5

Multichannel Speech Enhancement under Low-Latency Constraints: Balancing Quality and Computational Cost

Zahra Benslimane[1], Fabrice Auzanneau[1], Martyna Poreba[1], Michal Szczepanski[1], Fabian Chersi[1], and Romain Serizel[2]

[1]Université Paris-Saclay, CEA-List, France
[2]Université de Lorraine, INRIA, LORIA, France

Abstract

Multichannel speech enhancement is essential for robust speech intelligibility in noisy environments. Although many algorithms address this challenge, their deployment in real-time or embedded settings imposes strict computational and latency constraints. This work compares three representative methods: FaSNet-TAC, an end-to-end neural network-based approach; neural MVDR, a hybrid time-frequency method; and Tango, a fully distributed framework. For our evaluation, we employ a methodology that explicitly separates the input context needed for optimal performance from the output latency. Using a common noisy-speech test set and identical evaluation protocols, we assess the enhancement quality of these methods and their computational complexity under various target algorithmic latency requirements. Our analysis shows that a one second context is sufficient to approach full-utterance performance. It also highlights distinct trade-offs across architectures: Tango exhibits the most robust behaviour under stringent latency constraints, whereas FaSNet-TAC is the most sensitive to context and hop-size reduction. We also find that mask-estimation stages constitute the dominant computational bottleneck, underscoring the importance of

aligning model design with the intended latency regime for practical real-time deployment.

Keywords: Multichannel speech enhancement, Low-latency audio processing, Computational complexity, Neural beamforming.

5.1 Introduction and Background

Speech Enhancement (SE) aims to improve the perceptual quality and intelligibility of audio signals by reducing the impact of background noise, interference, and reverberation. While single-channel methods rely solely on spectral and temporal cues [1, 2], multichannel SE techniques leverage the spatial information provided by microphone arrays to better distinguish between speech and noise sources for more effective source separation and noise suppression.

Recent progress in machine learning has resulted in the development of new end-to-end neural network-based algorithms for enhancing multichannel speech [3–6]. These algorithms model complex relationships within the data and can learn to map time-domain or time-frequency representations of noisy speech directly to enhanced outputs, capturing rich spatiotemporal dependencies without relying on handcrafted feature extractors. Alongside these purely data-driven approaches, hybrid frameworks that embed classical signal-processing methods into neural pipelines have been explored in an attempt to find a balance between robustness, interpretability, and computational efficiency [7]. In these systems, deep neural networks (NN) are trained to estimate statistics that are used to compute filters such as beamformers [8].

Although these developments have led to substantial improvements in enhancement quality, practical deployment imposes additional constraints that traditional metrics fail to capture. Measures such as SNR, while informative, overlook factors critical for real-time and embedded operation. Consequently, a meaningful assessment of modern SE systems must also account for latency and computational efficiency, considerations that have driven a growing body of work on model compression and low-complexity designs.

To enable on-device SE, researchers have used pruning, quantization, and compact designs to reduce model size and inference cost. Wu and Yu (2019) [9] removed redundant channels and applied weight clustering. Tan and Wang (2021) [10] used sparse regularization and iterative pruning. Fedorov et al. (2020) [11] introduced TinyLSTMs with pruning, 8-bit

quantization, and state-update skipping. Stamenovic et al. (2021) [12] compared different sparsity methods. More recently, Cohen et al. (2024) [13] proposed a Fully Quantized SE model with a quantization-aware knowledge-distillation loss for negligible Signal to Distortion Ratio (SDR). However, low computational load does not necessarily correlate with a decreased latency. Recent studies have focused on minimizing algorithmic latency in SE to satisfy the stringent requirements of real-time applications. Various strategies have been proposed, including time-domain end-to-end models, Short Term Fourier Transform-domain (STFT) processing with dual windowing schemes [14], and decoupled spatial-temporal architectures [15]. Other approaches used spiking neural networks [16] or causal filter-and-sum frameworks with low-bit-rate inter-device communication for binaural setups [17].

In this paper, we explore, evaluate, and compare three deep learning-based algorithms for multichannel SE, with a focus on their performance under the stringent constraints of real-time embedded deployment. Our goal is to systematically analyse the trade-offs inherent to each of these models, focusing in particular on the relationship between input context size and output latency, as well as their computational load.

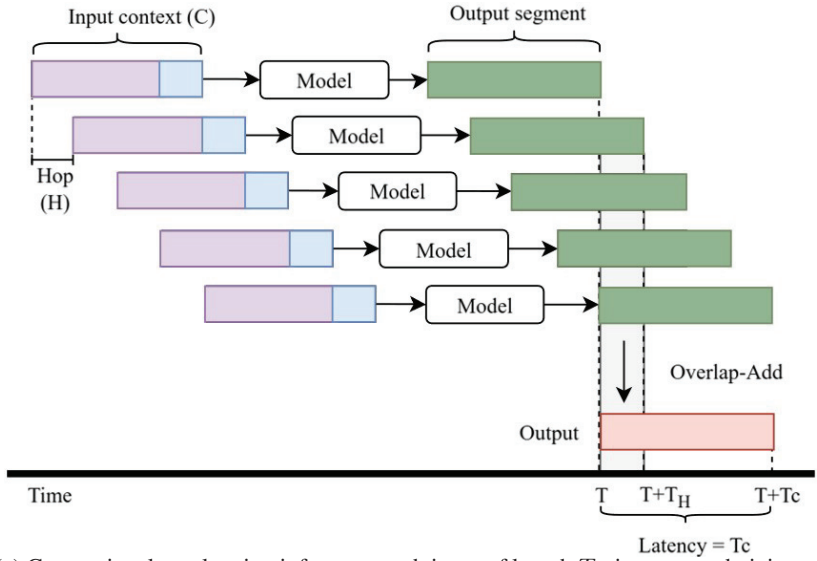
5.2 Methodology

5.2.1 Latency reduction strategy

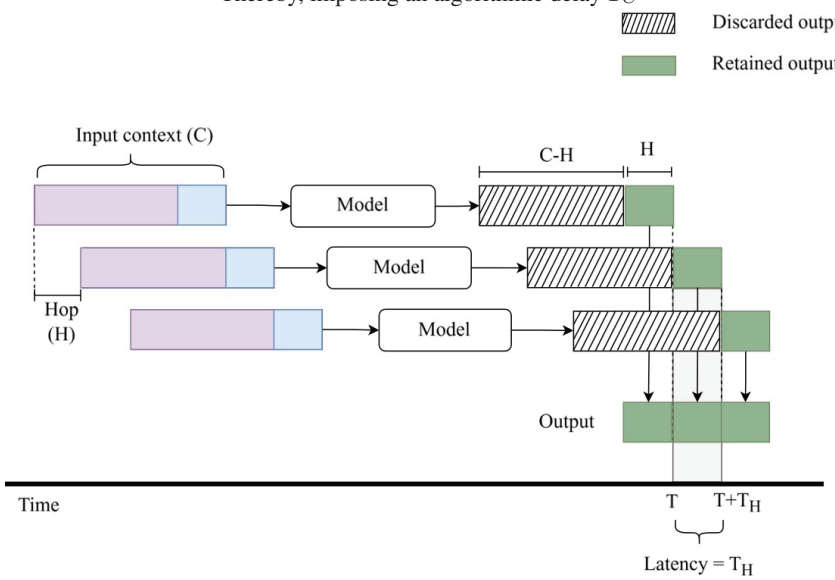
Real-time SE presents a fundamental trade-off between using a large input context (long audio segments) and achieving low output latency. A larger context provides the model with more information (e.g., speaker characteristics, noise patterns), which generally improves enhancement quality. However, it also forces the system to wait longer before producing any output, increasing the processing delay.

We distinguish two types of delay*: algorithmic latency*, the time required to accumulate sufficient audio data before processing, and *computational latency,* the time the algorithm itself spends performing calculations, which can vary depending on the device's processing power and resources. To keep this study hardware-agnostic, we assume that computational latency is negligible.

In contrast to the traditional STFT symmetric windows (Figure 5.1-a), Wang et al. [14] proposed a low-latency STFT framework that employs asymmetric windows for both analysis and reconstruction. Building on their approach, we conducted experiments to adapt their method for processing input context under various algorithmic-latency constraints. To achieve this,



(a) Conventional overlapping inference: each input of length T_C is processed giving an output of the same length. Overlap-add is used to reconstruct the continuous signal. Thereby, imposing an algorithmic delay T_C



(b) Low latency inference

Figure 5.1 Comparison of two overlapping-windows inference strategies. In (a), the model incurs an algorithmic latency of T_C . In (b), by discarding overlapping outputs and retaining only the last T_H seconds of each pass, the perceived latency is reduced to T_H.

we maintained a rolling input buffer of length C which was advanced in small hops of size H, where $\mathrm{H} < \mathrm{C}$. At each hop, the buffer window was passed through the model, producing an output window $y_k \in \mathbb{R}^C$ from which only the final H samples, $\mathrm{yk}\,[\mathrm{C} - \mathrm{H} : \mathrm{C} - 1]$ were retained. Here, C and H denote the sequence length and hop size in the model's internal representation. These could represent raw waveform samples, STFT frames, or any other internal representation. Their corresponding durations in seconds are denoted as T_C and T_H. The model consistently utilized the full context C, thereby maximizing enhancement performance while producing new audio samples every T_H seconds, as illustrated in Figure 5.1-b. As the hop size H was reduced, the model had to be invoked more frequently, increasing computational demand. To assess this load in a hardware-agnostic manner, we estimate the number of floating-point operations per second (FLOPs/s).

5.2.2 Overview of the selected algorithms

We selected three representative multichannel SE architectures (Figure 5.2), each reflecting a distinct modelling paradigm, in order to evaluate them under identical conditions. FaSNet-DPRNN-TAC [5] is a time-domain, end-to-end Neural Network operating on raw waveforms. For simplicity, we refer to it as FaSNet-TAC throughout this document. It builds upon the original FaSNet architecture [3] and incorporates dual-path RNN (DPRNN) blocks [4] to model inter-channel dependencies. Each DPRNN block incorporates a Transform Average-Concatenate (TAC) module that projects all channels features into a shared embedding space to produce an output that is invariant to both the number and ordering of microphones.

Neural MVDR [7], is a hybrid time-frequency domain architecture that integrates a neural implementation of the MVDR beamformer. Spatial filters are computed based on the Power Spectral Density (PSD) matrices of both speech and noise, which are estimated using time-frequency masks generated by an LSTM-based neural network. This allows the system to dynamically adapt the beamforming process to the acoustic scene, improving noise suppression while preserving speech integrity.

Finally, Tango [8], is a fully distributed, two-stage hybrid SE framework designed to operate on spatially unconstrained microphone arrays with multiple nodes. Each processing stage combines a CNN along with a Speech Distortion Weighted Multichannel Wiener Filter, a modified version of the traditional MWF [18], which provides an improved balance between noise suppression and speech distortion.

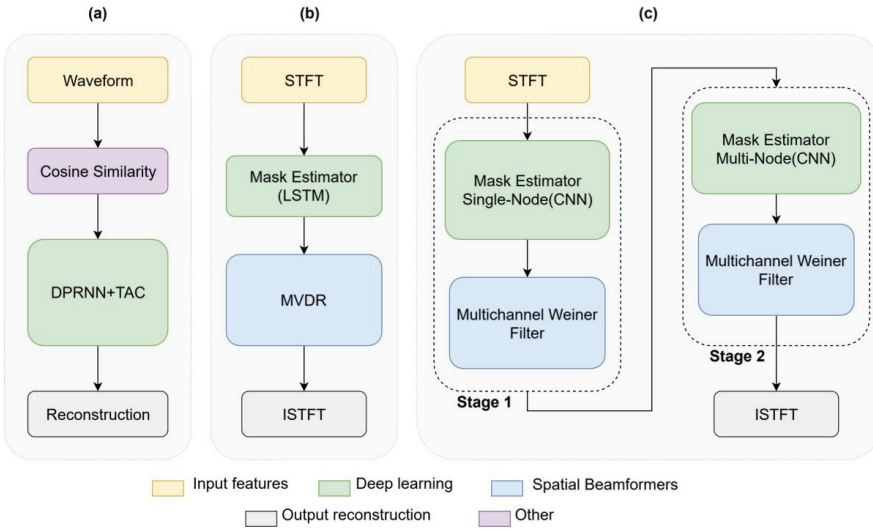


Figure 5.2 Studied models. (a) FaSNet-TAC (b) Neural MVDR (c) Tango.

5.3 Experimental Setup

5.3.1 Testing set

The testing set was created using 1,000 utterances drawn from the LibriSpeech test-clean corpus [19], sampled at 16 kHz, with durations ranging between 3 s and 11 s. Each clean utterance was spatialized by convolution with measured binaural Room Impulse Responses (RIRs) from the Binaurec database, using a speech source positioned at $0°$ and noise sources at $+45°$ and $+90°$. The RIRs were recorded with a dummy head in a room measuring 6.62 m $\times$ 2.57 m $\times$ 2.60 m with a reverberation time (RT60) of 0.20 s [20]. The spatialized clean and noise signals were then mixed at three different signal-to-noise ratios (SNRs) of -5 dB, 0 dB, and $+5$ dB. Two types of noise were considered: speech-shaped noise and white noise.

5.3.2 Implementation details

The three models were trained on a synthetic dataset, described in [21]. Clean speech signals were drawn from the LibriSpeech corpus and combined with two categories of noise: speech-shaped noise (30%) and environmental recordings from the Disco noise collection (70%). The speech-shaped component was created using LibriSpeech speakers excluded from the clean set to

maintain spectral similarity between the speech and noise. To simulate realistic reverberant conditions, room impulse responses were generated using the *Pyroomacoustics* library [22]. The simulated rooms had lengths between 3 and 8 m, widths between 3 and 5 m, and heights between 2.5 and 3 m. Reverberation times (RT60) were randomly sampled in the range of 0.15-0.4 s to represent a variety of acoustic environments.

A four-microphone binaural array was simulated to approximate a hearing-aid configuration. For each simulated array the interaural spacing (ear-to-ear center distance) was sampled uniformly between 0.12 m and 0.18 m. Each ear included two microphones positioned with small lateral offsets of 0.01-0.02 m and vertical offsets of 0.01-0.015 m. Speech and noise sources were randomly placed within the virtual room, and mixtures were generated with signal-to-interference ratios (SIRs) uniformly distributed between -10 dB and $+10$ dB.

During inference, Tango is configured to produce a single enhanced signal for each ear, resulting in binaural outputs. In contrast, the FaSNet-TAC and Neural MVDR were applied twice; once with the front-left channel and once with the front-right channel as the processing reference channel, yielding corresponding left and right enhanced signals.

5.3.3 Evaluation metrics

Speech enhancement performance was evaluated using the scale-invariant metrics: Scale-Invariant Signal-to-Distortion Ratio (SI-SDR), Scale-Invariant Signal-to-Interference Ratio (SI-SIR), and Scale-Invariant Signal-to-Artifact Ratio (SI-SAR), which respectively assess overall distortion, interference suppression, and artifact introduction.

Along with these enhancement metrics, we used the Python libraries *torchinfo* and *ptflops* to calculate the FLOPs required to process 1 second of audio for the deep learning components of each neural architecture presented in Section 5.2.2. For the beamforming components, where automated profiling was not feasible, FLOPs were computed manually.

5.3.4 Context and latency trade-off

We first determined the minimal acceptable context length $C_{\min}$, through a non-overlapping block segmentation experiment. Each input recording was divided into contiguous, disjoint windows of fixed length C, with the network processing each window independently. We tested context durations T_C of

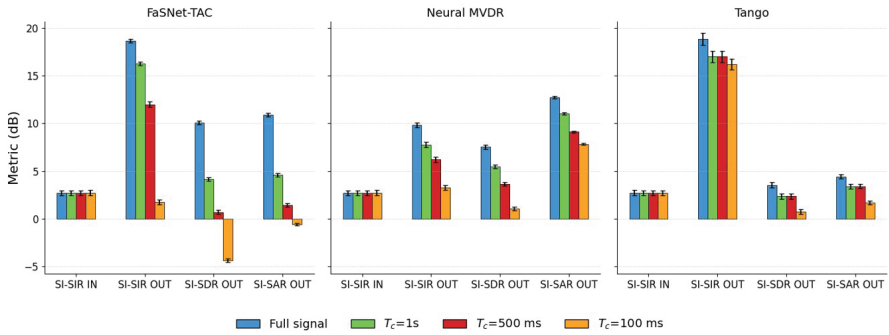
1 s, 500 ms, and 100 ms. For FaSNet-TAC, each window was fed directly into the network. For the two-hybrid mask-beamforming architectures, the same segmentation was applied to both the mask estimation and beamforming stages, enabling the entire pipeline to operate segment-wise. This setup allowed us to assess how increasing C influenced enhancement quality in the absence of overlapping segments.

After identifying C_{min} (the smallest context length that yields near-peak performance), we reduced the effective algorithmic latency using an overlapping-segment inference strategy, as described in Section 5.2.1 and shown in Figure 5.1-b. In this approach, the segment length is fixed at C_{min}, while the processing window advances by a smaller hop size H, such that $H < C_{min}$. We evaluated hop sizes of 500 ms, 100 ms, 50 ms, and 10 ms. For example, with $T_C = 1\ s$ and $T_H = 50\ ms$, the network processes a 1 s segment and outputs the final 50 ms of enhanced audio before shifting the input window forward by 50 ms and repeating the process.

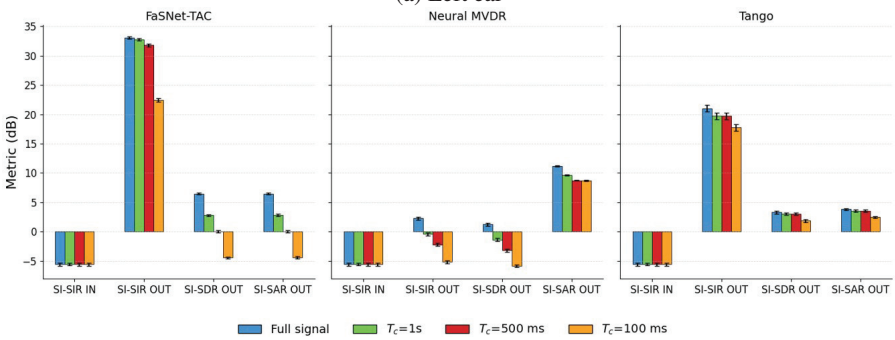
5.4 Results and discussion

Note that all metrics denoted with the "out" suffix correspond to evaluations performed on the enhanced signals obtained after the speech enhancement stage. Additionally, the SI-SIR IN metric, computed on the input mixtures, is included to quantify the suppression improvement provided by each model. The "in" and "out" designations are used only in the figures for clarity. Unless explicitly stated otherwise, all mentions of SI-SIR, SI-SDR, and SI-SAR in the text refer to the output metrics.

Figure 5.3 shows SI-SIR, SI-SDR, and SI-SAR for non-overlapping segments of varying input lengths, with the leftmost blue bar indicating full-utterance evaluation: Neural MVDR exhibits a sharp interaural difference, in terms of SI-SIR, compared to the rest, which is expected given the spatial configuration of the test set. In our setup, the dominant noise sources were located at $+45^\circ$ and $+90^\circ$ azimuths relative to the listener's head, *i.e.*, closer to the right ear. Consequently, the right-channel mixtures inherently contain stronger interference energy and a less favourable SNR prior to enhancement, creating a more challenging separation problem on that side. In contrast, Tango demonstrates the most consistent SI-SIR between the left and right ears, indicating robust interference suppression across spatial channels. Interestingly, FaSNet-TAC achieves slightly higher SI-SIR on the right ear, suggesting that its end-to-end time-domain processing can better exploit interaural cues under asymmetric noise conditions. However, this



(a) Left ear



(b) Right ear

Figure 5.3 Effect of input context length on enhancement performance on the left (a) and right (b) ear.

advantage is not reflected in the other metrics; both metrics remain lower overall, indicating that the additional SIR gain comes at the cost of higher distortion and artifact introduction.

Reducing the input context window from 1 s to 100 ms causes a steady SDR decline, highlighting the importance of temporal information. This behaviour depends on how each architecture represents or aggregates temporal context. For example, Tango's convolutional mask estimator works on fixed 128 ms segments at a time, so any window larger than 128 ms leaves its mask predictions unaffected by context reduction. Consequently, the observed degradation in Tango can be attributed primarily to the downstream PSD matrices estimation segmentation, leaving it the least affected by shorter windows among the evaluated models.

Neural MVDR's performance depends on both the LSTM-based mask estimator and the speech-and-noise covariance matrices, so errors accumulate. To isolate each component's impact under reduced context, we

ran two experiments: one varying only the mask-estimator's input window (with full-utterance context for PSD matrices estimation and beamforming; Figure 5.4), and one varying only the PSD matrices input window (with full-utterance context for mask estimation; Figure 5.5). The results show that segmentation of the PSD estimation stage causes a markedly larger performance drop, even when the context is just reduced to 1s, compared to the neural-network segmentation. This indicates that accurate and temporally coherent covariance estimates are critical for maintaining spatial filtering stability.

Our preliminary experiments demonstrate that a 1s context window achieves near full-signal performance; therefore, in order to ensure optimal SE results, we set $T_c = 1s$ for all subsequent evaluations, despite the associated increase in computational cost this may entail. Extending the context beyond 1 s was not considered, as the additional temporal information would likely yield diminishing returns while substantially increasing computational demand and latency. Alternatively, other scenarios could justifiably opt for shorter context windows to reduce resource consumption, at the expense of a

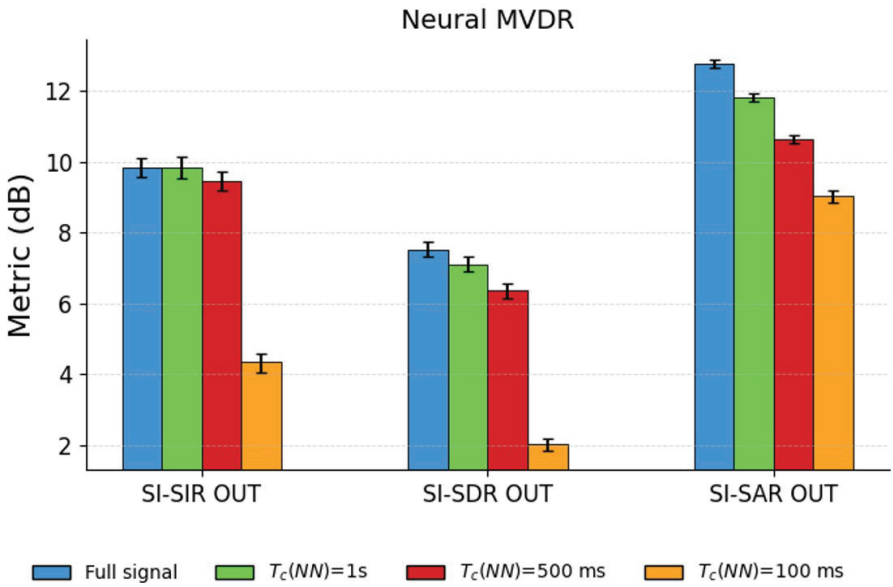


Figure 5.4 Effect of mask estimator input context length on the neural MVDR performance (with a full-utterance PSD matrices context) on left ear.

higher drop in enhancement quality, but these cases fall outside the scope of this paper.

The next experiment (Figure 5.6) keeps constant input context, allowing us to isolate the effect of varying hop size and consequently the perceived latency, on each model's behaviour. Across both ears, Tango remains the most stable, showing only minimal degradation even at a 10 ms hop. Its consistent left-right performance reflects its binaural design, which explicitly models interaural cues. Neural MVDR exhibits moderate sensitivity to latency, with performance gradually declining, but it preserves relatively similar trends between the two ears. In contrast, FaSNet-TAC shows the steepest performance drop, particularly in SI-SDR and SI-SAR, starting from 500 ms latency. It also displays more pronounced disparities between the left and right ears, which is expected given its monaural end-to-end formulation and the noise conditions of the test set.

Table 5.1 reports the estimated floating-point operations per second (FLOPs/s) for each major processing component of our three architectures, evaluated at the four different perceived latencies. As the hop size shrinks, the model must be invoked more frequently, thus, the overall computational cost increases linearly: for example, FaSNet-TAC's DPRNN+TAC block jumps from about 2.5 GFLOPs/s at the 1 s baseline to 250 GFLOPs/s at a 10 ms hop. In both Neural MVDR and Tango, the neural mask-estimation stage incurs by far the largest share of computation. The PSD matrices estimation and beamforming steps remain in the low-mega to single-giga FLOPs/s

Table 5.1 GFLOPs/s of different processing components at various hop sizes (perceived latencies)

		Hop size / Perceived latency			
Component	**Base**	**500ms**	**100ms**	**50ms**	**10ms**
	Neural MVDR				
Mask estimation	13.5	27.0	135	270	1350
Speech & Noise PSD matrices	0.009	0.018	0.094	0.18	0.94
MVDR Beamformer	0.001	0.002	0.012	0.024	0.60
	Tango (Step 1 + Step 2)				
Mask estimation	3.14	6.28	31.4	62.8	310
Speech & Noise PSD matrices	0.008	0.017	0.086	0.17	0.80
MCWF Beamformer	0.002	0.004	0.02	0.04	0.50
	FaSNet-TAC				
DPRNN+TAC	2.5	5	25	50	250

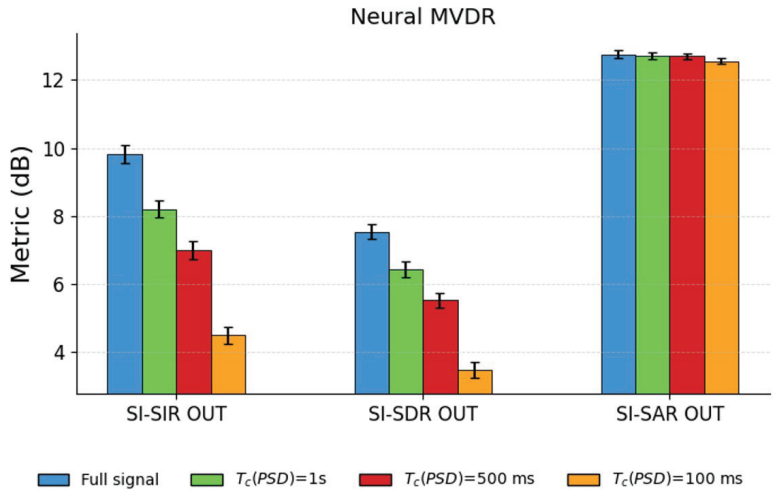
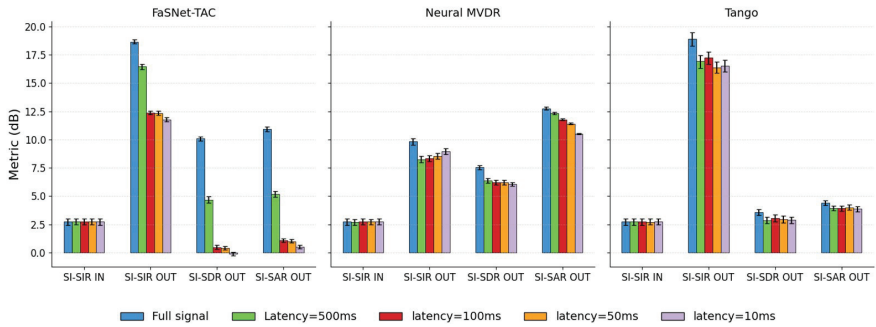
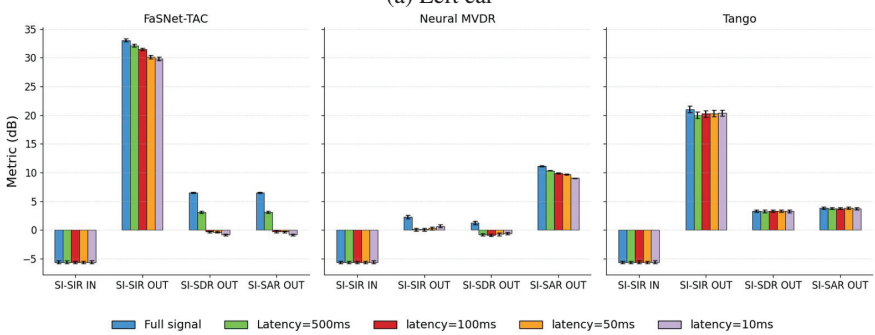


Figure 5.5 Effect of PSD matrices input context on the neural MVDR (with a full-utterance mask estimation context) on left ear.



(a) Left ear



(b) Right ear

Figure 5.6 Performance comparison of the three models at varying latency with a 1s input context of the left (a) and right (b) ear

regime, even at 10 ms latency, confirming that mask estimation is the primary computational bottleneck when targeting very low latencies.

5.5 Conclusion

This study evaluated three multichannel SE models; FaSNet-TAC, Neural MVDR, and Tango, to examine how architectural design influences the trade-offs between enhancement quality, latency, and computational efficiency. The results highlight that different modelling paradigms exhibit distinct sensitivities to context reduction and latency constraints. End-to-end time-domain approaches show strong efficiency but higher sensitivity to reduced context, while hybrid methods demonstrate greater robustness under varying inference conditions. The overlapping-segment scheme therefore offers a flexible framework for adapting real-time SE to the latency and computational constraints of embedded hardware platforms. However, sustaining full-context performance at sub-100 ms latencies on resource-constrained devices becomes prohibitively expensive if one relies solely on brute-force windowing and frequent model inferences.

Overall, these analyses show that the overlapping-segment strategy provides a practical means to manage the trade-off between computational cost and real-time responsiveness. As the hop size decreases, all three models incur a substantial increase in FLOPs/s, driven almost entirely by the frequency of mask-estimation inference. This emphasizes the importance of aligning model design with the intended deployment scenario and latency requirements. Future work will investigate complementary approaches, such as streaming network architectures along with model-compression techniques that reduce FLOPs and end-to-end latency.

Acknowledgements

This project has received funding from the French National Research Agency (ANR) under the project REFINED – “REal-time artiFicial INtelligence for hEaring aiDs” (ANR-21-CE19-0043).

References

[1] C. Zheng et al., “Sixty Years of Frequency-Domain Monaural Speech Enhancement: From Traditional to Deep Learning Methods,” Trends in

Hearing, vol. 27, Jan. 2023, doi: https://doi.org/10.1177/23312165231209913.

[2] D. O'Shaughnessy, "Speech Enhancement—A Review of Modern Methods," IEEE Transactions on Human-Machine Systems, vol. 54, no. 1, pp. 110–120, Jan. 2024, doi: https://doi.org/10.1109/thms.2023.3339663.

[3] Y. Luo, C. Han, Nima Mesgarani, Enea Ceolini, and S.-C. Liu, "FaSNet: Low-Latency Adaptive Beamforming for Multi-Microphone Audio Processing," 2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU), Dec. 2019, doi: https://doi.org/10.1109/asru46091.2019.9003849.

[4] Y. Luo, Z. Chen, and T. Yoshioka, "Dual-Path RNN: Efficient Long Sequence Modeling for Time-Domain Single-Channel Speech Separation," International Conference on Acoustics, Speech, and Signal Processing, May 2020, doi: https://doi.org/10.1109/icassp40776.2020.9054266.

[5] Y. Luo, Z. Chen, Nima Mesgarani, and T. Yoshioka, "End-to-end Microphone Permutation and Number Invariant Multi-channel Speech Separation," Apr. 2020, doi: https://doi.org/10.1109/icassp40776.2020.9054177.

[6] D. Lee, S. Kim, and J.-W. Choi, "Inter-channel Conv-TasNet for multi-channel speech enhancement," arXiv.org, 2021. https://arxiv.org/abs/2111.04312.

[7] J. Heymann, L. Drude, and Reinhold Haeb-Umbach, "Neural network based spectral mask estimation for acoustic beamforming," International Conference on Acoustics, Speech, and Signal Processing, Mar. 2016, doi: https://doi.org/10.1109/icassp.2016.7471664.

[8] N. Furnon, Romain Serizel, I. Illina, and Slim Essid, "DNN-based Distributed Multichannel Mask Estimation for Speech Enhancement in Microphone Arrays," pp. 4672–4676, Apr. 2020, doi: https://doi.org/10.1109/icassp40776.2020.9054643.

[9] J.-Y. Wu, C. Yu, S.-W. Fu, C.-T. Liu, S.-Y. Chien, and Y. Tsao, "Increasing Compactness of Deep Learning Based Speech Enhancement Models With Parameter Pruning and Quantization Techniques," IEEE Signal Processing Letters, vol. 26, no. 12, pp. 1887–1891, Dec. 2019, doi: https://doi.org/10.1109/lsp.2019.2951950.

[10] K. Tan and D. Wang, "Towards Model Compression for Deep Learning Based Speech Enhancement," IEEE/ACM Transactions on Audio,

Speech, and Language Processing, vol. 29, pp. 1785–1794, 2021, doi: https://doi.org/10.1109/taslp.2021.3082282.

[11] I. Fedorov et al., "TinyLSTMs: Efficient Neural Speech Enhancement for Hearing Aids," Interspeech 2020, Oct. 2020, doi: https://doi.org/10.21437/interspeech.2020-1864.

[12] M. Stamenovic, Westhausen, Nils L, L.-C. Yang, C. Jensen, and A. Pawlicki, "Weight, Block or Unit? Exploring Sparsity Tradeoffs for Speech Enhancement on Tiny Neural Accelerators," arXiv.org, 2021. https://arxiv.org/abs/2111.02351.

[13] E. Cohen, Hai Victor Habi, and A. Netzer, "Towards Fully Quantized Neural Networks For Speech Enhancement," Aug. 2023, doi: https://doi.org/10.21437/interspeech.2023-883.

[14] Z.-Q. Wang, G. Wichern, S. Watanabe, and J. Le Roux, "STFT-Domain Neural Speech Enhancement With Very Low Algorithmic Latency," IEEE/ACM Transactions on Audio, Speech, and Language Processing, vol. 31, pp. 397–410, 2023, doi: https://doi.org/10.1109/taslp.2022.3224285.

[15] A. Pandey and B. Xu, "Decoupled Spatial and Temporal Processing for Resource Efficient Multichannel Speech Enhancement," pp. 12206–12210, Mar. 2024, doi: https://doi.org/10.1109/icassp48485.2024.10446087.

[16] T. Sun and S. Bohté, "DPSNN: Spiking Neural Network for Low-Latency Streaming Speech Enhancement," arXiv.org, 2024. https://arxiv.org/abs/2408.07388.

[17] N. L. Westhausen and B. T. Meyer, "Low Bit Rate Binaural Link for Improved Ultra Low-Latency Low-Complexity Multichannel Speech Enhancement in Hearing Aids," 2023 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA), pp. 1–5, Oct. 2023, doi: https://doi.org/10.1109/waspaa58266.2023.10248154.

[18] S. Doclo, A. Spriet, J. Wouters, and M. Moonen, "Frequency-domain criterion for the speech distortion weighted multichannel Wiener filter for robust noise reduction," Speech Communication, vol. 49, no. 7–8, pp. 636–656, Jul. 2007, doi: https://doi.org/10.1016/j.specom.2007.02.001.

[19] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: An ASR corpus based on public domain audio books," 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Apr. 2015, doi: https://doi.org/10.1109/icassp.2015.7178964.

[20] L. Delebecque and Romain Serizel, "BinauRec: A dataset to test the influence of the use of room impulse responses on binaural speech enhancement," HAL (Le Centre pour la Communication Scientifique Directe), pp. 126–130, Sep. 2023, doi: https://doi.org/10.23919/eusipco58844.2023.10289772.

[21] N.-E. Monir, P. Magron, and R. Serizel, "Frequency-Weighted Training Losses for Phoneme-Level DNN-based Speech Enhancement," arXiv.org, 2025. https://arxiv.org/abs/2506.18714

[22] R. Scheibler, E. Bezzam, and I. Dokmanić, "Pyroomacoustics: A Python Package for Audio Room Simulation and Array Processing Algorithms," IEEE Xplore, Apr. 01, 2018. https://ieeexplore.ieee.org/abstract/document/8461310?casa_token=u0fNFSqYAs4AAAAA:4-mLzpbwKd5EAOyAuEK-0OLSwxV6H7vHedb3jzkAnJsGyCqFt4irxzx6NBpHh2JeQrLQP4i.