When Learning Meets the Channel: Edge AI through Split and Semantic Design
Part I

Vukan Ninković

October 28, 2025

About Me...

Vukan Ninkovic, PhD

Research Associate / Postdoc

The Institute for Artificial Intelligence Research and Development of Serbia / University of Novi Sad

Research Interests

- Split Learning, Edge AI, UAV-assisted IoT
- Semantic Communications, AE-based Coding
- Beyond 5G Wireless Communication Systems

Contact

vukan.ninkovic@ivi.ac.rs

ninkovic@uns.ac.rs

Vukan Ninkovic

Overview

1. Introduction & Motivation

2. Theoretical Background

3. Split Learning with Channel Integration

4. Performance Evaluation

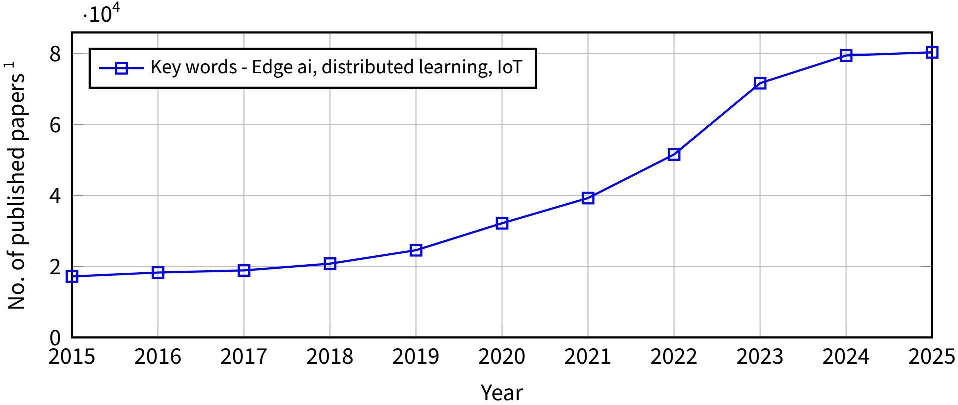
5. (Instead of) Conclusion

6. Appendix: More Explanations and Interesting Results…

Introduction & Motivation

Introduction

- Artificial Intelligence (AI) and Internet of Things (IoT) convergence:
  - Inference, reasoning, and decision-making closer to the data sources.
  - Optimization of resource allocation - Cloud $\rightarrow$ Edge.
  - Benefits - Reduce the communication pressure and latency, faster response time...



[1] Source: Google Scholar (https://scholar.google.com/)

Motivation

- Problem - Integrating DL-based inference on resource-constrained IoT edge devices.
  - Memory, computational capabilities...
- Solution - Distributed inference approaches:
  - Collaborative DL model execution across IoT devices.
  - Reduce dependence on cloud resources and enhance data privacy.
- Split learning (SL) - Training workload is delegated between edge nodes and servers:
  - Raw data locally stored.
  - **Problem** - Significant challenges due to unpredictable wireless channel conditions.
  - **Solution** - Channel integration into training pipeline.

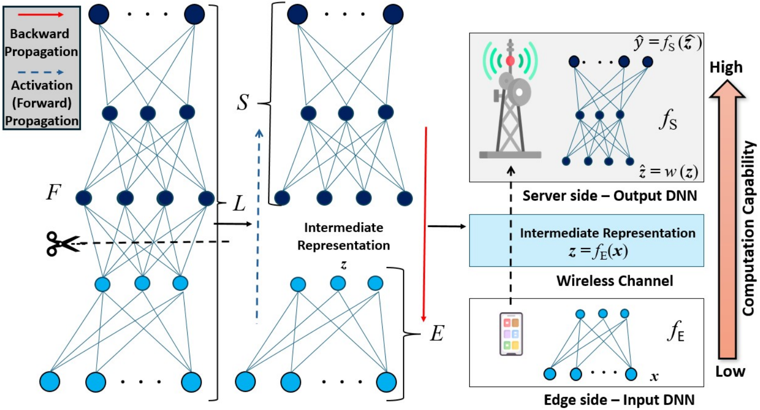
Related Work

Paper	Comm.–aware SL approach	Erasure channel	AWGN channel	Early–exit	Heterogeneous setup	Time series
[1, 2, 3]					✓	
[4]	✓	✓				
[5, 6]	✓		✓			
[7]	✓		✓			
[8]	✓		✓	✓		
[9, 10]	✓			✓		
[11]	✓		✓		✓	
[12, 13, 14, 15]	✓				✓	
[16]	✓					✓
[17]					✓	✓
COMSPLIT	✓	✓	✓	✓	✓	✓

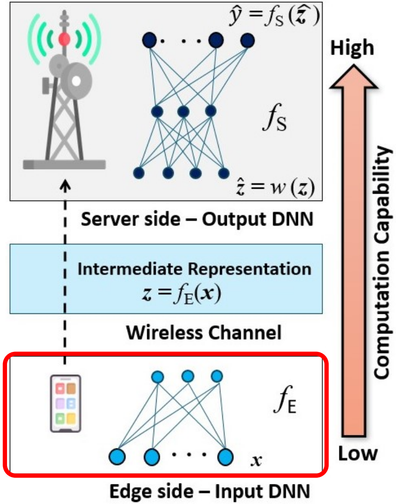
Theoretical Background

Split Learning & Inference

- Total L layers:
  - Edge side - E
  - Server side - S
  - $L = E + S \ (E < S)$
- $F = f_{\mathrm{E}} \circ f_{\mathrm{S}}$



Split Learning - (High) Mathematical Formulation

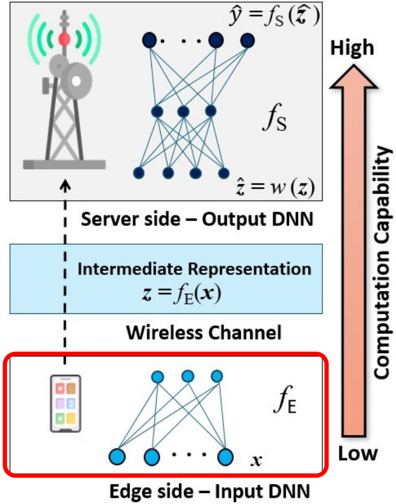


$\boldsymbol{x} \in \mathbb{R}^N$

Split Learning - (High) Mathematical Formulation

$$f_E : \mathbb{R}^N \to \mathbb{R}^n, \quad n < N$$

$$\boldsymbol{x} \in \mathbb{R}^N$$

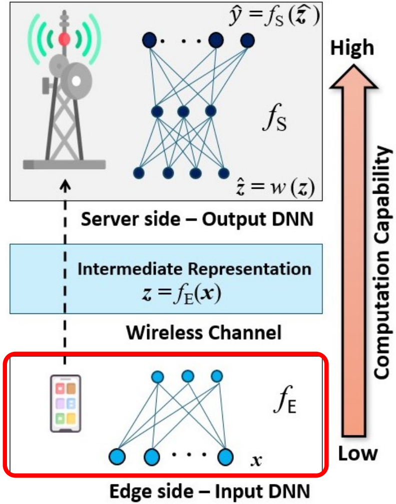


Split Learning - (High) Mathematical Formulation

$$z = f_E(x)$$

$$f_E : \mathbb{R}^N \to \mathbb{R}^n, \quad n < N$$

$$x \in \mathbb{R}^N$$



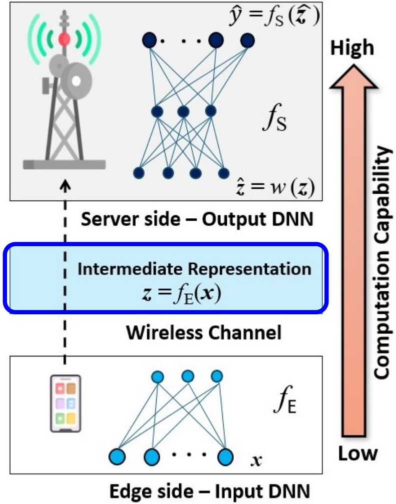
Split Learning - (High) Mathematical Formulation

$$\hat{\boldsymbol{z}} = \mathcal{W}(\boldsymbol{z})$$

$$\boldsymbol{z} = f_{\mathrm{E}}(\boldsymbol{x})$$

$$f_{\mathrm{E}} : \mathbb{R}^N \to \mathbb{R}^n, \quad n < N$$

$$\boldsymbol{x} \in \mathbb{R}^N$$



Split Learning - (High) Mathematical Formulation

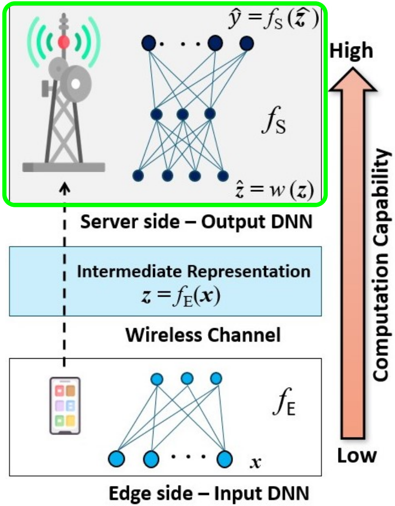
$$f_S : \mathbb{R}^n \to \mathbb{R}^*$$

$$\hat{\boldsymbol{z}} = \mathcal{W}(\boldsymbol{z})$$

$$\boldsymbol{z} = f_E(\boldsymbol{x})$$

$$f_E : \mathbb{R}^N \to \mathbb{R}^n, \quad n < N$$

$$\boldsymbol{x} \in \mathbb{R}^N$$



Split Learning - (High) Mathematical Formulation

$$\hat{y} = f_S(\hat{\boldsymbol{z}})$$

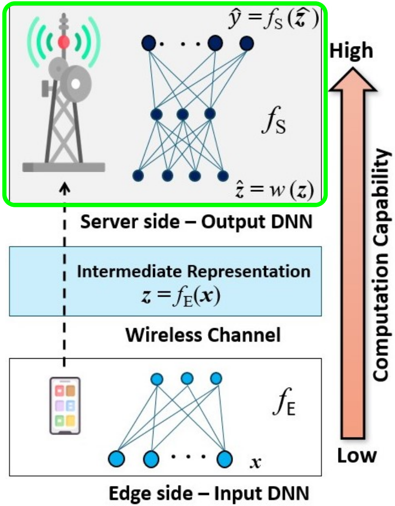
$$f_S : \mathbb{R}^n \to \mathbb{R}^*$$

$$\hat{\boldsymbol{z}} = \mathcal{W}(\boldsymbol{z})$$

$$\boldsymbol{z} = f_E(\boldsymbol{x})$$

$$f_E : \mathbb{R}^N \to \mathbb{R}^n, \quad n < N$$

$$\boldsymbol{x} \in \mathbb{R}^N$$



Split Learning—Based RNNs

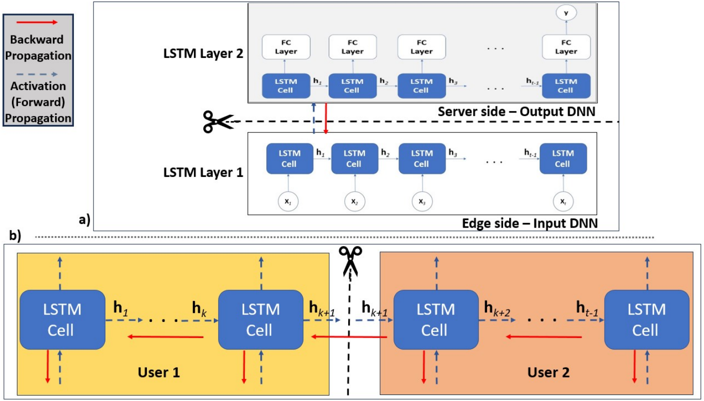


Figure 1: LSTM–based split learning/inference algorithms: a) LSTMSPLIT[2] b) Fedsl [3]

[2]L. Jiang, et al. , "LSTMSPLIT: effective SPLIT learning based LSTM on sequential time-series data," 2022

[3]A. Abedi and S. S. Khan, "Fedsl: Federated split learning on distributed sequential data in recurrent neural networks," *Multimed. Tools. Appl.*, vol. 83, pp. 28891–28911, Sept. 2023.

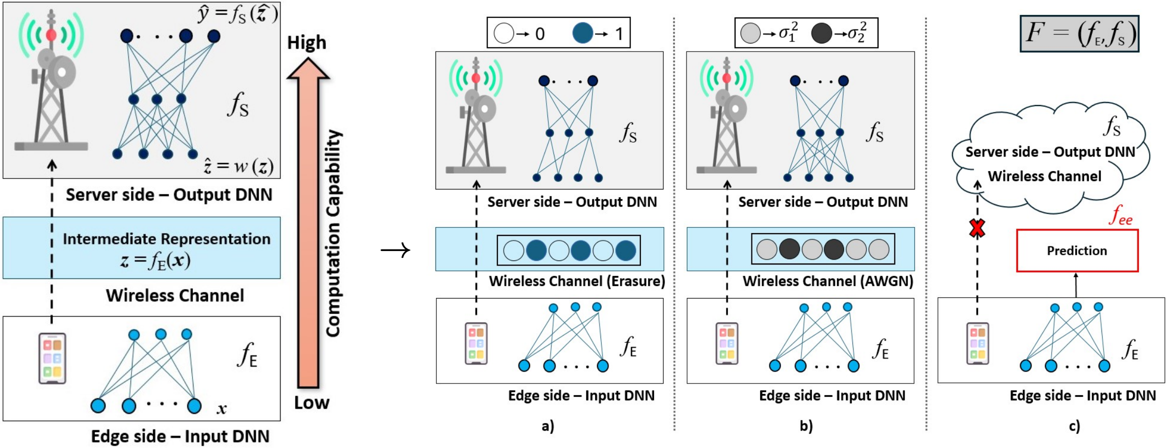
Split Learning with Channel Integration

Motivation

- **Challenge** - Random and time-varying nature of the wireless channel.
- **Steps toward a solution:**
  - Integrate diverse channel conditions during the offline SL phase.
  - Learn optimal intermediate representations for feature transmission.
  - Design a robust server-side subnetwork resilient to channel variations.
  - Mitigate channel perturbations during the online split inference phase.
- **Ultimate goal:** Minimize the mean-squared error (MSE) for a given channel $\mathcal{W}$:

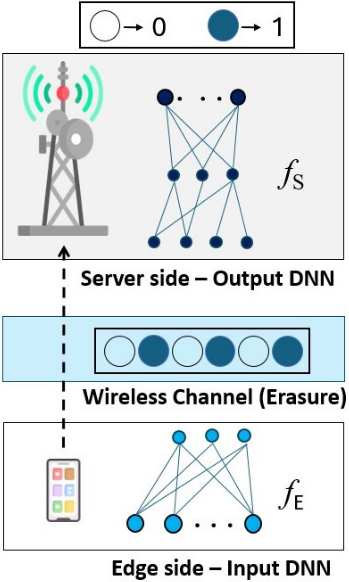
$$\mathcal{L}(y, \hat{y}) = \frac{1}{P} \sum_{\mathcal{D}} (y - \hat{y})^2$$

System Model



Proposed Framework - **Erasure Channel**

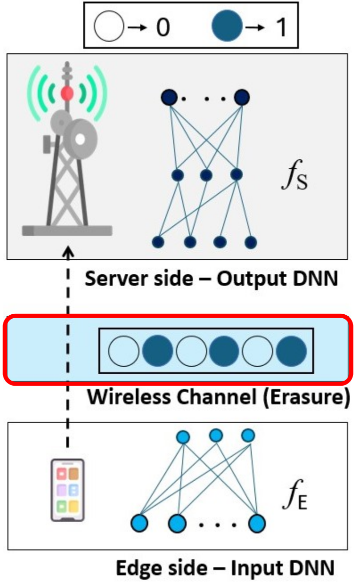
- Erasure channel:
  - n-dimensional binary vector $\boldsymbol{q} \in \{0, 1\}^n$
  - Each symbol of $\boldsymbol{q}$ - Bernoulli distribution
  - Erasure probability p



Proposed Framework - **Erasure Channel**

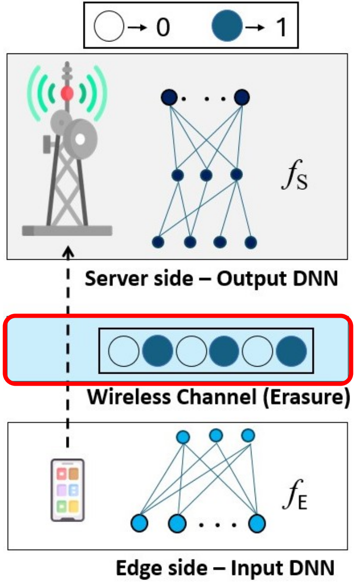
- Erasure channel:
  - n-dimensional binary vector $\boldsymbol{q} \in \{0, 1\}^n$
  - Each symbol of $\boldsymbol{q}$ - Bernoulli distribution
  - Erasure probability p

- $\hat{\boldsymbol{z}} = \mathcal{W}(\boldsymbol{z}) = \boldsymbol{z} \odot \boldsymbol{q}$



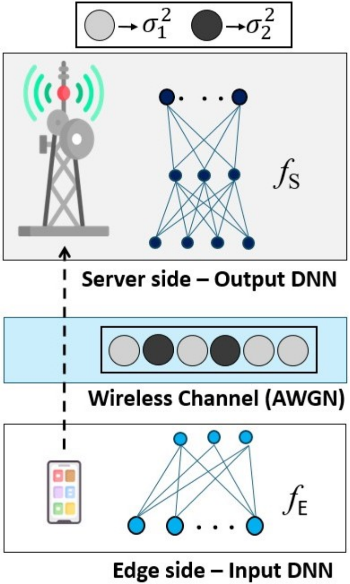
Proposed Framework - **Erasure Channel**

- Erasure channel:
  - n-dimensional binary vector $\boldsymbol{q} \in \{0,1\}^n$
  - Each symbol of $\boldsymbol{q}$ - Bernoulli distribution
  - Erasure probability p
- $\hat{\boldsymbol{z}} = \mathcal{W}(\boldsymbol{z}) = \boldsymbol{z} \odot \boldsymbol{q}$
- Training integration:
  - **Dropout layer** between the edge and server
  - Hidden units are randomly dropped with prob. p



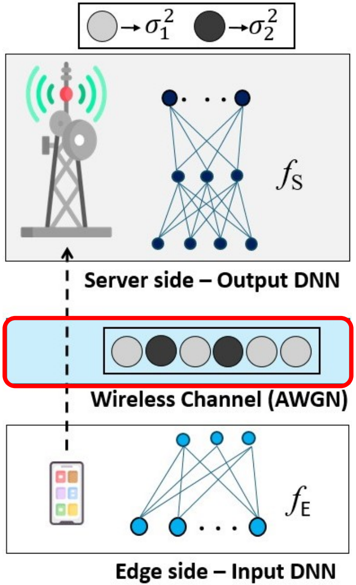
Proposed Framework - **AWGN Channel**

- AWGN channel (and its extension):
  - M–dimensional real vector $\boldsymbol{n} \in \mathbb{R}^n$.
  - Introduction of occasional **deep fades**.
  - UAV–based scenarios



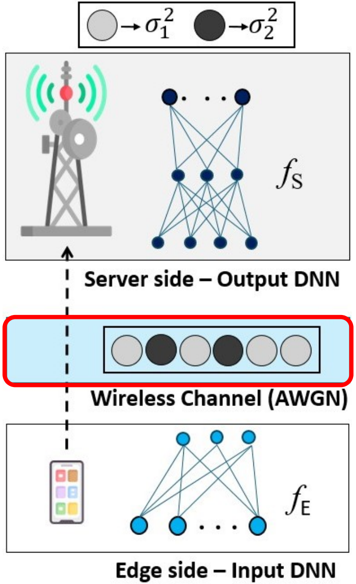
Proposed Framework - **AWGN Channel**

- AWGN channel (and its extension):
  - M–dimensional real vector $\boldsymbol{n} \in \mathbb{R}^n$.
  - Introduction of occasional **deep fades**.
  - UAV–based scenarios
- $\hat{\boldsymbol{z}} = \mathcal{W}(\boldsymbol{z}) = \boldsymbol{z} + \boldsymbol{n}$



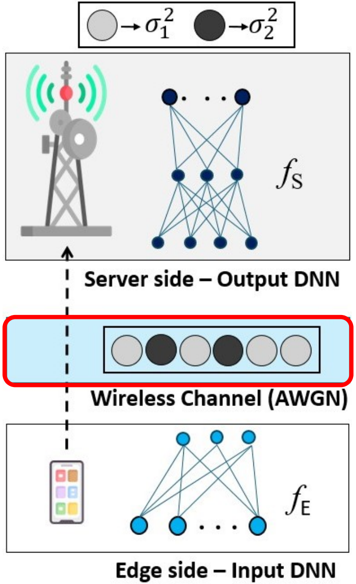
Proposed Framework - **AWGN Channel**

- AWGN channel (and its extension):
  - M–dimensional real vector $\boldsymbol{n} \in \mathbb{R}^n$.
  - Introduction of occasional **deep fades**.
  - UAV–based scenarios
- $\hat{\boldsymbol{z}} = \mathcal{W}(\boldsymbol{z}) = \boldsymbol{z} + \boldsymbol{n}$
- $\boldsymbol{n}$ contains:
  - n_1 i.i.d. samples of a Gaussian variable $\mathcal{N}_1(0, \sigma_1^2)$.
  - n_2 i.i.d. samples of a Gaussian variable $\mathcal{N}_2(0, \sigma_2^2)$.
  - $\sigma_1^2 < \sigma_2^2, n = n_1 + n_2$



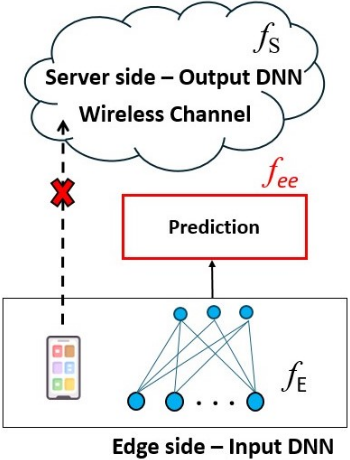
Proposed Framework - **AWGN Channel**

- AWGN channel (and its extension):
  - M–dimensional real vector $\boldsymbol{n} \in \mathbb{R}^n$.
  - Introduction of occasional **deep fades**.
  - UAV–based scenarios
- $\hat{\boldsymbol{z}} = \mathcal{W}(\boldsymbol{z}) = \boldsymbol{z} + \boldsymbol{n}$
- $\boldsymbol{n}$ contains:
  - n_1 i.i.d. samples of a Gaussian variable $\mathcal{N}_1(0, \sigma_1^2)$.
  - n_2 i.i.d. samples of a Gaussian variable $\mathcal{N}_2(0, \sigma_2^2)$.
  - $\sigma_1^2 < \sigma_2^2, n = n_1 + n_2$
- Training integration:
  - **Non-trainable noise layer**
  - Symbols of $\boldsymbol{z}$ corrupted independently.
  - Random nature of the channel - **Regularization**.



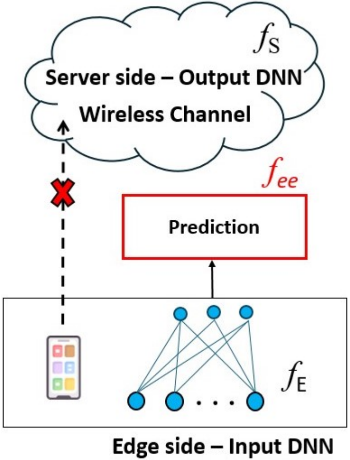
Proposed Framework - **Early–Exit Strategy**

- Motivation:
  - IoT devices operate in challenging environments.



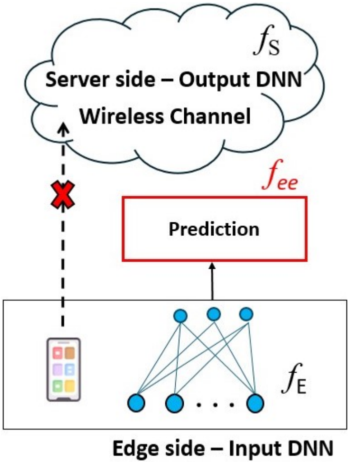
Proposed Framework - **Early–Exit Strategy**

- Motivation:
  - IoT devices operate in challenging environments.
  - Impact of adverse channel conditions on overall performances?



Proposed Framework - **Early–Exit Strategy**

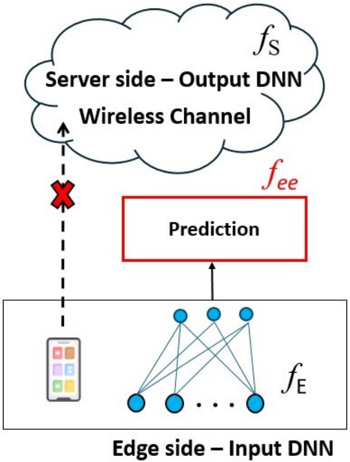
- Motivation:
  - IoT devices operate in challenging environments.
  - Impact of adverse channel conditions on overall performances?
  - Should we send (highly corrupted) intermediate representation?



Proposed Framework - **Early–Exit Strategy**

- Motivation:
  - IoT devices operate in challenging environments.
  - Impact of adverse channel conditions on overall performances?
  - Should we send (highly corrupted) intermediate representation?

- Solution - **Early–Exit Strategy**:
  - Additional NN output at the edge device - f_{ee}.
  - $\hat{y}_{ee} = f_{ee}(\mathbf{z})$
  - CSI knowledge, local decision (prediction MSE).



Proposed Framework - **Early–Exit Strategy**

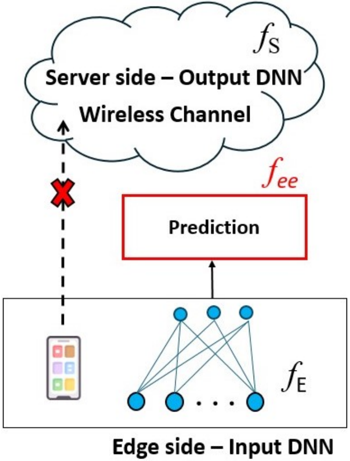
- Training integration - Joint training with overall pipeline:
  - Compound loss function.
  - Without prior channel knowledge:

$$\mathcal{L}_{ee} = \mathcal{L}(\hat{y}, y) + \mathcal{L}(\hat{y}_{ee}, y)$$

  - With prior channel knowledge:

$$\mathcal{L}_{ee} = \lambda\mathcal{L}(\hat{y}, y) + (1-\lambda)\mathcal{L}(\hat{y}_{ee}, y),$$

  - $\lambda \in [0, 1]$ - Weight parameter, shape system behavior.
- Flexible trade-off - EE and full system performance.



Heterogeneous IoT Devices

- IoT network consists of C edge devices:
  - Independent local datasets.
  - $\boldsymbol{z}_i = f_{\mathrm{E}_i}(\boldsymbol{x}_i)$
- Server side - Concatenation:

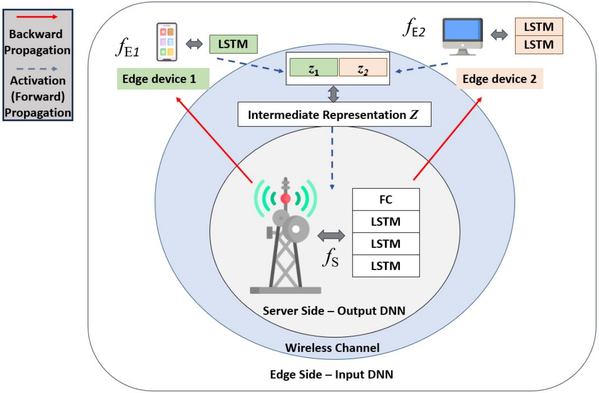
$$\hat{\boldsymbol{Z}} = \mathcal{W}(\boldsymbol{Z}) = (\hat{\boldsymbol{z}}_1, \hat{\boldsymbol{z}}_2, \ldots, \hat{\boldsymbol{z}}_C)$$

- Separate prediction for each of C devices:

$$(\hat{y}_1, \hat{y}_2, \ldots, \hat{y}_C) = f_{\mathrm{S}}(\hat{\boldsymbol{z}}_1, \hat{\boldsymbol{z}}_2, \ldots, \hat{\boldsymbol{z}}_C)$$

- Joint optimization of $(\{f_{\mathrm{E}_i}\}_{i=1,\ldots,C}, f_{\mathrm{S}})$.
- Training integration - Loss function:

$$\mathcal{L}_{het} = \sum_{i=1}^{i=C} \mathcal{L}_i(y_i, \hat{y}_i)$$



Performance Evaluation

Experimental Setup - Use Case 1

- **Generic Time–Series IoT System:**
  - Is this approach effective?
  - Amazon Stock Data dataset (6516 instances).
  - Predict *Open* using previous 30 days of *Close*.
  - LSTM–based neural network[4].
- Training/Validation/Testing - 60/10/30%.

[4] More details about the data preprocessing and training procedure: V. Ninkovic, D. Vukobratovic, D. Miskovic and M. Zennaro, "COMSPLIT: A Communication-Aware Split Learning Design for Heterogeneous IoT Platforms," *IEEE Internet of Things Journal,* vol. 12, no. 5, pp. 5305-5319, 2025.

Use Case 1 - **Erasure Channel**

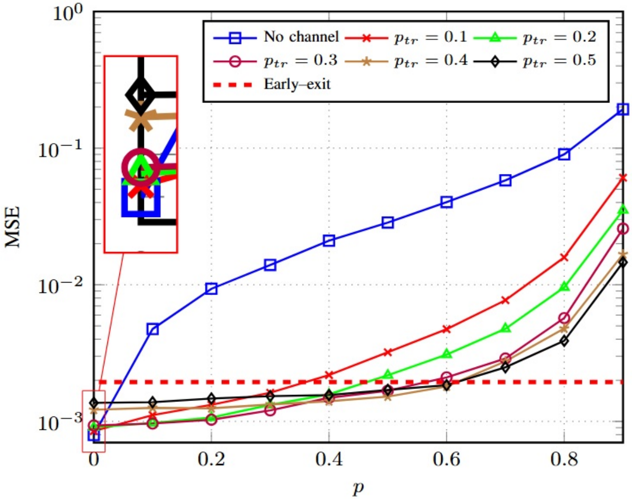


Figure 2: Erasure channel: *MSE* versus symbol erasure probability (p) performances for various training symbol erasure probability p_{tr} values introduced during training and early–exit strategy

Use Case 1 - **AWGN Channel**

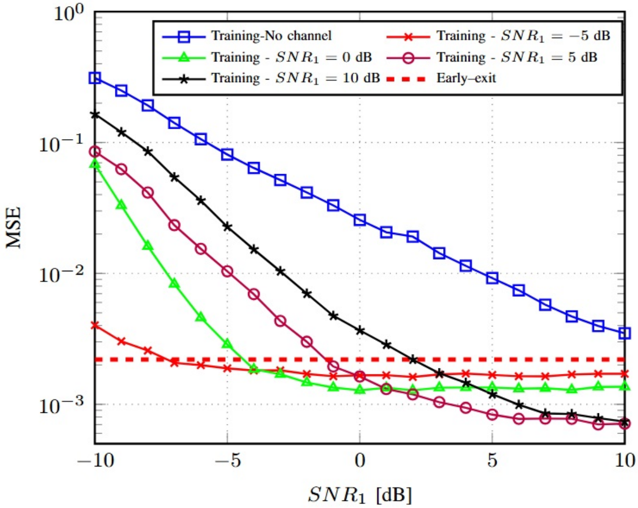


Figure 3: AWGN channel: *MSE* versus *SNR* performances for various SNR_1 and $SNR_2 = SNR_1 - 5$ dB values introduced during training and early–exit strategy ($n_1 = n_2 = 5$).

Use Case 1 - **Heterogeneous Setup**

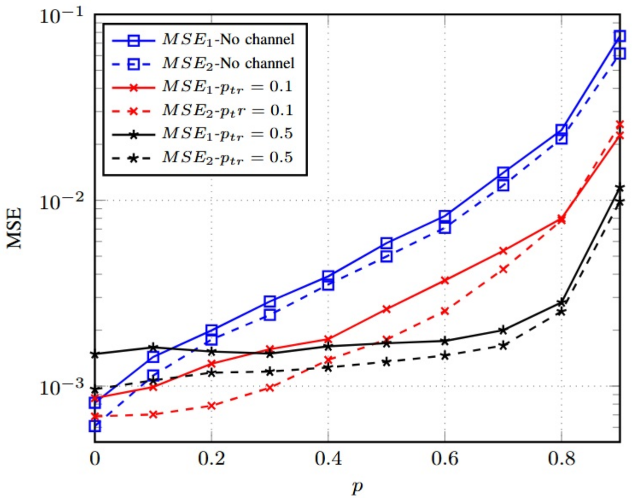


Figure 4: Heterogeneous IoT Devices: *MSE* versus symbol erasure probability p performances for 2 devices ($\mathcal{C}(f_{E_1}) < \mathcal{C}(f_{E_2})$) under different p_{tr}

Experimental Setup - Use Case 2

- Is this approach effective on **real–world** data?

Experimental Setup - Use Case 2

- Is this approach effective on **real–world** data?
- Towards real-world deployment: **Water Quality Monitoring IoT System.**

Experimental Setup - Use Case 2

- Is this approach effective on **real–world** data?
- Towards real-world deployment: **Water Quality Monitoring IoT System.**
- Dataset perfectly suited to the environmental problem:
  - Pollution of the Danube river near Novi Sad
  - 3,264 instances - Each instance represents a daily measurement from 2013 to 2022
  - Eight different water quality parameters - Temperature, pH value, electrical conductivity, dissolved oxygen, oxygen saturation, ammonium, and nitrite

Experimental Setup - Use Case 2

- Is this approach effective on **real–world** data?
- Towards real-world deployment: **Water Quality Monitoring IoT System.**
- Dataset perfectly suited to the environmental problem:
  - Pollution of the Danube river near Novi Sad
  - 3,264 instances - Each instance represents a daily measurement from 2013 to 2022
  - Eight different water quality parameters - Temperature, pH value, electrical conductivity, dissolved oxygen, oxygen saturation, ammonium, and nitrite
- Estimation of dissolved oxygen based on previous 30 days:
  - Data storage limitations.

Experimental Setup - Use Case 2

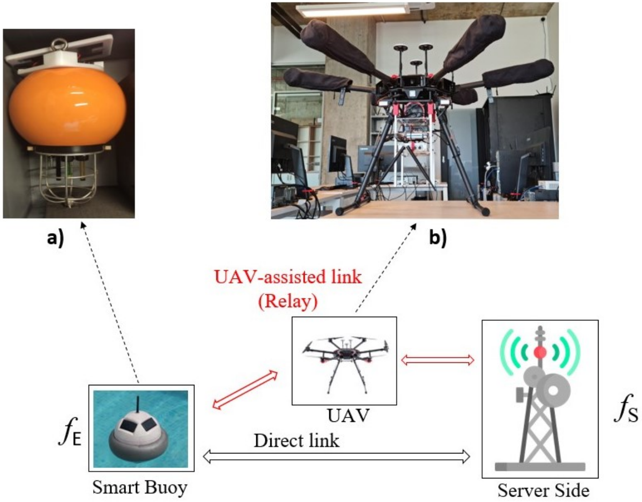


Figure 5: Water quality monitoring IoT system - Initial setup with a) smart buoy and b) UAV equipped with communication equipment

Use Case 2 - Danube River

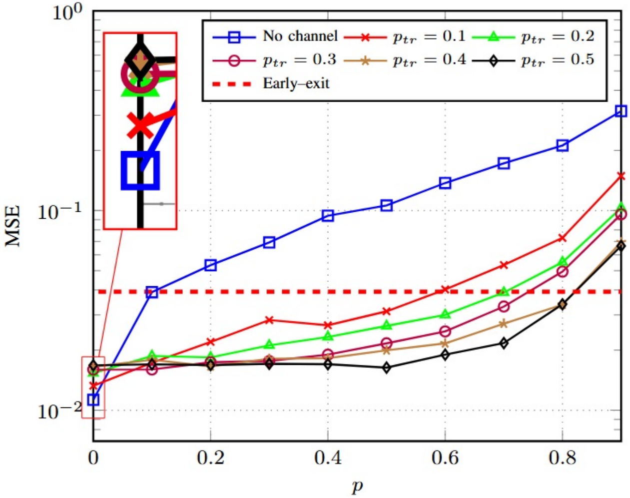


Figure 6: Erasure channel: *MSE* versus symbol erasure probability (p) performances for various p_{tr} values introduced during training and early–exit strategy

Use Case 2 - Danube River

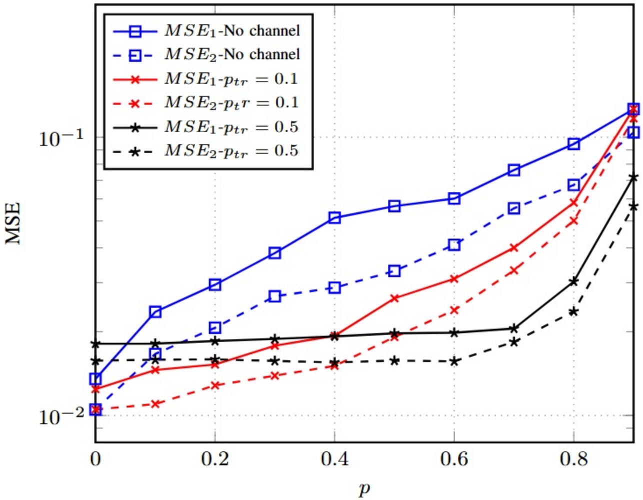


Figure 7: Heterogeneous IoT Devices: *MSE* versus symbol erasure probability p performances for 2 devices ($\mathcal{C}(f_{E_1}) < \mathcal{C}(f_{E_2})$) under different p_{tr}

(Instead of) Conclusion

Discussion & Ongoing Work

- Integration of SL with UAV relaying in IoT networks?[5]

[5] V. Ninkovic, D. Vukobratovic and D. Miskovic, "UAV-assisted Distributed Learning for Environmental Monitoring in Rural Environments," in *Proc. 2024 7th BalkanCom*, Ljubljana, Slovenia, 2024, pp. 296-300

[6] R. S. Molina, V. Ninkovic, D. Vukobratovic, M. L. Crespo and M. Zennaro, "Efficient Split Learning LSTM Models for FPGA-based Edge IoT Devices," in *Proc. 2025 IEEE ICMLCN,* Barcelona, Spain, 2025, pp. 1-6

Discussion & Ongoing Work

- Integration of SL with UAV relaying in IoT networks?[5]
- FPGA deployment of $(f_{\mathrm{E}}, f_{\mathrm{S}})$?[6]

[5] V. Ninkovic, D. Vukobratovic and D. Miskovic, "UAV-assisted Distributed Learning for Environmental Monitoring in Rural Environments," in *Proc. 2024 7th BalkanCom*, Ljubljana, Slovenia, 2024, pp. 296-300

[6] R. S. Molina, V. Ninkovic, D. Vukobratovic, M. L. Crespo and M. Zennaro, "Efficient Split Learning LSTM Models for FPGA-based Edge IoT Devices," in *Proc. 2025 IEEE ICMLCN,* Barcelona, Spain, 2025, pp. 1-6

Discussion & Ongoing Work

- Integration of SL with UAV relaying in IoT networks?[5]
- FPGA deployment of $(f_\mathrm{E}, f_\mathrm{S})$?[6]
- Real-world design and deployment of end-to-end system.

[5] V. Ninkovic, D. Vukobratovic and D. Miskovic, "UAV-assisted Distributed Learning for Environmental Monitoring in Rural Environments," in *Proc. 2024 7th BalkanCom*, Ljubljana, Slovenia, 2024, pp. 296-300

[6] R. S. Molina, V. Ninkovic, D. Vukobratovic, M. L. Crespo and M. Zennaro, "Efficient Split Learning LSTM Models for FPGA-based Edge IoT Devices," in *Proc. 2025 IEEE ICMLCN,* Barcelona, Spain, 2025, pp. 1-6

Discussion & Ongoing Work

- Integration of SL with UAV relaying in IoT networks?[5]
- FPGA deployment of $(f_{\mathrm{E}}, f_{\mathrm{S}})$?[6]
- Real-world design and deployment of end-to-end system.
- **Challenge(s)**:
  - How to efficiently protect the symbols of $\boldsymbol{z}$?
  - How to maximize the informativeness of $\boldsymbol{z}$ for efficient transmission?

[5] V. Ninkovic, D. Vukobratovic and D. Miskovic, "UAV-assisted Distributed Learning for Environmental Monitoring in Rural Environments," in *Proc. 2024 7th BalkanCom*, Ljubljana, Slovenia, 2024, pp. 296-300

[6] R. S. Molina, V. Ninkovic, D. Vukobratovic, M. L. Crespo and M. Zennaro, "Efficient Split Learning LSTM Models for FPGA-based Edge IoT Devices," in *Proc. 2025 IEEE ICMLCN,* Barcelona, Spain, 2025, pp. 1-6

Discussion & Ongoing Work

- Integration of SL with UAV relaying in IoT networks?[5]
- FPGA deployment of $(f_\mathrm{E}, f_\mathrm{S})$?[6]
- Real-world design and deployment of end-to-end system.
- **Challenge(s)**:
  - How to efficiently protect the symbols of $\boldsymbol{z}$?
  - How to maximize the informativeness of $\boldsymbol{z}$ for efficient transmission?
- **Solution(s)** - When Learning Meets the Channel: Edge AI through Split and Semantic Design **(Part II)**

---

[5] V. Ninkovic, D. Vukobratovic and D. Miskovic, "UAV-assisted Distributed Learning for Environmental Monitoring in Rural Environments," in *Proc. 2024 7th BalkanCom*, Ljubljana, Slovenia, 2024, pp. 296-300

[6] R. S. Molina, V. Ninkovic, D. Vukobratovic, M. L. Crespo and M. Zennaro, "Efficient Split Learning LSTM Models for FPGA-based Edge IoT Devices," in *Proc. 2025 IEEE ICMLCN,* Barcelona, Spain, 2025, pp. 1-6

Acknowledgment[7]

Dejan Vukobratovic, PhD

Dragisa Miskovic, PhD

Romina Soledad Molina, PhD

Maria Liz Crespo, PhD

Marco Zennaro, PhD

[7] This work has received funding from the Horizon 2020 grant agreement No 101086387 - REMARKABLE.

References I

[1] O. Gupta and R. Raskar, "Distributed learning of deep neural network over multiple agents," *J. Netw. Comput. Appl.,* vol. 116, pp. 1–8, Aug. 2018.

[2] P. Vepakomma, O. Gupta, T. Swedish, and R. Raskar, "Split learning for health: Distributed deep learning without sharing raw patient data," 2018, arXiv:1812.00564. [Online]. Available: https://doi.org/10.48550/arXiv.1812.00564

[3] M. G. Poirot, P. Vepakomma, K. Chang, J. Kalpathy-Cramer, R. Gupta, and R. Raskar, "Split learning for collaborative deep learning in healthcare," 2019, arXiv:1912.12115. [Online]. Available: https://doi.org/10.48550/arXiv.1912.12115

[4] S. Itahara, T. Nishio, Y. Koda, and K. Yamamoto, "Communication-Oriented Model Fine-Tuning for Packet-Loss Resilient Distributed Inference Under Highly Lossy IoT Networks," *IEEE Access*, vol.10, pp. 14969–14979, 2022.

[5] J. Shao and J. Zhang, "Communication-Computation Trade-off in Resource-Constrained Edge Inference," *IEEE Commun. Mag.,* vol. 58, no. 12, pp. 20-26, Dec. 2020.

References II

[6] M. Krouka, A. Elgabli, C. b. Issaid, and M. Bennis, “Communication efficient split learning based on analog communication and over the air aggregation,” in *Proc. 2021 IEEE Global Commun. Conf. (GLOBECOM),* Madrid, Spain, Dec. 7–11, pp. 1–6, 2021.

[7] J. Shao and J. Zhang, “Bottlenet++: An end-to-end approach for feature compression in device-edge co-inference systems,” in *Proc. 2020 IEEE Int. Conf. on Commun. Workshops (ICC Workshops),* Virtual Event, June 7–11, pp. 1–6, 2020.

[8] M. Jankowski, D. Gunduz, and K. Mikolajczyk, "Adaptive Early Exiting for Collaborative Inference over Noisy Wireless Channels," 2023, arXiv: 2311.18098. [Online]. Available: https://doi.org/10.48550/arXiv.2311.18098

[9] E. Samikwa, A. Di Maio, and T. Braun, "Adaptive early exit of computation for energy-efficient and low-latency machine learning over IoT networks,” in *Proc. IEEE 19th Annu. Consum. Commun. Netw. Conf. (CCNC),* Virtual Event, Jan. 8–11, pp. 200–206, 2022.

References III

[10] E. Baccarelli, M. Scarpiniti, A. Momenzadeh, and S. S. Ahrabi, "Learning in-the-fog (LiFo): Deep learning meets fog computing for the minimum energy distributed early-exit of inference in delay-critical IoT realms," *IEEE Access*, vol. 9, pp. 25716–25757, 2021.

[11] Y. Yang, Z. Zhang, Y. Tian, Z. Yang, C. Huang, C. Zhong, and K.-K. Wong, "Over-the-air split machine learning in wireless MIMO networks," *IEEE J. Sel. Areas Commun.*, vol. 41, no. 4, pp. 1007–1022, Feb. 2023.

[12] E. Samikwa, A. D. Maio, and T. Braun, "DISNET: Distributed Micro-Split Deep Learning in Heterogeneous Dynamic IoT," *IEEE Internet Things J.,* vol. 11, no. 4, pp. 6199-6216, Feb., 2024.

[13] S. Tuli, G. Casale, and N. R. Jennings, "SplitPlace: AI augmented splitting and placement of large-scale neural networks in mobile edge environments," *IEEE Trans. Mobile. Comput.*, vol. 22, no. 9, pp. 5539-5554, Sept. 2023.

[14] E. Samikwa, A. Di Maio, and T. Braun, "ARES: Adaptive resource aware split learning for Internet of Things," *Comput. Netw.,* vol. 218, no. 109380., Dec. 2022.

References IV

[15] W. Wu, M. Li, K. Qu, C. Zhou, X. Shen, W. Zhuang, X. Li, and W. Shi, "Split Learning Over Wireless Networks: Parallel Design and Resource Management," *IEEE J. Sel. Areas Commun.*, vol. 41, no. 4, pp. 1051-1066, April 2023.

[16] Y. Koda, J. Park, M. Bennis, K. Yamamoto, T. Nishio, M. Morikura, and K. Nakashima, "Communication-efficient multimodal split learning for mmWave received power prediction," *IEEE Commun. Lett.*, vol. 24, no. 6, pp. 1284–1288, June 2020.

[17] L. Jiang, Y. Wang, W. Zheng, C. Jin, Z. Li, and G. S. Teo, "LSTMSPLIT: effective SPLIT learning based LSTM on sequential time-series data," 2022, arXiv: cs.LG/2203.04305. [Online]. Available: https://doi.org/10.48550/arXiv.2203.04305

Thank you for your attention!

vukan.ninkovic@ivi.ac.rs

ninkovic@uns.ac.rs

Appendix: More Explanations and Interesting Results...

Use Case 1 - **Erasure Channel**

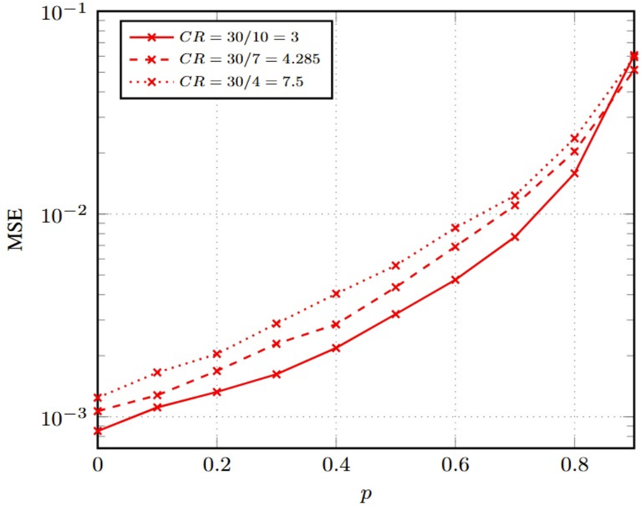


Figure 8: MSE versus erasure probability p for different compression rates obtained at $p_{tr} = 0.1$.

Use Case 1 - **AWGN Channel**

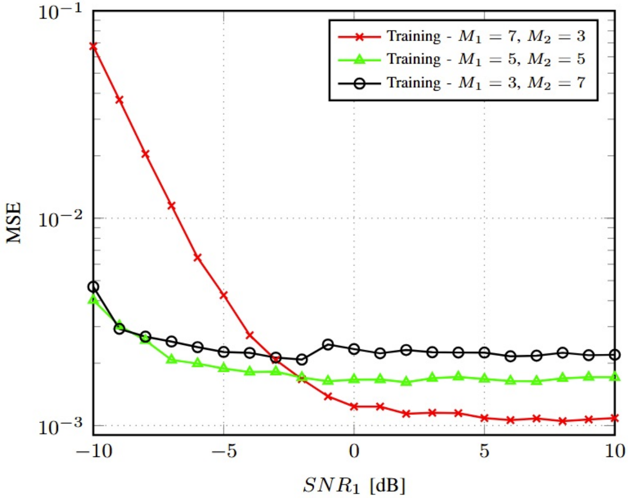


Figure 9: MSE versus *SNR* performances for various n_1 and n_2 values trained at $SNR_1 = -5$ dB.

Early–Exit - λ Influence

Table 1: Earl–exit versus full performances regarding loss function parameter λ.

λ	Early–exit MSE	Full system MSE ($p_{tr} = 0.1$)
0.1	0.00102	0.00093
0.5	0.00194	0.00085
0.9	0.004	0.00052