voice Span (EVS) metric commonly used in simultaneous interpretation . All five latency figures are reported in Table 6 for both translation directions. Latency tests were run on a NVIDIA GeForce GTX 2080 Ti.

The latency added by the segmentation is mostly explained by the consolidation of the ASR output, that also depends on the long-range dependencies of the neural language models behind. The wait-k ($k = 4$) policy of MT and the lookahead-k policy of TTS define the corresponding latencies introduced by these components of the pipeline.

[3] All generated cross-lingual synthetic samples are available at https://mllp.upv.es/interspeech21-demo/

Table 6: *Accumulative latency mean and standard deviation in seconds for the successive points in the S2S pipeline.*

Model	English-Spanish	Spanish-English
1) ASR	2.7 ± 1.5	1.8 ± 1.2
2) Segmenter	3.5 ± 1.8	2.8 ± 1.3
3) MT ($k = 4$)	5.2 ± 2.4	4.3 ± 2.0
4) TTS		
$k = 0$	5.6 ± 2.5	4.6 ± 2.0
$k = 1$	5.8 ± 2.6	4.8 ± 2.2
$k = 2$	6.0 ± 2.7	5.1 ± 2.3
5) Playback		
$k = 0$	8.1 ± 1.7	6.4 ± 1.6
$k = 1$	8.7 ± 1.7	7.1 ± 1.7
$k = 2$	9.1 ± 1.7	7.6 ± 1.7

The playback delay is explained by the impossibility to recover from the delays introduced in previous chunks when the synthesized speech in the target language is longer or equal to the original speech in the source language. This could be addressed by explicitly controlling the speaking rate of the synthetic speech. The EVS latency of our S2S system configuration ranges approximately from 7 to 9 seconds. This is an acceptable range, not far from that of human interpreters, which varies between 3 to 6 seconds [36]. Nevertheless, the S2S pipeline components can be tuned for a trade-off between response time and translation performance, depending on the application needs.

6. Conclusions

This paper presented a state-of-the-art simultaneous S2S system composed of streaming ASR, simultaneous MT and multilingual incremental TTS components connected in a cascaded manner. The proposed speech synthesis architecture enables cross-lingual synthesis while preserving reasonably similar vocal characteristics. We analyzed the impact of limiting the future context for the proposed incremental TTS approach in terms of speech naturalness, and provided latency figures for the individual components of the pipeline. The figures presented for the streaming ASR and simultaneous MT, as well as the subjective speech naturalness MOS achieved by the proposed TTS models makes us very optimistic about the utilization of these technologies in real-life applications.

As future work, individual components in the S2S pipeline can be improved by incorporating long-span dependency neural models that naturally take advantage of a streaming scenario to improve system accuracy. We will also investigate the impact of directly controlling the synthesis speaking rate to recover from the successively added delays in the playback for very long speech segments.

7. Acknowledgements

Work supported by the EU's H2020 research and innovation programme under grant agreement no. 761758 (X5gon), the Spanish government under grant RTI2018-094879-B-I00 (Multisub, MCIU/AEI/FEDER), and the Universitat Politècnica de València's PAID-01-17 R&D support programme.

8. References

[1] A. Lavie *et al.*, "JANUS-III: Speech-to-speech translation in multiple languages," in *Proc. of ICASSP 1997*, vol. 1, 1997, pp. 99–102.

[2] W. Wahlster, *Verbmobil: foundations of speech-to-speech translation*. Springer Science & Business Media, 2013.

[3] Y. Jia, R. J. Weiss, F. Biadsy, W. Macherey, M. Johnson, Z. Chen, and Y. Wu, "Direct Speech-to-Speech Translation with a Sequence-to-Sequence Model," in *Proc. Interspeech 2019*, 2019, pp. 1123–1127. [Online]. Available: http://dx.doi.org/10.21437/Interspeech.2019-1951

[4] M. Sperber *et al.*, "Attention-passing models for robust and data-efficient end-to-end speech translation," *Transactions of the Association for Computational Linguistics*, vol. 7, pp. 313–325, 2019.

[5] K. Sudoh *et al.*, "Simultaneous Speech-to-Speech Translation System with Neural Incremental ASR, MT, and TTS," *arXiv preprint arXiv:2011.04845*, 2020.

[6] J. Jorge *et al.*, "Real-Time one-pass decoder for speech recognition using LSTM language models," in *Proc. of Interspeech*, 2019, pp. 3820–3824.

[7] J.Jorge *et al.*, "LSTM-Based One-Pass Decoder for Low-Latency Streaming," in *Proc. of ICASSP*, 2020, pp. 7814–7818.

[8] P. Baquero-Arnal *et al.*, "Improved Hybrid Streaming ASR with Transformer Language Models," in *Proc. of InterSpeech*, 2020, pp. 2127–2131.

[9] M. Elbayad *et al.*, "Efficient Wait-k Models for Simultaneous Machine Translation," in *Proc. of Interspeech*, 2020, pp. 1461–1465.

[10] J. Iranzo-Sánchez *et al.*, "Europarl-ST: A Multilingual Corpus for Speech Translation of Parliamentary Debates," in *Proc. of ICASSP*, 2020, pp. 8229–8233.

[11] D. Yu and L. Deng, *Automatic Speech Recognition: A Deep Learning Approach*. Springer Publishing Company, Incorporated, 2014.

[12] A. Graves and J. Schmidhuber, "Framewise phoneme classification with bidirectional LSTM and other neural network architectures," *Neural networks*, vol. 18, no. 5-6, pp. 602–610, 2005.

[13] W. Chan and I. Lane, "Deep recurrent neural networks for acoustic modelling," *arXiv preprint arXiv:1504.01482*, 2015.

[14] K. Chen and Q. Huo, "Training deep bidirectional LSTM acoustic model for LVCSR by a context-sensitive-chunk BPTT approach," *IEEE/ACM TASLP*, vol. 24, no. 7, pp. 1185–1193, 2016.

[15] K. Irie *et al.*, "Language Modeling with Deep Transformers," in *Proc. of Interspeech*, 2019, pp. 3905–3909.

[16] R. Jozefowicz *et al.*, "Exploring the limits of language modeling," *arXiv preprint arXiv:1602.02410*, 2016.

[17] Y. Shi *et al.*, "Efficient one-pass decoding with NNLM for speech recognition," *IEEE Signal Processing Letters*, vol. 21, no. 4, pp. 377–381, 2014.

[18] A. Vaswani *et al.*, "Attention is all you need," in *Proc. of NIPS*, 2017, pp. 5998–6008.

[19] M. Ma *et al.*, "STACL: Simultaneous translation with implicit anticipation and controllable latency using prefix-to-prefix framework," *arXiv preprint arXiv:1810.08398*, 2018.

[20] N. Arivazhagan *et al.*, "Monotonic Infinite Lookback Attention for Simultaneous Machine Translation," in *Proc. of ACL*, 2019, pp. 1313–1323.

[21] M. Ma *et al.*, "Incremental text-to-speech synthesis with prefix-to-prefix framework," in *Proc. of EMNLP*, 2020, pp. 3886–3896.

[22] Y. Wang *et al.*, "Tacotron: Towards end-to-end speech synthesis," in *Proc. of Interspeech*, 2017, pp. 4006–4010.

[23] Y. Ren *et al.*, "FastSpeech: Fast, Robust and Controllable Text to Speech," in *Proc. of NIPS*, 2019.

[24] C. Yu *et al.*, "Durian: Duration informed attention network for speech synthesis," *Proc. of Interspeech*, pp. 2027–2031, 2020.

[25] J. Shen *et al.*, "Natural TTS Synthesis by Conditioning Wavenet on MEL Spectrogram Predictions," in *Proc. of ICASSP*, 2018, pp. 4779–4783.

[26] Y. Zhang *et al.*, "Learning to Speak Fluently in a Foreign Language: Multilingual Speech Synthesis and Cross-Language Voice Cloning," in *Proc. of Interspeech*, 2019, pp. 2080–2084.

[27] J. Shen *et al.*, "Non-Attentive Tacotron: Robust and Controllable Neural TTS Synthesis Including Unsupervised Duration Modeling," arXiv preprint arXiv:2010.04301, 2020.

[28] I. Elias *et al.*, "Parallel Tacotron: Non-Autoregressive and Controllable TTS," arXiv preprint arXiv:2010.11439, 2020.

[29] J. Vainer and O. Dušek, "SpeedySpeech: Efficient Neural Speech Synthesis," in *Proc. of Interspeech*, 2020, pp. 3575–3579.

[30] G. Yang *et al.*, "Multi-band MelGAN: Faster Waveform Generation for High-Quality Text-to-Speech," *arXiv preprint arXiv:2005.05106*, 2020.

[31] J. Iranzo-Sánchez *et al.*, "Direct Segmentation Models for Streaming Speech Translation," in *Proc. of EMNLP*, 2020, pp. 2599–2611.

[32] J. Tiedemann, "Parallel Data, Tools and Interfaces in OPUS," in *Proc. of LREC*, 2012.

[33] K. Papineni *et al.*, "Bleu: a Method for Automatic Evaluation of Machine Translation," in *Proc. of ACL*, 2002, pp. 311–318.

[34] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. of ICLR*, Y. Bengio and Y. LeCun, Eds., 2015. [Online]. Available: http://arxiv.org/abs/1412.6980

[35] B. Li *et al.*, "Towards Fast and Accurate Streaming End-To-End ASR," in *Proc. of ICASSP*, 2020, pp. 6069–6073.

[36] M. Lederer, "Simultaneous interpretation—units of meaning and other features," in *Language interpretation and communication*. Springer, 1978, pp. 323–332.