Evaluating Text-to-Speech and Audio Codec Performance For Voice Communication in Resource-Constrained Networks

Batuhan Mekiker
Beartooth Radio
batuhan@beartooth.com

Mike P. Wittie
Montana State University
mwittie@cs.montana.edu

***Abstract*—Voice communications are valued for their ease of use and the rich information they provide, offering an immediate, clear, and efficient way to convey messages. However, ensuring the clarity and reliability of voice communications in low-bandwidth networks poses a technical challenge. This research explores the efficacy of Text-to-Speech (TTS) models and vocoder combinations versus traditional audio codecs in low-bandwidth networks, highlighting considerations for voice clarity and network resource management. Traditional audio codecs in bandwidth-limited environments often compromise audio quality and reliability. On the contrary, TTS models, supported by the advancements in deep and machine learning, present a potential alternative. Through a methodical comparison using various evaluation metrics, the study aims to offer valuable insights into their comparative impacts on audio quality and network behavior.**

***Index Terms*—TTS, Audio Codecs, CLIP, Voice Communication, Resource-Constrained Networks**

I. Introduction

Voice communications over networks are pivotal in many scenarios, largely due to their inherent efficiency and the rich information they can transmit. Unlike text-based methods, a voice message conveys both the identity of the user and the content of the message without the need for recipients to physically engage with their devices. This direct and easily discernible method of communication is crucial where quick and unmistakable understanding is vital. However, maintaining the clarity and reliability of voice communications in settings with limited bandwidth introduces a challenge.

Traditionally, audio encoding has been the go-to solution for voice communication needs over networks, serving as the conventional method for transmitting voice. While audio codecs are optimized for many scenarios, and may even function in low-bandwidth networks, their dependency on network performance for real-time communication becomes a glaring limitation in bandwidth-constrained networks, especially when congestion happens in urban areas or disruption occurs due to terrain-induced effects on Received Signal Strength (RSS) and link quality degradation in rural wireless networks. The compromise then is often on the audio quality, delay, and disruption tolerances. These trade-offs not only jeopardize the clarity and accuracy of the message but also become notably detrimental in scenarios such as mission-critical applications where the accuracy and prompt delivery of the information are crucial.

Moreover, the landscape of audio communication is rapidly evolving. The advancements in neural networks, deep learning techniques, and the pace of hardware development are pushing the boundaries of what is possible in voice communication over resource-constraint wireless networks. The evolution in technology introduces a novel method that capitalizes on Text-to-Speech (TTS) models to address the challenge of clear voice communication in resource-constrained networks. Instead of transmitting larger packets of encoded voice data, the strategy of utilizing TTS models involves sending only text and basic user information. Once received, the voice can be regenerated at the receiver, capitalizing on the fact that text data is significantly leaner compared to its encoded audio counterpart. While traditional audio codecs do not require excessive bandwidth, their data packets are considerably larger and demand more bandwidth to transmit compared to the data needed to regenerate TTS audio. Employing TTS for audio communication offers an efficient means to manage scarce resources in limited-bandwidth scenarios. This approach shows a robust, resource-efficient alternative for voice communication in resource-constrained networks.

The main contribution of this paper is to provide insight into the comparative nuances of utilizing Text-to-Speech (TTS) models with varying vocoders versus traditional audio codecs in low-bandwidth networks. The insight derived provides a valuable perspective toward improving voice intelligibility, quality, clarity, and managing valuable network resources.

Similar to the work by Dantas et al. in 2019 on a speech-to-text-to-speech pipeline [1], our study expands the scope by analyzing combinations of Text-to-Speech (TTS) models, vocoders, audio codecs, and their effects on audio quality and network performance in resource-constrained scenarios. Since the work of Dantas et al., which demonstrated the potential of TTS systems in voice communication, rapid advancements have led to the introduction of new machine and deep learning-based codecs and new developments in TTS systems. Furthermore, Dantas et al. conducted their evaluation using only one audio codec (PCM) and one TTS system (Baidu's Deep Speech architecture). Their methodology

relied on participant assessments, Word Error Rate for recognizing TTS outputs, and Levenshtein Distance [2] to discern words and quantify the number of edits required to correct the text for the speech-to-text process. These methods, susceptible to human error and bias, contrast with our research, which adopts a more robust, quantitative approach. We evaluate the efficiency of different TTS systems and vocoder combinations using quantitative metrics such as Fréchet Distance [3], Intelligibility Score (IS) based on Automated Speech Recognition (ASR), and Contrastive Language-Voice Pretrained (CLVP) [4] scores to assess these technologies under resource constraints. This approach provides a more standardized and reproducible measure of performance, emphasizing the impact of different TTS systems and vocoder combinations on network performance and highlighting areas where TTS could significantly enhance user experience and network efficiency.

Our initial findings indicate a promising trend and potential replacement of audio codecs with TTS systems. Specifically, the VITS [5] model delivers remarkable clarity, closely mirroring the original recordings, while Fastspeech2 [6] impresses with its rapid sample generation. As we dive deeper, it becomes evident that TTS systems might not just be alternatives, but potentially perform better in limited bandwidth networks.

The rest of this paper is organized as follows. Section II explores TTS models, their key components, audio codecs, and the evaluation metrics for both TTS and audio codecs. In Section III we present measurements and analyze the results. Finally, we conclude in Section IV by summarizing our findings.

II. Background

In this section, we explore key components and concepts of our research on voice communication in limited networks. We discuss Text-to-Speech (TTS) models and their main components, describe key concepts and features of the audio codecs we used, and describe the metrics that guide our evaluation.

A. Text-to-Speech

Text-to-Speech (TTS) synthesizes understandable and natural-sounding speech from text using natural language processing, signal processing, and machine learning. The process involves three key components: Text Analysis, Acoustic Modeling, and Vocoding.

Text Analysis processes raw text into linguistic features, handling pronunciation, normalization, and segmentation [7], [8]. Modern end-to-end neural TTS methods have simplified this stage but tasks like grapheme-to-phoneme conversion are still crucial for managing diverse text formats [9].

Acoustic Modeling transforms these linguistic features into spectral representations, preparing them for vocoding [10]. Different models address various TTS challenges. For example, Tacotron employs a sequence-to-sequence model with attention to map text to mel spectrograms, while FastSpeech uses a non-autoregressive method for faster synthesis [6], [11]. VITS on the other hand, combines Variational AutoEncoders (VAEs) and adversarial training from GANs for high-quality speech output [5].

The last component, vocoding, is responsible for generating the playable speech waveform. Techniques vary among vocoders: Autoregressive vocoders operate sequentially, potentially slowing down speech generation [12], while Flow-based vocoders use normalizing flows for faster, parallel waveform generation [13]. GAN-based vocoders like Parallel WaveGAN and MelGAN optimize waveform quality using Generative Adversarial Networks [14].

B. Audio Codecs

Audio codecs compress audio samples for transmission over networks, typically using lossy compression to reduce file size at the expense of clarity and quality. Our study focuses on open-source codecs suitable for low-bandwidth networks.

Starting with a more traditional audio codec, Codec 2 is a low-bitrate codec using sinusoidal coding optimized for human speech, operating at bit rates from 450 bit/s to 3.2 Kbit/s, making it ideal for Mobile Ad-Hoc Networks (MANETs) [15]. This technique models speech using harmonically related sine waves, efficiently encoding pitch and amplitude.

Transitioning from the traditional methods employed by Codec 2, Google's Lyra represents a modern approach by integrating a machine learning technique, generative model to recreate the speech signal for audio compression [16]. It enhances audio quality at low bitrates (3.2 Kbit/s to 9 Kbit/s), making it suitable for real-time communications in bandwidth-constrained environments.

Similarly, Facebook's Encodec uses a neural network-based encoder-decoder architecture for high-fidelity audio compression at various rates (1.5 to 24 Kbit/s) [17]. Both Lyra and Encodec utilize neural techniques to improve compression efficiency and audio quality, catering to streaming and communication in resource-limited networks.

C. Metrics

Next, we delve into the metrics we used for evaluating both TTS models and audio codecs in low-bandwidth networks. These metrics offer a measurable insight into the quality and efficiency of the audio compression techniques and efficacy of TTS systems.

1) Fréchet Distance: The first metric we utilize is the Fréchet Distance (FD), a concept described in detail in work by Alt et al. [3]. In audio codecs and Text-to-Speech (TTS) models, FD plays a pivotal role in quantitatively assessing model performance. Specifically, FD is employed to compare the reference audio and the audio that is in question. However, it is crucial to clarify that in our approach, we do not directly compute Fréchet Distance on voice signals but rather extract feature vectors from the audio to compute FD. Extracting feature vectors offers significant advantages: it reduces the complexity of voice signals, therefore enhancing computational

efficiency, and normalizes the data, ensuring comparability between different samples. To compute FD, we first use feature vectors extracted via CLVP model for both real and generated/processed speech samples. Then, we calculate the mean and covariance of these feature vectors for each set of samples. The FD score is finally obtained by measuring the Fréchet Distance between the two Gaussian distributions represented by these statistical measures. Mathematically, this distance is given by:

$$FD = ||\mu_r - \mu_g||^2 + Tr(\Sigma_r + \Sigma_g - 2(\Sigma_r\Sigma_g)^{1/2}) \quad (1)$$

where, μ_r, μ_g represent the means, and Σ_r and Σ_g denote the covariance matrices of the feature vectors from the real and generated speech samples, respectively. Tr is the trace of a matrix, representing the sum of its main diagonal elements.

A lower FD score indicates a closer resemblance between the two distributions, signifying a higher fidelity of the generated speech in mirroring real spoken text, thereby reflecting superior model performance.

2) CLVP Score: Inspired by OpenAI's Contrastive Language-Image Pretraining (CLIP), the Contrastive Language-Voice Pre-trained (CLVP) model adapts this approach for audio-text pairs [18]. It uses contrastive learning to distinguish and align corresponding audio-text pairs, enhancing model accuracy and understanding with datasets like LJ Speech [19]. The dual-encoder architecture of the CLVP model processes audio clips and textual descriptions to transform them into embeddings. These embeddings are projected into a shared latent space, where the model calculates the CLVP score by performing a dot product called Einstein Sum between matched pairs [20]. This score measures the similarity between text and speech embeddings, with higher scores indicating better alignment and model effectiveness.

3) Intelligibility Score: The Intelligibility Score (IS) evaluates TTS systems using the Wav2Vec model, which discerns correct audio snippets from distractors with a contrastive loss function [21]. Specifically, the `Wav2Vec2ForCTC` transcribes TTS-generated audio for comparison with the original text. This process, grounded in Connectionist Temporal Classification (CTC) loss, aligns audio input with text output without fixed sequence alignment [22]. As shown in Figure 1, CTC loss introduces a 'blank' character for managing sequence discrepancies, calculates the likelihood of accurate transcriptions by considering all potential alignments. This ensures that the model optimizes for accuracy during training.

In TTS systems, the Intelligibility Score (IS) uses the CTC loss function to evaluate the accuracy of ASR models in transcribing TTS-generated speech. This involves normalizing audio samples from both TTS and real speech to unify input levels. The ASR model, `Wav2Vec2ForCTC`, computes CTC loss by comparing its transcriptions to the actual text and adjusting for natural speech variations when real speech is present. The final IS, the average of these adjusted losses,

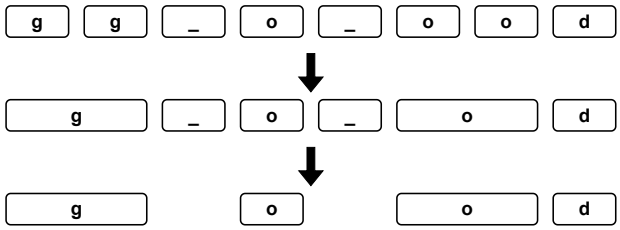

Fig. 1: Steps taken by CTC method discern the word 'good'.

measures how closely the TTS-generated speech matches the original text in clarity and content, with lower scores indicating better model intelligibility and naturalness.

4) Inference Time: In the context of real-time voice communication over bandwidth-constrained networks, evaluating TTS models using the inference time metric is crucial because it directly affects real-time data delivery in real-time voice communication scenarios. In this study, inference time was precisely measured from the timestamp the input text was provided to the TTS model until the audio output file was generated. The duration a TTS model takes to translate text into natural-sounding speech is a key performance indicator in environments where network resources are limited, and keeping latency as low as possible is essential. For such applications, it is important that the TTS model not only generates clear and understandable audio but does so with minimal delay. This requirement is vital in maintaining effective communication, ensuring that the generated speech is delivered promptly without taxing the limited network resources, especially in mission-critical applications. The challenge lies in optimizing TTS models to achieve a balance between swift response times and maintaining speech clarity, all within the constraints of limited bandwidth. This balancing act is especially crucial in domains like mission-critical applications where real-time data delivery is as important as the quality and clarity of the voice signal output.

III. Methodology

We conducted experiments using a computer equipped with an Intel Core i9-9900K CPU, NVIDIA GeForce RTX 2080 SUPER GPU (3072 CUDA cores), 32 GB of RAM, and an INTEL 660P series SSD, ensuring efficient data processing for model training. We utilized the LJ Speech dataset, which includes approximately 24 hours of single-speaker English recordings at a 22.05 KHz sampling rate, for its extensive use in TTS research and for facilitating robust comparisons of TTS models with traditional audio codecs [19].

We tested three prominent TTS models: FastSpeech2 [6], Tacotron2 [11], and VITS [5], using the ESPNet 2 framework [23]. These models were paired with various vocoders including Parallel WaveGAN [24], HiFiGAN [25], Style MelGAN [26], Fullband MelGAN, and Multiband MelGAN [27]. Additionally, we used audio codecs such as Codec 2 [15], Lyra [16] and Encodec [17]–suitable for

Vocoder	# of Iterations	Checkpoint Size
Parallel WaveGAN (v3)	3M	214.9MB
Fullband MelGAN (v2)	1M	138.4MB
Multiband MelGAN (v2)	1M	105.3MB
HiFiGAN (v1)	2.5M	968.9MB
Style MelGAN (v1)	1.5M	108.5MB

TABLE I: Training and checkpoint information for vocoders used in the evaluation.

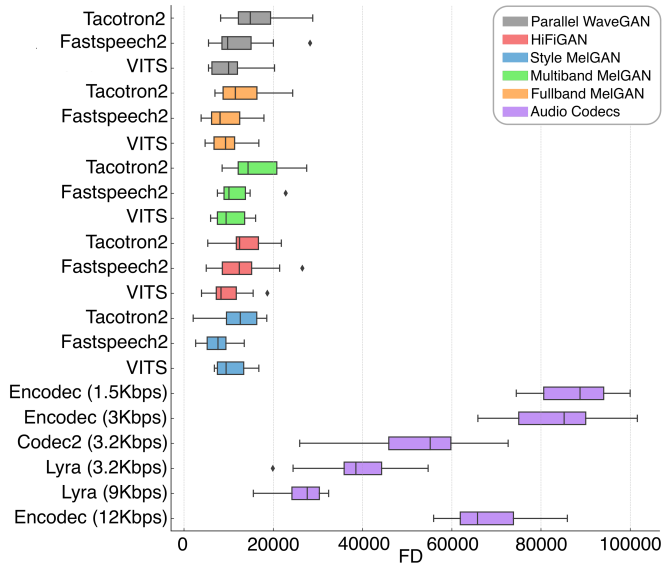

Fig. 2: Frechet Distance between the original sample from LJ Speech dataset and synthesized/decoded speech.

bandwidth-constrained networks. We randomly selected voice samples from the LJ Speech dataset, and generated, encoded and decoded the audio samples using the chosen codecs. The effectiveness of the TTS models was assessed by comparing the TTS-generated samples with the codec-processed versions using specific evaluation metrics.

In this section, we compare TTS results to traditional audio codecs. To achieve objective comparison we use the following metrics.

A. *Frechet Distance*

Figure 2 presents the distribution of FD values among 15 unique combinations of TTS models and 6 audio codecs. The x-axis denotes the FD values, while the various pairings of TTS models and audio codecs are outlined on the y-axis.

The Figure 2 shows that both FastSpeech2 and Tacotron2 exhibit higher FD values compared to VITS, indicating that VITS maintains a closer resemblance to the original recording during its speech synthesis. It is also evident that Tacotron2 displays a broader range of values, suggesting some level of inconsistency in its output. In the realm of vocoders, Style MelGAN and Fullband MelGAN consistently demonstrate lower FD values, outperforming their counterparts when integrated with all three models.

In comparing audio codecs, it is clear that the combined output from any TTS model and vocoder more closely

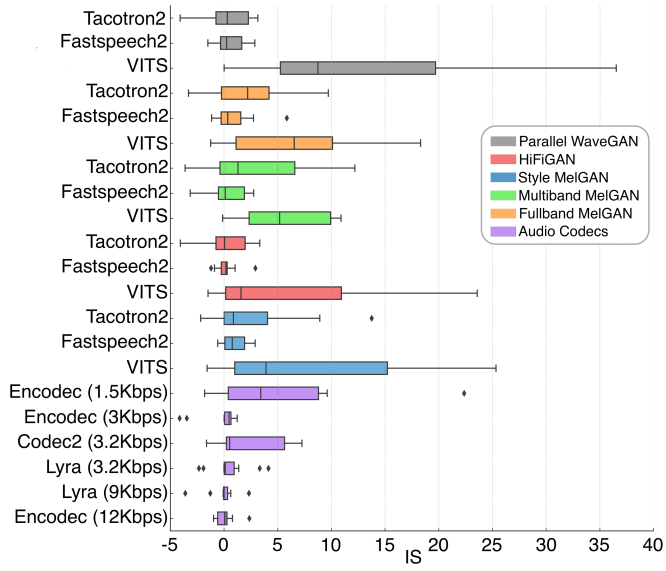

Fig. 3: Intelligibility Score based on LJ Speech transcripts and synthesized/decoded speech.

mirrors the original sample than that produced solely by the audio codecs. As anticipated, a reduction in encoding rate is associated with a compromise in quality. This correlation is pronounced in the Encodec with a 1.5 Kbit/s encoding rate, which exhibits the highest FD, diverging most from the original sample.

When drawing parallels among audio codecs with proximate encoding rates – Encodec at 3 Kbit/s, Codec2, and Lyra at 3.2 Kbit/s – Encodec produces speech with higher FD. While Codec2 and Lyra showcase comparable efficacy, Lyra slightly edges out Codec2, possibly due to its unconventional audio encoding approach. Notably, despite boasting a loftier encoding rate, Encodec at 12 Kbit/s still registers a higher FD than Lyra at 9 Kbit/s. This observation clearly shows the optimized nature of the Lyra audio codec, marking it as a better choice over Encodec.

B. *Intelligibility Score (IS)*

Figure 3 depicts the distribution of IS across 15 different combinations of TTS models paired with 6 distinct audio codecs. Similar to Figure 2, on the x-axis we have IS, whereas the y-axis represents the various combinations of TTS models and audio codecs.

An observation emerging from the data is the pronounced spread of VITS model in its distribution relative to the other two models. Specifically, when paired with Parallel WaveGAN, this combination yields results with notable variability and it suggests that the remaining models offer a more consistent mapping to the original transcript. Furthermore, Fastspeech2 manifests the narrowest distribution, leading to highly consistent outcomes. In contrast, Tacotron2, despite its broader spread, consistently reports the lowest IS across all vocoder pairings.

Overall, audio codecs and TTS models seem to showcase comparable performance. However, a subtle performance

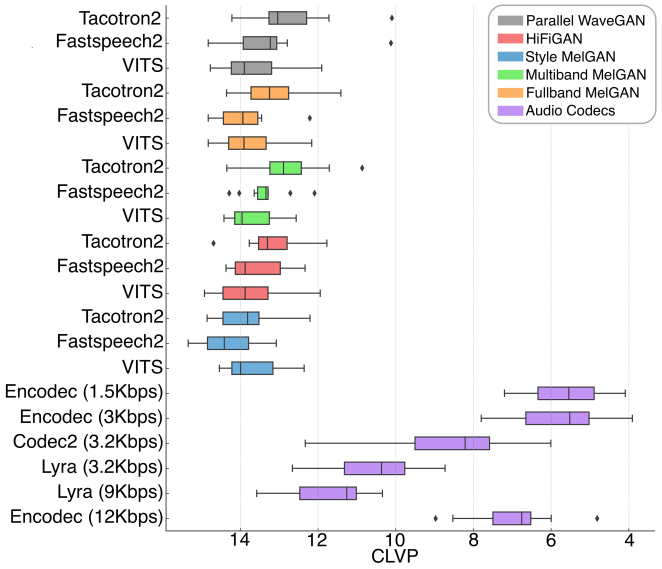

Fig. 4: CLVP Score based on LJ Speech transcripts and synthesized/decoded speech.

improvement is discernible in favor of audio codecs when considering the spread of their distributions.

A crucial point of consideration that we need to make is that the CLVP model is trained on the LibriSpeech [28] and Common Voice datasets [29], followed by fine-tuning on libriTTS [30]. Given that our evaluation uses the LJ Speech dataset, we encounter results that appear counterintuitive, suggesting superior intelligibility over the actual ground truth. This anomaly can be explained by the ideal recording conditions of the LJ Speech dataset and its single-speaker nature. Consequently, certain TTS model pairings might appear to synthesize speech surpassing the original quality.

C. CLVP Score

Figure 4 displays the CLVP score in a descending order on the x-axis, with combinations of TTS models and vocoders, as well as the related audio codecs, on the y-axis. Traditionally, a lower CLVP score should indicate a closer representation of text within the audio according to [4]. However, our results challenge this premise. Audio codecs with lower encoding rates, which would be expected to have greater losses, curiously produce lower CLVP scores. This counterintuitive finding suggests that a higher CLVP score might actually offer a more accurate representation of text in the audio. Furthermore, it is clear that TTS model and vocoder pairings generally outperform audio codecs in fidelity. The standout is the Fastspeech2 model paired with the Style MelGAN vocoder, achieving a CLVP score close to 15, while other TTS models hover between CLVP scores of 13 and 14. Among audio codecs, Lyra with 9 Kbit/s consistently achieves the highest CLVP score, yet it still lags behind the average performance of TTS models.

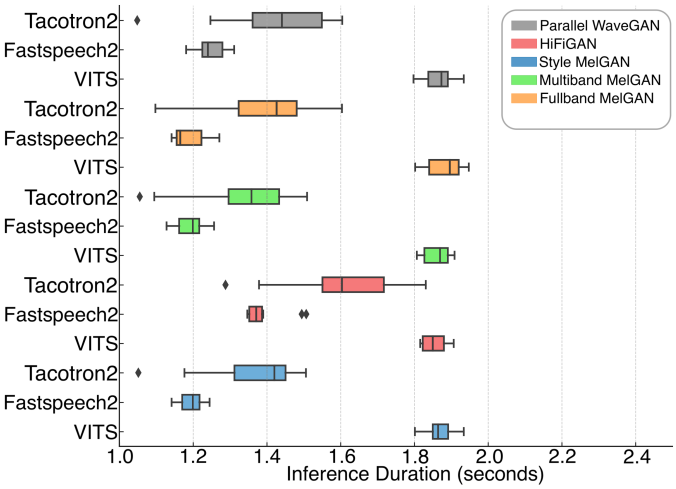

Fig. 5: Inference time required for each TTS model and vocoder combinations.

D. Inference Duration

In assessing hardware performance and computational complexity, we focused on the inference duration required by TTS models to generate speech samples. Compared to the simpler decoding process of audio codecs, TTS models show varying inference durations based on model and vocoder combinations. Figure 5 displays the inference duration, with the x-axis denoting duration in seconds and the y-axis showing model and vocoder combinations. The results were consistent. VITS, which produced more accurate speech samples, took about 300 to 700 ms longer per sample. Interestingly, the choice of vocoder in VITS did not affect the duration. On the other hand, Fastspeech2 generated samples fastest as the name suggests. Among vocoders, HiFiGAN had the longest inference duration for both Tacotron2 and Fastspeech2 correlated with its checkpoint size as shown in Table I, aligning with its higher quality and clarity as well as complexity.

The results further indicate the performance in a bandwidth-

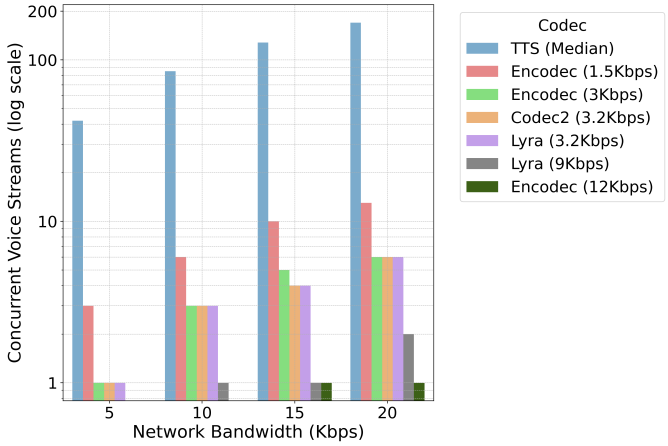

Fig. 6: Concurrent voice streams possible within varying network bandwidth.

constrained network scenario, for instance, a network with a bandwidth of 10 Kbit/s, an audio codec with an encoding rate of 3.2 Kbit/s can facilitate three concurrent real-time voice streams. Link layer protocol can partition encoded real-time voice samples into smaller chunks for transmission over the network. To ensure smoother playback, voice samples can undergo a buffering process at the receiver. The overall latency, including the buffering latency, typically remains below the range of 500 ms, as the evidence shown in an earlier research [31]. In a network with similar resources and configuration, TTS can handle not just three concurrent data streams, but a significantly larger number. We determined the median text size from the LJ Speech dataset samples used for TTS evaluation by counting characters and character memory allocation which is approximately 128 B or 1.024 Kbit. With the median duration of generated TTS voice samples at 8.74 s, we can calculate the encoding rate (ER) using the formula:

$$ER = \frac{Data\,Size}{Duration} \tag{2}$$

Substituting the given values yields:

$$ER = \frac{1.024\,Kbit}{8.74\,s} \approx 0.117\,Kbit/s \tag{3}$$

This translates to eighty-five concurrent transmissions, a significant contrast to the three allowed by audio codecs.

The Figure 6 offers a more detailed understanding of concurrent transmissions in logarithmic scale on the y-axis against diverse network bandwidths on the x-axis. TTS utilization greatly boosts the potential for concurrent transmissions, outpacing other audio codecs by a wide margin. However, it is crucial to recognize that the inference process at the receiver does slightly increase latency compared to traditional audio codecs, resulting in delays between 1.2 s to 1.9 s as shown in Figure 5 [31].

IV. Conclusion and Future Work

In this study, we investigated the efficiency of Text-to-Speech (TTS) models in comparison to traditional audio codecs in low-bandwidth conditions. Utilizing metrics such as Fréchet Distance, Intelligibility, and CLVP scores, as well as inference time, we discerned the performance characteristics of various models, vocoders, and audio codecs in bandwidth-constrained environments. While audio codecs consistently performed well in the Intelligibility Score, TTS models, especially when paired with the appropriate vocoders, demonstrated superior audio clarity as evidenced by metrics like Fréchet Distance, Intelligibility, and CLVP Scores. Notably, VITS emerged as the leading model in terms of audio fidelity, whereas Fastspeech2 excelled in processing speed, as indicated by the inference duration metric. We further investigated the implications of inference duration in resource-constrained networks. In such settings, TTS systems offer efficient resource management, allowing a network to support a higher number of concurrent TTS-generated playbacks, provided the application can tolerate the inherent latency associated with inference.

References

[1] R. Dantas, C. Exton, and A. L. Gear, "Communications using a speech-to-text-to-speech pipeline," in *Wireless and Mobile Computing, Networking and Communications*, Oct. 2019.

[2] V. I. Levenshtein, "Binary codes capable of correcting deletions, insertions and reversals." *Soviet Physics Doklady*, Feb. 1966.

[3] H. Alt and M. Godau, "Computing the Fréchet Distance between two polygonal curves," *Computational Geometry & Applications*, 1995.

[4] J. Betker, "TTS-scores," Apr. 2022. [Online]. Available: https://github.com/neonbjb/tts-scores

[5] J. Kim, J. Kong, and J. Son, "Conditional variational autoencoder with adversarial learning for end-to-end text-to-speech," *ArXiv*, Jun. 2021.

[6] Y. Ren, C. Hu, X. Tan, T. Qin, S. Zhao, Z. Zhao, and T. Liu, "FastSpeech 2: Fast and high-quality end-to-end text to speech," *ArXiv*, Aug. 2022.

[7] K. Tokuda, Y. Nankaku, T. Toda, H. Zen, J. Yamagishi, and K. Oura, "Speech synthesis based on hidden markov models," *IEEE*, Apr. 2013.

[8] N. Xue, "Chinese word segmentation as character tagging," *Computational Linguistics & Chinese Language Processing*, Feb. 2003.

[9] K. Yao and G. Zweig, "Sequence-to-sequence neural net models for grapheme-to-phoneme conversion," *CoRR*, Aug. 2015.

[10] X. Tan, T. Qin, F. Soong, and T. Liu, "A survey on neural speech synthesis," *ArXiv*, Jul. 2021.

[11] J. Shen, R. Pang, R. J. Weiss, M. Schuster, N. Jaitly, Z. Yang, Z. Chen, Y. Zhang, Y. Wang, R. Skerrv-Ryan, R. A. Saurous, Y. Agiomvrgiannakis, and Y. Wu, "Natural tts synthesis by conditioning wavenet on MEL spectrogram predictions," in *Acoustics, Speech and Signal Processing*, Feb. 2018.

[12] 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*, Sep. 2016.

[13] D. Rezende and S. Mohamed, "Variational inference with normalizing flows," in *Conference on Machine Learning*, Jul. 2015.

[14] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial networks," *arXiv*, Dec. 2014.

[15] "Codec 2," Oct. 2023. [Online]. Available: http://www.rowetel.com/?page_id=452

[16] "SoundStream: An End-to-End Neural Audio Codec," Oct. 2023. [Online]. Available: https://blog.research.google/2021/08/soundstream-end-to-end-neural-audio.html

[17] A. Défossez, J. Copet, G. Synnaeve, and Y. Adi, "High fidelity neural audio compression," *ArXiv*, Oct. 2022.

[18] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever, "Learning transferable visual models from natural language supervision," *ArXiv*, Feb. 2021.

[19] K. Ito and L. Johnson, "The LJ Speech dataset," 2017. [Online]. Available: https://keithito.com/LJ-Speech-Dataset/

[20] K. Åhlander, "Einstein summation for multidimensional arrays," *Computers & Mathematics with Applications*, Oct. 2002.

[21] S. Schneider, A. Baevski, R. Collobert, and M. Auli, "wav2vec: Unsupervised pre-training for speech recognition," *ArXiv*, Sep. 2019.

[22] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, "Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural networks," in *Conference on Machine Learning*, Jun. 2006.

[23] T. Hayashi, R. Yamamoto, T. Yoshimura, P. Wu, J. Shi, T. Saeki, Y. Ju, Y. Yasuda, S. Takamichi, and S. Watanabe, "Espnet2-tts: Extending the edge of tts research," *ArXiv*, Oct. 2021.

[24] R. Yamamoto, E. Song, and J. Kim, "Parallel WaveGAN: A fast waveform generation model based on generative adversarial networks with multi-resolution spectrogram," *ArXiv*, Feb. 2020.

[25] J. Kong, J. Kim, and J. Bae, "HiFi-GAN: Generative adversarial networks for efficient and high fidelity speech synthesis," *ArXiv*, Oct. 2020.

[26] A. Mustafa, N. Pia, and G. Fuchs, "StyleMelGAN: An efficient high-fidelity adversarial vocoder with temporal adaptive normalization," *ArXiv*, Feb. 2021.

[27] G. Yang, S. Yang, K. Liu, P. Fang, W. Chen, and L. Xie, "Multi-band MelGAN: Faster waveform generation for high-quality text-to-speech," *ArXiv*, Nov. 2020.

[28] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: An asr corpus based on public domain audio books," in *Acoustics, Speech and Signal Processing*, Apr. 2015.

[29] R. Ardila, M. Branson, K. Davis, M. Henretty, M. Kohler, J. Meyer, R. Morais, L. Saunders, F. M. Tyers, and G. Weber, "Common voice: A massively-multilingual speech corpus," in *Language Resources and Evaluation*, Mar. 2020.

[30] H. Zen, R. Clark, R. J. Weiss, V. Dang, Y. Jia, Y. Wu, Y. Zhang, and Z. Chen, "LibriTTS: A corpus derived from librispeech for text-to-speech," *arXiv*, Apr. 2019.

[31] B. Mekiker, M. Wittie, J. Jones, and M. Monaghan, "Beartooth relay protocol: Supporting real-time application streams with dynamically allocated data reservations over lora," in *Computer Communications and Networks (ICCCN)*, Aug. 2021.