OTFS Modulations

Fadi Hantouli, Milestone2

Abstract

This project will explore the concept of Orthogonal Time Frequency Space (OTFS) for use in establishing a transmitter and receiver. OTFS is an advanced form of Orthogonal Frequency Division Multiplexing (OFDM) which has recently been used in several wireless communication systems. The goals and objectives of this project is to study the Delay and Doppler effects on an OTFS system and explore the use of RTL-SDR software-defined radio technology for data analysis and reception. Through the design, modeling, and implementation of an OTFS system, experiments will be conducted to analyze its performance in terms of data rate, packet error rate, and spectral efficiency. The findings of this project will be invaluable for future research in the field of wireless communication and the use of RTL-SDR technology and will provide insight into the potential of OTFS systems in the development of communication systems. See figure 1 for more details.

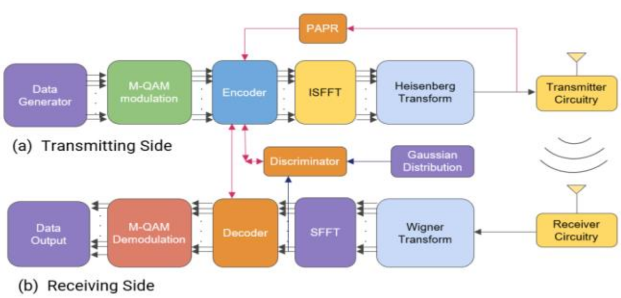

Figure 1- OTFS with NN

1. DEFINITIONS

A. OTFS (Orthogonal Time Frequency Space) modulation is a novel modulation technique that allows for transmitting large amounts of data over a traditional wireless channel. It combines the benefits of several existing modulation techniques, including OFDM, CDMA, and spread spectrum, to provide a robust and efficient modulation scheme. The main benefits of OTFS modulation are that it can provide higher data rates, better immunity to interference, and improved spectral efficiency compared to other modulation techniques. Additionally, OTFS modulation can be used to provide a better link budget and higher spectral efficiency than traditional modulation techniques [1][2][3].

B. Adding a neural network to improve the PAPR (Peak to Average Power Ratio) and BET (Bit Error Rate) of OTFS modulation can be done by using a neural network-based precoder and decoder. The neural network-based precoder and decoder can be used to optimize the OTFS modulation parameters and adapt them to the wireless channel conditions. This can help to reduce the PAPR and BET of the OTFS modulation, resulting in better transmission performance. Additionally, the neural network can be used to identify and mitigate interference in a wireless channel, resulting in improved reception performance. In summary, using a neural network to improve the PAPR and BET of OTFS modulation can provide an efficient and robust modulation scheme for transmitting large amounts of data over a wireless channel [4].

C. Neural Network model is the Adversarial Autoencoder. Adversarial Autoencoders (AAEs) are a type of generative model that combines the ideas of generative adversarial networks (GANs) and autoencoders [5]. They are used to learn latent representations of data, with the goal of making the generated data indistinguishable from real data. The model consists of two parts: an encoder network, which maps the input data to a latent representation, and a decoder network, which maps the latent representation back to the original data. The latent representation is then used to generate new data. In addition to the autoencoder networks, AAEs also have a discriminator network, which is trained to distinguish between the generated data and real data. The discriminator is used to provide feedback to the encoder and decoder networks during training [6]. The encoder and decoder networks are optimized to generate data that is realistic and indistinguishable from real data, while the discriminator is optimized to classify the data correctly [7]. AAEs are used for a variety of tasks, including image generation, image super-resolution, text generation, and anomaly detection. They can also be used to learn latent representations of data which can then be used for downstream tasks such as classification and clustering. AAEs can also be used to generate synthetic data for training and testing machine learning models [8].

D. MATLAB live: I plan on using a MATLAB live script that is well-documented and well-formatted. Additionally, I plan on providing users with small modules to help them get familiar with OTFS, as well as a tutorial-like approach to teaching them the basics of pre-coding OFDM. This will give users the opportunity to learn the necessary steps, while also providing them with a visual representation of the process [9][10].

E. RTL-SDR stands for “Real-Time Long Software Defined Radio”. It is a technology that enables software to be used to decode radio frequencies, such as FM, AM, and digital signals. The advantage of RTL-SDR is that it is low cost, wide frequency range, and high-resolution. It can be used to receive and decode signals from a variety of sources, including shortwave, ham radio, and even digital television. It can also be used to create custom applications such as spectrum analyzers, weather receivers, and more[11][12].

2. WORK ORGANIZATION

Open-source software has been an integral part of technological development for decades, and it is more important now than ever before. With the help of open-source projects, developers and users alike can benefit from the collective knowledge and effort of the entire community. As a result, I am excited to contribute to the open-source community with the OTFS Performance using SDR project [13][14].

The OTFS Performance using SDR project is an exciting opportunity for me to contribute to the open-source community. By participating in this project, I will be helping to build a better, more efficient communication infrastructure for the future. My primary contribution to this project will be in the form of research and development. I will be researching the various aspects of the project and working to develop new ways to improve the performance of the OTFS system. This will involve researching the latest technologies, such as SDR, and how they can be used to improve the system. I will also be exploring ways to reduce the complexity of the system so that it can be deployed quickly and efficiently. In addition to research and development, I will also be contributing to the project by:

1. Create a README file: The README file is a text file that contains information about a repository. This can include information about the OTFS project, its purpose, how to install and use it, how to contribute, license information, and more. It is often the first file visitors to a repository will look at, so it should provide a good overview of the project.

2. Create a documentation folder: This folder should contain all the necessary documentation a user needs to understand and use the OTFS code. This could include installation instructions, user guides, tutorials, API references, and other relevant information.

3. Create an issue tracker: An issue tracker will help users report issues they encounter while using the OTFS code and allow other users to help troubleshoot and provide solutions.

4. Create a license file: The license file should explain how the user can use the code and what restrictions apply.

5. Create a contributor’s guide: The contributor's guide should explain how to contribute to the repository, what rules and standards need to be followed, and how to submit changes to the code.

6. Create a code folder: This folder should contain the actual code (MATLAB live script) the repository comprises. This should include any libraries, modules:

   I. Data Generation: This module includes the MATLAB code necessary to generate random data and create training and testing sets.
   II. Pre-Coding: This module contains the MATLAB code to pre-code the OTFS data before sending it over the channel.
   III. Channel Model: This module contains MATLAB code to simulate a channel model for the OTFS system.
   IV. Reception and Decoding: This module contains the MATLAB code to receive and decode the OTFS data.
   V. Performance Evaluation: This module contains the MATLAB code to evaluate the performance of the OTFS system.

7. Create a testing folder: The testing folder should contain any tests written for the code. This will help ensure the code works as expected and the features are implemented correctly.

8. Create a tutorials folder: The following tutorials are included in this folder:
   I. Introduction to OTFS: This tutorial provides an introduction to the OTFS system, its applications, and its implementation.

II. Pre-Coding: This tutorial provides an introduction to pre-coding and explains how to use the pre-coding module.
III. Channel Model: This tutorial provides an introduction to the channel model and explains how to use the channel model module.
IV. Reception and Decoding: This tutorial provides an introduction to reception and decoding and explains how to use the reception and decoding module.
V. Performance Evaluation: This tutorial provides an introduction to performance evaluation and explains how to use the performance evaluation module.

9. Results Foder: The Results Folder in GitHub is a place to store the output from the various scripts and programs associated with a repository. It is a folder that is used to store the results of the various processes that are run on the code in the repository. This results folder contains the output from tests code. It is also used to store the output from the build and deploy processes. This results folder is a place to store the results of all the processes that have been run on the code so that they can be easily accessed and analyzed.

10. Create a release folder: The release folder should contain any releases of the code that have been made available to the public. This will allow users to access the latest version of the code.

3. MODELING OF THE OTFS

4. WORK DISTRIBUTION

I will be working alone. Please see Appendix A.

5. ASSESSMENT AND CONTINGENCY PLANS

A. conditions. Finally, advanced equalization techniques such as joint iterative equalization and decoding can be used to improve the system's performance further.

In summary, the main expected issue in developing OFDM to OTFS is the Doppler spread caused by the time variation of the channel. To address this issue, a decision-directed approach, robust filter bank, adaptive modulation and coding, and advanced equalization techniques such as joint iterative equalization and decoding can be used to reduce the impact of Doppler spread and improve system performance. The assessment of this project will involve a thorough evaluation of the performance of the OTFS system. This evaluation will include creating a system simulation to compare the system's performance to existing OFDM systems. The simulation will be run with different parameters such as data rate, packet error rate, and spectral efficiency. The data generated from these simulations will be analyzed to determine the optimal parameters for the system. Additionally, the system's performance will be compared to existing OFDM systems to identify any strengths or weaknesses of the OTFS system. Finally, the assessment results will be used to identify areas of improvement and strategies to optimize the system's performance. This could include techniques such as increasing data rates, improving error rates, or increasing spectral efficiency. Ultimately, the assessment will provide insight into the system's performance, which can be used to make informed decisions concerning the optimization of the system.

The contingency plan for dealing with any issues during project development involves four steps:

A. The first step is to begin troubleshooting, it is important to identify the issue that has occurred. This can be done by examining the system's performance and analyzing the signal-to-noise ratio, bit error rate, and other related metrics. Additionally, the channel's frequency response can be used to detect any distortions caused by the Doppler spread. Once the issue has been identified, further analysis can be conducted to determine the root cause and formulate an appropriate solution.

B. The second step is to explore potential remedies. Researching existing solutions and coming up with innovative solutions will be necessary. A decision-directed approach, which tracks the channel dynamics and compensates for the Doppler spread, can be used to tackle this challenge. A robust filter bank at the receiver would also be beneficial in counteracting the impact of Doppler spread. Adaptive modulation and coding is another solution that allows for appropriate adjustment of the modulation and coding scheme in response to the channel conditions. Moreover, advanced equalization techniques such as joint iterative equalization and decoding can be employed to further enhance the system's performance.

C. To evaluate the different solutions to the main expected issue and decide on the best option, it is necessary to conduct simulations or experiments. In these simulations, the OFDM and OTFS systems can be compared regarding their performance under different channel conditions. The results of the simulations can be used to assess the pros and cons of each solution and select the one that is the most effective and efficient. Additionally, the complexity of each solution and the number of resources required for its implementation should also be taken into account in order to determine which solution is the most cost-effective.

The fourth step is to implement the chosen solution. This will involve updating the code, making necessary changes, and testing the solution to ensure it works as expected. It is essential to consider the complexity of the solution and the number of resources needed to implement it. To do this, a decision-directed approach can be used, which involves tracking the channel dynamics and compensating for the Doppler spread. Additionally, a robust filter bank can be used at the receiver to reduce the effect of Doppler spread. Adaptive modulation and coding can also be used to adjust the modulation and coding scheme based on the channel

7. FINAL REPORT

The final report will define and provide support for the OTFS system by utilizing Delay Doppler and RTL-SDR concepts to test the system's performance. This report will also evaluate the design of the OTFS system, seeking out any weaknesses and identifying any strengths when compared to existing OFDM systems. The analysis provided in the report will allow for the identification of areas of improvement, as well as the development of strategies to optimize the performance of the system. Additionally, the report will provide insight into the potential of the OTFS system as a viable communication system for real-world applications.

8. SUMMARY

This report outlines the objectives, research, and implementation of a project to explore the concept of Orthogonal Time Frequency Space (OTFS) for use in establishing a transmitter and receiver. The goals are to study the Delay and Doppler effects on an OTFS system and explore the use of RTL-SDR software-defined radio technology for data analysis and reception at the receiver. Through the design, modeling, and implementation of an OTFS system, experiments will be conducted to analyze its performance in data rate, packet error rate, and spectral efficiency. The findings of this project will be invaluable for future research in wireless communication and the use of RTL-SDR technology. They will provide insight into the potential of OTFS systems in the development of communication systems. [15].

REFERENCES

[1] A. S. Bondre and C. D. Richmond, "On the Zak Transform-based Interpretation of OTFS Modulation," 2022 56th Asilomar Conference on Signals, Systems, and Computers, Pacific Grove, CA, USA, 2022, pp. 715-721, doi: 10.1109/IEEECONF56349.2022.10051983.

[2] W. Liu, L. Zou, B. Bai and T. Sun, "Low PAPR channel estimation for OTFS with scattered superimposed pilots," in China Communications, vol. 20, no. 1, pp. 79-87, Jan. 2023, doi: 10.23919/JCC.2023.01.007.

[3] B. Wang, N. Li, Z. Jiang, J. Zhu, X. She and P. Chen, "On Performance Evaluation of OTFS and OFDM Modulations for Sensing," 2022 14th International Conference on Wireless Communications and Signal Processing (WCSP), Nanjing, China, 2022, pp. 427-431, doi: 10.1109/WCSP55476.2022.10039302.

[4] Y. Gong, Q. Li, F. Meng, X. Li and Z. Xu, "Data-driven deep learning for OTFS detection," in China Communications, vol. 20, no. 1, pp. 88-101, Jan. 2023, doi: 10.23919/JCC.2023.01.008.

[5] H. H. Thai, N. D. Hieu, N. Van Tho, H. D. Hoang, P. T. Duy and V. -H. Pham, "Adversarial AutoEncoder and Generative Adversarial Networks for Semi-Supervised Learning Intrusion Detection System," 2022 RIVF International Conference on Computing and Communication Technologies (RIVF), Ho Chi Minh City, Vietnam, 2022, pp. 584-589, doi: 10.1109/RIVF55975.2022.10013926.

[6] K. Jang, S. Hong, M. Kim, J. Na and I. Moon, "Adversarial Autoencoder Based Feature Learning for Fault Detection in Industrial Processes," in IEEE Transactions on Industrial Informatics, vol. 18, no. 2, pp. 827-834, Feb. 2022, doi: 10.1109/TII.2021.3078414.

[7] N. Tawara, T. Kobayashi, M. Fujieda, K. Katagiri, T. Yazu and T. Ogawa, "Adversarial autoencoder for reducing nonlinear distortion," 2018 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC), Honolulu, HI, USA, 2018, pp. 1669-1673, doi: 10.23919/APSIPA.2018.8659540.

[8] T. A. Eriksson, H. Bülow, and A. Leven, “Applying neural networks in optical communication systems: possible pitfalls,” IEEE Photonics Technology Letters, vol. 29, no. 23, pp. 2091-2094, Sept. 2017.

[9] https://www.mathworks.com/products/matlab/live-editor.html

[10] https://www.mathworks.com/help/matlab/matlab_prog/live-editor-for-documenting-functions.html

[11] https://www.rtl-sdr.com/

[12] https://osmocom.org/projects/rtl-sdr/wiki/Rtl-sdr

[13] https://en.wikipedia.org/wiki/Open-source_software

[14] https://github.com/topics/open-source

[15] G. D. Surabhi, R. M. Augustine and A. Chockalingam, "Peak-to-Average Power Ratio of OTFS Modulation," in IEEE Communications Letters, vol. 23, no. 6, pp. 999-1002, June 2019, doi: 10.1109/LCOMM.2019.2914042.

Appendix A: Schedule of Labor

		March						April								May	
WSB#	Task Name	13	15	20	22	27	29	3	5	10	12	17	19	24	26	1	3
1	OFDM							Milestone 3>				Milestone 4 >>				Milestone 5 Final Report >>>>>	
1.1	Learn OFDM																
1.2	OFDM Code	Milestone 2 >>															
1.3	OFDM TEST																
2	Develop OTFS																
2.1	Learn OTFS																
2.2	OTFS Code																
2.3	ORFS Test																
3	Simulations																
3.1	OFDM Simulation																
3.2	OTFS Simulation																
3.3	Simulations Compare																
4	Deliverables																