

Wavelet-Enhanced Graph Neural Networks: Towards Non-Parametric Network Traffic Modeling

Carlos Güemes-Palau
carlos.guemes@upc.edu

Barcelona Neural Networking Center,
Universitat Politècnica de Catalunya
Barcelona, Catalonia, Spain

Miquel Ferriol-Galmés
miquel.ferriol@upc.edu

Barcelona Neural Networking Center,
Universitat Politècnica de Catalunya
Barcelona, Catalonia, Spain

Jordi Paillisse-Vilanova
jordi.paillisse@upc.edu

Barcelona Neural Networking Center,
Universitat Politècnica de Catalunya
Barcelona, Catalonia, Spain

Albert López-Brescó
albert.lopez@upc.edu

Barcelona Neural Networking Center,
Universitat Politècnica de Catalunya
Barcelona, Catalonia, Spain

Pere Barlet-Ros
pere.barlet@upc.edu

Barcelona Neural Networking Center,
Universitat Politècnica de Catalunya
Barcelona, Catalonia, Spain

Albert Cabellos-Aparicio
alberto.cabellos@upc.edu

Barcelona Neural Networking Center,
Universitat Politècnica de Catalunya
Barcelona, Catalonia, Spain

Abstract

Network modeling is crucial for the design, management, and optimization of modern telecommunications and data networks. Recent advancements in Machine Learning (ML), specifically Graph Neural Networks (GNNs), offer promising solutions but rely on parameterized traffic representations, necessitating retraining for new traffic patterns. This paper explores integrating the Discrete Wavelet Transform (DWT) with GNNs to enhance network traffic modeling. By leveraging wavelets, which decompose signals into both time and frequency components, we aim to encode traffic patterns without assuming specific distributions, improving model adaptability and accuracy. We modify the state-of-the-art RouteNet-Fermi model to incorporate wavelet-based traffic encoding and evaluate its performance across different synthetic and real traffic scenarios. Our findings show that wavelet-based encoding handles unseen traffic distributions with minimal impact on performance, unlike traditional parameter-based approaches. This work represents a step forward in bridging the gap between non-parametric traffic representation and advanced network modeling, offering a promising solution for dynamic and complex network environments.

CCS Concepts

• **Networks** → **Network performance modeling**; • **Computing methodologies** → **Feature selection**; *Supervised learning by regression*; *Neural networks*.

Keywords

Deep Learning, Discrete Wavelet Decomposition, Graph Neural Networks, Machine Learning, Network Modelling, Wavelets

1 Introduction

Network modeling is a crucial aspect of modern telecommunications and data networks. It provides a framework for understanding and predicting network behavior under various conditions, enabling efficient design, management, and optimization of network resources. Accurate network models are vital for ensuring reliable and high-performance communication services, which are increasingly essential in our interconnected world, and to address challenges like congestion [31], latency [6], and fault tolerance [4].

Traditional network modeling approaches include packet-event simulators [23, 30] and analytical models like Queuing Theory and Network Calculus [2, 26]. On one hand, packet-event simulators are known to be accurate yet computationally expensive. Despite recent efforts to address this through parallelization and integration with Machine Learning (ML) models [15, 33], these still require powerful hardware when applied onto larger flows and networks. Analytical models, although significantly faster, often lack accuracy when simulating networks that do not adhere to their assumptions, such as Poisson arrival times for Queueing Theory [12].

Recently, flow-level Machine Learning (ML) models, and more specifically Deep Learning (DL), have emerged as promising solutions that offer both speed and accuracy. In particular, Graph Neural Network (GNN) models have been successful due to their ability to generalize to unseen network topologies during training [13, 18]. However, these models typically rely on parameterized approaches to represent network traffic, requiring the traffic to be adjusted or approximated to fit a predefined distribution. This limitation is significant because real-world network traffic is highly variable and often does not conform to these standard distributions [24, 32]. Additionally, these models must be retrained whenever they encounter new traffic patterns, which can be time-consuming and computationally expensive. This retraining requirement hinders their ability to quickly adapt to changing network conditions and limits their practical applicability in dynamic and unpredictable environments, as the model remains unreliable until it can be adjusted (e.g. a spike in network traffic due to a node failure).

Representing arbitrary traffic patterns can be challenging due to their complexity and variability. One common approach for analyzing signals is the Fourier Transform, which decomposes signals into their frequency components. However, the Fourier Transform does not consider the time domain, meaning it cannot capture when certain frequencies occur. This limitation makes it less suitable for non-stationary signals, where frequency components can change over time. Alternatively, wavelets have emerged as a powerful method for signal analysis [17, 21, 28]. Unlike the Fourier Transform, wavelets decompose signals into both time and frequency components, allowing for a more comprehensive representation. The wavelet transform enables multi-resolution analysis, capturing

both high-frequency details and low-frequency trends in the signal. This ability to provide time-localized frequency information makes wavelets particularly well-suited for analyzing dynamic and non-stationary signals such as network traffic.

In this paper, we explore whether wavelets can be effectively used to encode arbitrary input traffic for use in modern GNN models. Specifically, we modify one of the state-of-the-art flow-level network models, RouteNet-Fermi [13], to encode flows using the wavelet decomposition of the sequences of packets that form it. This results in a robust and adaptive network model capable of handling diverse traffic patterns without the need for extensive retraining. We evaluate our approach by training both the original and modified versions of RouteNet-Fermi with samples belonging to two parametric traffic distributions and then testing it on samples from a third non-parametric distribution. Our results show that: (i) the wavelet-based RouteNet-Fermi is capable of achieving similar performance to the parameter-based model, and, more importantly, (ii) it can keep the same performance when presented with flows whose traffic distribution is different from ones in the training set.

The paper is structured as follows: Section 2 introduces wavelets and the Discrete Wavelet Transform (DWT). Section 3 then describes our proposal to expand RouteNet-Fermi with DWT-transformed data. Finally, in Section 4 we study the impact of DWT's most important hyperparameters over the model's error, and we compare our proposal against the original parameter-based approach.

2 Background: Wavelet Transform

The wavelet transform is a process that allows for decomposing signals in both the frequency and temporal domains. This is done through the use of wavelets, originally defined as "harmonic oscillations" that can be shaped to accommodate different resolutions [14, 16]. This contrasts with the Fourier transform, which describes signals solely in the frequency domain. Furthermore, while other forms of time-frequency decomposition exist, the wavelet transform supports multi-level decomposition, allowing the examination of the signal at different resolutions, unlike alternatives such as the Short-Time Fourier Transform. The wavelet transform (WT) for signal f is expressed through either of the following equations:

$$WT_f(t, s) = \frac{1}{\sqrt{|s|}} \int f(x) \Psi\left(\frac{x-t}{s}\right) dx$$

$$WT_f(t, s) = \frac{1}{\sqrt{|s|}} \sum_{i=0}^{|f|-1} f[x_i] \Psi\left(\frac{x_i - t_i}{s}\right)$$

As it can be seen, the wavelet transform can be defined as the convolution between the original signal and the wavelet function Ψ . As a convolution, the operation can be applied to continuous signals, as in the first equation, or discrete sequences, as in the second; in the case of the latter f is represented as an array of length $|f|$. Since the wavelet function must follow a set of properties for the correct application of the WT [11, 22], these functions are usually selected from a set of well-known wavelet families [3, 10, 29].

The wavelet transform requires two inputs: the temporal translation to obtain the convolution value at time t , and the scaling factor s , which applies a temporal dilation of the wavelet. Using different values of s will affect how the wavelet interacts with the signal's components. For example, small values of s shorten the

wavelet, which allows it to better capture the temporal resolution around t and high-frequency components of the signal. By contrast, large values of s widen the wavelet, which makes it better when capturing the low-frequency components of the signal.

However, the issue with the continuous wavelet transform is the amount of information returned by the process, which should instead be consolidated. Hence, the discrete wavelet transform (DWT) is a version of the WT that uses values of s and t based on powers of two (dyadic) to balance accuracy across scales and efficiency in the number of features [10, 22]:

$$DWT_f(a, b) = \frac{1}{\sqrt{|b|}} \sum_{i=0}^{|f|-1} f[x_i] \Psi\left(\frac{x_i - a}{b}\right)$$

Under this equation, the t and s are replaced by $a = a_0 2^j$ and $b = 2^j$ respectively, where j indicates the level of the decomposition.

3 Methodology

Figure 1 shows a summary of the current and proposed methods for flow encoding. The current approach encodes each flow as an array of features. For example, for On/Off traffic, we specify the duration of the "on" and "off" periods duration, or in Poisson traffic, the λ parameter (1). Then, we normalize these features with the Z-score (2) and use a Multilayer Perceptron (MLP) (3) to generate the flow state that will be used by RouteNet-Fermi.

In contrast, the proposed approach begins by extracting the packet sequences from packet traces and calculating the Inter-Packet Gap (IPG) and packet size for each packet in the flow (1), which are then normalized using the Z-score (2). After this step, we have a sequence of pairs (IPG, packet size) for each flow in the network. The next step involves applying the DWT to the sequence of IPGs and packet sizes at different decomposition levels to extract the desired coefficients (3). At this point, the process diverges depending on whether we use a single level or multiple levels of decomposition. If only one level is used, the resulting coefficients, encoded as a sequence, are inputted into a Gated Recurrent Unit (GRU) [8] to generate the encoding of a flow state (4). If multiple levels of decomposition are used, each level, defined as a different-sized sequence of coefficients, is processed by its own GRU (4). The outputs from each GRU are then concatenated and summarized using an MLP (5), which returns the encoding of the desired shape. This encoding process is repeated for all flows in the network.

3.1 Efficient Multilevel DWT: Mallat decomposition

To perform DWT at multiple decomposition levels efficiently, we use the Mallat decomposition algorithm [22]. It uses the DWT as a band-pass filter by adjusting the wavelet function and the level of decomposition j used. The algorithm proceeds as follows: it takes as input a signal f , represented as a one-dimensional sequence, and a maximum level of decomposition l as an hyperparameter. At the first level, the original signal is transformed using a high pass and a low pass filter (implemented through DWT). The signal resulting from the low pass filter at level j is referred to as the approximation signal A_j , and the signal resulting from the high pass filter is referred to as the detail signal D_j . During the process, the number of elements in each signal is halved. Furthermore, each

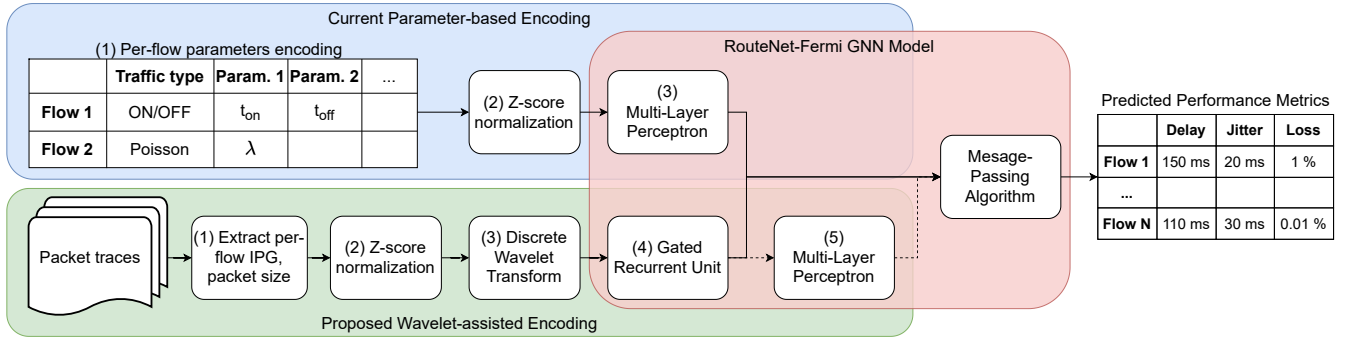


Figure 1: Current approach to encode traffic characteristics (top) versus proposed (bottom). Step (5) of the proposed wavelet-assisted encoding is only applied when using the outputs of multiple decomposition levels.

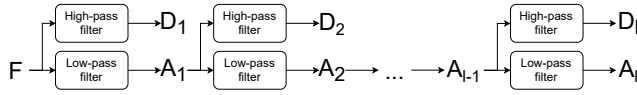


Figure 2: Visualization of Mallat decomposition algorithm

level j after the first uses the previous approximation signal A_{j-1} as input. The algorithm returns A_I , representing an approximation of the original signal f , and all the detail signals $[D_1 \dots D_I]$, each representing the impact of the harmonic components identified at each level. This process is visualized in Figure 2.

The computational cost of the decomposition is $O(n)$, where n is the sequence length, independent of the maximum level of decomposition. This efficiency arises because, at each decomposition level, the sequence length is halved, thereby halving the number of operations. This is similar to the cost of obtaining the parameters in parameter encoding if these are unknown and need to be approximated from the packet traces. However, using the GRU for summarizing the wavelet coefficients increases the inference cost of the ML model, which will be explored later in the evaluation.

3.2 Wavelet Selection

As for the wavelets used in the DWT, while our methodology is compatible with any appropriate wavelet, we consider wavelets belonging to these three families:

(1) **Daubechies** (db) [10]: One of the first families of wavelets developed, and hence one of the most widely used. They are designed to contain multiple vanishing moments, which allows capturing more complex signal components. The naming scheme indicates the number of these vanishing moments (e.g., the "db2" wavelet is comprised of two vanishing moments).

(2) **Coiflets** (coif) [3, 10]: Also designed by Daubechies, these wavelets are designed to have an increased quantity of vanishing moments, making them suitable for tasks requiring higher accuracy in both time and frequency domains. The naming convention indicates that the "coifN" wavelet contains $2N$ vanishing moments (e.g., the "coif2" wavelet is comprised of four vanishing moments).

(3) **Biorthogonal** (bior) [29]: This family of wavelets is characterized by using different wavelets for the decomposition and reconstruction of the original signal, as well as providing linear phase

characteristics, properties beneficial in signal processing tasks. The naming convention indicates the number of vanishing moments in both the decomposition and reconstruction wavelets. For example, "bior1.3" indicates one vanishing moment in the decomposition wavelet and three in the reconstruction wavelet.

3.3 Implementation

All the code was written using Python 3.10, and our version of RouteNet-Fermi was implemented using TensorFlow 2.15 [1]. We used the DWT implementation from PyWavelets 1.6 [19]. RouteNet-Fermi's original hyperparameters were kept. The DWT was performed using the Haar wavelet (equivalent to "db1") and the "zero" mode for padding, to then extract the approximation signal after 10 levels of decomposition, unless specified otherwise. All the following experiments were conducted on a server with an AMD Ryzen 9 3950X 16-Core 32 Threads 3.5 GHz CPU and 64 GiB of RAM, running Ubuntu 22.04 LTS. We published our modified RouteNet-Fermi's implementation in a public repository¹.

4 Evaluation

In this work, we aim to answer the following questions relating to the use of the DWT as a data preprocessing method:

- (1) What is the impact of the wavelet chosen?
- (2) What is the impact of the decomposition level, and does combining different levels improve the model's accuracy?
- (3) How does using DWT compare to using RouteNet-Fermi's default flow encoding method? Specifically, does DWT improve RouteNet-Fermi's predictions when working with unseen traffic distributions?

To address these questions, we implemented a modified version of RouteNet-Fermi, starting from their public repository². We evaluated the effectiveness of the DWT and its chosen hyperparameters by measuring how well the enhanced model predicts the average packet delay of flows within a network scenario. Each network scenario consists of a network experiencing for 4 seconds of unrecorded traffic (windup for the network buffers to stabilize) followed by another 4 seconds of recorded traffic. The data was

¹<https://github.com/BNN-UPC/Papers/wiki/Wavelet-Enhanced-Graph-Neural-Networks>

²<https://github.com/BNN-UPC/RouteNet-Fermi>

Distribution	Poisson	On/Off	MAWI	Total
Training	3012	3070	3059	9141
Validation	604	629	594	1827
Test	418	407	393	1218
Total	4034	4106	4046	12186

Table 1: Number of samples per distribution and partition.

Wavelet	MAPE (%)		Wavelet	MAPE (%)		Wavelet	MAPE (%)	
	Validation	Test		Validation	Test		Validation	Test
db1	13.737	13.881	db20	13.615	13.740	bior2.2	13.678	13.851
db2	13.689	13.816	coif1	13.658	13.810	bior2.6	13.634	13.758
db4	13.725	13.852	coif2	<u>13.611</u>	13.739	bior3.1	13.769	13.912
db8	13.664	13.777	coif4	13.686	13.827	bior4.4	13.638	13.792
db12	13.685	13.828	bior1.1	13.737	13.881	bior5.5	13.620	13.745
db16	13.684	13.818	bior1.5	13.670	13.776	bior6.8	13.661	13.816

Table 2: Impact of wavelet selection over model error.

generated by running traffic traces into an 8-node network. We considered three types of scenarios according to the time distribution that describes its packet traffic:

- **Poisson Distribution:** Equal-sized packets are generated following a synthetic Poisson distribution.
- **On/Off Distribution:** Equal-sized packets are generated in alternating periods of constant bit rate activity ("on" period) and inactivity ("off" period).
- **MAWI PCAPs:** Traces are generated from the MAWI Working Group Traffic Archive [7].

The scenarios also vary in terms of the topology of the computer network, the number of flows generated, and their routing. Furthermore, the data samples were split into training, validation, and test partitions to avoid bias in the evaluation. The number of samples generated is summarized in Table 1.

4.1 Impact of the chosen wavelet

We begin by examining how the type of wavelet affects the effectiveness of the DWT when encoding traffic flows. Table 2 shows the impact of different wavelets on the Mean Absolute Percentage Error (MAPE) of RouteNet-Fermi when comparing the predicted average packet delay per flow against the true measured value. The table presents results for both the validation and test partitions. The explored wavelets are examples from the wavelet families described in subsection 3.2.

The results show that all the wavelets tested performed similarly, with the test MAPE ranging between 13.739% and 13.912%. Furthermore, the difference between the validation and test performance of each wavelet is minimal, with differences of 0.1% to 0.2% between the two, indicating good generalization to unseen samples. While experimenting with different wavelets may be useful during hyperparameter tuning to maximize the model's accuracy, these results suggest that, in general, any wavelet is appropriate.

Signal Used	Decomposition levels	MAPE (%)		Inference cost per sample (ms)
		Validation	Test	
Single approximation signal	4	13.669	13.842	1016
	6	13.710	13.865	267
	8	13.709	13.860	76
	10	13.737	13.881	<u>30</u>
Signal detail signal	4	13.680	13.833	1011
	6	13.646	13.792	267
	8	13.656	13.807	77
	10	13.761	13.885	<u>30</u>
Multiple detail signals	4, 6, 8 and 10	13.655	13.803	1025
Multiple approximation and detail signals	Detail signals 4, 6, 8 and 10 Approximation signal: 10	<u>13.569</u>	13.668	1033

Table 3: Impact of decomposition level and signal selection.

4.2 Impact of the level of decomposition

Next, Table 3 shows how using different signals at various decomposition levels affects the results. We tested the use of signals at different decomposition levels, as well as the use of multiple detail signals combined since the properties of the DWT ensure these are independent of each other. The results indicate that the level of approximation barely affected the error of the model when predicting packet delay per flow, with all errors being around 13.8% MAPE. The model with the highest amount of information available achieved the lowest error, but only marginally, at 13.6% MAPE.

As explained in Section 3, the sequences generated by the DWT are processed by a GRU model. The results suggest that the GRU is as effective in summarizing these sequences as using a higher level of decomposition. However, this results in a higher inference cost, as the sequences generated by the DWT double per each decrease in the level of decomposition. Given the negligible impact on the model's error, it is recommended to use the highest level of decomposition available to minimize the sequence length and therefore reduce the inference time. This approach balances computational efficiency and model accuracy, making it practical for real-time network traffic prediction scenarios.

4.3 Comparison against the current parameter-based encoding

Finally, we evaluate how wavelet decomposition flow encoding compares against the original, parameter-based method. In the original parameter-encoding approach, the model has access to the flow's bandwidth, packet rate, distribution type (represented through one-hot encoding), the λ parameter in Poisson distributions, and the "on" period duration, "off" period duration, and "on" period bit-rate in On/Off distributions. For the wavelet decomposition encoding, we evaluate both single- and multi-level approaches. The single-level approach uses the "coif2" wavelet to extract the approximation signal after 10 levels of decomposition, as discussed in subsection 4.1. The multi-level approach uses the "db1" wavelet with multiple levels of decomposition, as detailed in subsection 4.2.

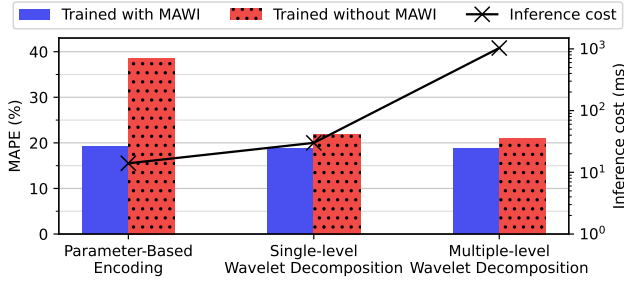


Figure 3: MAWI prediction error and inference cost of Parameter-Based encoding (RouteNet-Fermi) versus Wavelet Decomposition. In "trained without MAWI" models are trained only with Poisson and On/Off samples, while in "trained with MAWI" with all three distributions.

Figure 3 illustrates the MAPE of all three approaches when predicting the average packet delay per flow in the MAWI PCAPs samples. In the "trained with MAWI" values, the dataset used to train RouteNet-Fermi includes MAWI samples (which are independent of the samples used for evaluation). In the "trained without MAWI" values, the dataset used to train RouteNet-Fermi includes only samples from the Poisson and On/Off distributions. The figure also shows the average per-sample inference cost for each approach.

As shown, when the model using parameter-based encoding attempts to predict the MAWI PCAPs dataset, its error increases from 19.17% to 38.4%, an 100% increase. Conversely, both wavelet approaches prove more resilient, with the error increasing from 18.8% to 21.0%-21.9%, roughly a 10% increase. These results also show that when the MAWI samples are not seen during training, the wavelet encoding decreases the prediction error by 45%. Overall, the parameter-based encoding fails to describe unseen distributions, resulting in a significant increase in MAPE, whereas the wavelet encoding method still effectively describes unseen distributions with minimal impact on prediction error.

Furthermore, while both wavelet approaches performed similarly, the multiple-level approach has an average inference cost per sample of over one second, the single-level's cost is comparable to parameter-based encoding, taking 30 ms and 14 ms respectively. This reinforces the conclusions obtained back in subsection 4.1, where the single-level approach obtains the best balance in inference cost and model performance. We also note that, thanks to the efficient $O(n)$ cost of performing the wavelet decomposition, the training cost remains similar as with the parameter-based encoding, in range of one to two hours.

5 Related work

After wavelets and the wavelet transform were introduced, they have been prevalent in a variety of fields, such as chemistry [17], image compression [25] and vehicular traffic prediction [9]. Within computer networking, the wavelet transform has been more prevalent in traffic prediction models [21, 27, 28, 34], while its use in network modeling is more limited. In [20] the authors use DWT and Markov wavelet models for Fractional Gaussian Noise traffic, although they can also be adapted to non-Gaussian traffic. Later,

in [5], a Wavelet-based Rosenblatt synthetic traffic model was designed for non-Gaussian traffic present within Networks-on-Chips.

In comparison, state-of-the-art network models are often based on ML models, Discrete Event Simulation (DES), or a mixture of both. DES-based and DES+ML solutions, such as DONS [15], OM-NeT++ [23], NS3 [30] and DQN [33], take as input the raw packet traces, as they simulate how each packet traverses the network. Alternatively, ML models like RouteNet-Fermi [13] use aggregated characteristics such as per-flow average bandwidth, packet rate, and the parameters of the distributions supported by it.

On one hand, current network models that process data using wavelet models can be applied to traffic with unknown or non-parametric distributions, unlike network models like RouteNet-Fermi. Also, unlike DES-based solutions, these wavelet models remain computationally inexpensive (e.g. in [20], the cost of building the model is linear to the length of the training sequence). However, the biggest limitation of current wavelet models is that they can only model individual flows, while the state-of-the-art alternatives can work with entire network scenarios. This limits their accuracy, as they cannot take into account the impact of the network or other traffic flows, and require performing inference separately for each flow. Ultimately, in this work, we explore how to bridge the differences between both approaches by strengthening one of the non-wavelet solutions using the DWT in its preprocessing.

6 Conclusion

In conclusion, in this paper, we showed how flow-level network models, specifically RouteNet-Fermi, can use the DWT to encode and work with non-parametric network traffic. We also demonstrated how the different hyperparameters of the DWT algorithm may affect the model's accuracy and cost. The combined approach suggests that exploiting GNN's ability to generalize to unseen network topologies with wavelet signal decomposition allows the generation of network models that reduce the burden of needing to retrain the model to adapt to new scenarios.

While our proposed methodology demonstrates a promising approach, there are several avenues for future research and improvement. For instance, instead of using the DWT, the Stationary Wavelet Transform could be explored, as it has shown advantages in traffic prediction [28]. Furthermore, integrating wavelet transformations directly into the ML architecture offers exciting possibilities, such as Wavelet Networks [35] and the Graph Wavelet Gated Recurrent Network [9], as these advancements enhance the model's performance and adaptability.

Acknowledgments

This publication is part of the Spanish I+D+i project TRAINER-A (ref.PID2020-118011GB-C21), funded by MCIN/AEI/10.13039/501100011033. This work is also partially funded by the Catalan Institution for Research and Advanced Studies (ICREA). Carlos Güemes is funded by the Joan Oró predoctoral program, from the Secretariat for Universities and Research, part of the Ministry of Research and Universities of the Government of Catalonia, and the European Social Fund Plus (ref. BDNS 657443). Jordi Paillisse is funded by NextGen EU, Ministry of Universities and Recovery,

Transformation and Resilience Plan, through a call from UPC (Grant Ref. 2022 UPC-MSC-93871).

References

- [1] Martín Abadi et al. 2016. TensorFlow: A System for Large-Scale Machine Learning. In *12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16)*. USENIX Association, Savannah, GA, 265–283. <https://www.usenix.org/conference/osdi16/technical-sessions/presentation/abadi>
- [2] Mina Tahmasbi Arashloo, Ryan Beckett, and Rachit Agarwal. 2023. Formal methods for network performance analysis. In *20th USENIX Symposium on Networked Systems Design and Implementation (NSDI 23)*. 645–661.
- [3] G. Beylkin, R. Coifman, and V. Rokhlin. 1991. Fast wavelet transforms and numerical algorithms I. *Communications on Pure and Applied Mathematics* 44, 2 (1991), 141–183. <https://doi.org/10.1002/cpa.3160440202> arXiv:<https://onlinelibrary.wiley.com/doi/pdf/10.1002/cpa.3160440202>
- [4] Peter Bodik, Ishai Menache, Mosharaf Chowdhury, Pradeepkumar Mani, David A. Maltz, and Ion Stoica. 2012. Surviving failures in bandwidth-constrained datacenters. *SIGCOMM Comput. Commun. Rev.* 42, 4 (aug 2012), 431–442. <https://doi.org/10.1145/2377677.2377760>
- [5] Amit Chaurasia, Vijay Shankar Sharma, Chiranjil Lal Chowdhary, Shakila Basheer, and Thippa Reddy Gadekallu. 2023. Non-Gaussian Traffic Modeling for Multicore Architecture Using Wavelet Based Rosenblatt Process. *IEEE Access* 11 (2023), 38523–38533. <https://doi.org/10.1109/ACCESS.2023.3265572>
- [6] Li Chen, Justinas Lingys, Kai Chen, and Feng Liu. 2018. AuTO: Scaling Deep Reinforcement Learning for Datacenter-Scale Automatic Traffic Optimization. In *Proceedings of the 2018 Conference of the ACM Special Interest Group on Data Communication (Budapest, Hungary) (SIGCOMM '18)*. Association for Computing Machinery, New York, NY, USA, 191–205. <https://doi.org/10.1145/3230543.3230551>
- [7] Kenjiro Cho, Koushirou Mitsuya, and Akira Kato. 2000. Traffic data repository at the WIDE project. In *Proceedings of the Annual Conference on USENIX Annual Technical Conference (San Diego, California) (ATEC '00)*. USENIX Association, USA, 51.
- [8] Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078* (2014).
- [9] Zhiyong Cui, Ruimin Ke, Ziyuan Pu, Xiaolei Ma, and Yinhai Wang. 2020. Learning traffic as a graph: A gated graph wavelet recurrent neural network for network-scale traffic prediction. *Transportation Research Part C: Emerging Technologies* 115 (2020), 102620. <https://doi.org/10.1016/j.trc.2020.102620>
- [10] Ingrid Daubechies. 1988. Orthonormal bases of compactly supported wavelets. *Communications on Pure and Applied Mathematics* 41, 7 (1988), 909–996. <https://doi.org/10.1002/cpa.3160410705> arXiv:<https://onlinelibrary.wiley.com/doi/pdf/10.1002/cpa.3160410705>
- [11] I. Daubechies. 1990. The wavelet transform, time-frequency localization and signal analysis. *IEEE Transactions on Information Theory* 36, 5 (1990), 961–1005. <https://doi.org/10.1109/18.57199>
- [12] Miquel Ferriol-Galmés, Krzysztof Rusek, José Suárez-Varela, Shihan Xiao, Xiang Shi, Xiang Cheng, Bo Wu, Pere Barlet-Ros, and Albert Cabellos-Aparicio. 2022. Routenet-erlang: A graph neural network for network performance evaluation. In *IEEE INFOCOM 2022-IEEE Conference on Computer Communications*. IEEE, 2018–2027.
- [13] Miquel Ferriol-Galmés, Jordi Paillisse, José Suárez-Varela, Krzysztof Rusek, Shihan Xiao, Xiang Shi, Xiang Cheng, Pere Barlet-Ros, and Albert Cabellos-Aparicio. 2022. RouteNet-Ferri: Network Modeling with Graph Neural Networks. arXiv:2212.12070 [cs.NI]
- [14] D. Gabor. 1946. Theory of communication. Part 1: The analysis of information. *Journal of the Institution of Electrical Engineers - Part III: Radio and Communication Engineering* 93 (November 1946), 429–441(12). Issue 26. <https://digital-library.theiet.org/content/journals/10.1049/ji-3-2.1946.0074>
- [15] Kaihui Gao, Li Chen, Dan Li, Vincent Liu, Xizheng Wang, Ran Zhang, and Lu Lu. 2023. DONS: Fast and Affordable Discrete Event Network Simulation with Automatic Parallelization. In *Proceedings of the ACM SIGCOMM 2023 Conference (New York, NY, USA) (ACM SIGCOMM '23)*. Association for Computing Machinery, New York, NY, USA, 167–181. <https://doi.org/10.1145/3603269.3604844>
- [16] P. Goupillaud, A. Grossmann, and J. Morlet. 1984. Cycle-octave and related transforms in seismic signal analysis. *Geoexploration* 23, 1 (1984), 85–102. [https://doi.org/10.1016/0016-7142\(84\)90025-5](https://doi.org/10.1016/0016-7142(84)90025-5) Seismic Signal Analysis and Discrimination III.
- [17] Vu Dang Hoang. 2014. Wavelet-based spectral analysis. *TrAC Trends in Analytical Chemistry* 62 (2014), 144–153. <https://doi.org/10.1016/j.trac.2014.07.010>
- [18] Benedikt Jaeger, Max Helm, Lars Schwegmann, and Georg Carle. 2022. Modeling TCP performance using graph neural networks. In *Proceedings of the 1st International Workshop on Graph Neural Networking (Rome, Italy) (GN-Net '22)*. Association for Computing Machinery, New York, NY, USA, 18–23. <https://doi.org/10.1145/3565473.3569190>
- [19] Gregory R. Leary. 2019. PyWavelets: A Python package for wavelet analysis. *Journal of Open Source Software* 4, 36 (2019), 1237. <https://doi.org/10.21105/joss.01237>
- [20] Sheng Ma and Chuanyi Ji. 2001. Modeling heterogeneous network traffic in wavelet domain. *IEEE/ACM Transactions on Networking* 9, 5 (2001), 634–649. <https://doi.org/10.1109/90.958331>
- [21] Rishabh Madan and Partha Sarathi Mangipudi. 2018. Predicting Computer Network Traffic: A Time Series Forecasting Approach Using DWT, ARIMA and RNN. In *2018 Eleventh International Conference on Contemporary Computing (IC3)*. 1–5. <https://doi.org/10.1109/IC3.2018.8530608>
- [22] S.G. Mallat. 1989. A theory for multiresolution signal decomposition: the wavelet representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 11, 7 (1989), 674–693. <https://doi.org/10.1109/34.192463>
- [23] J.C. Neumann. 2015. *The Book of GNS3: Build Virtual Network Labs Using Cisco, Juniper, and More*. No Starch Press. <https://books.google.es/books?id=9wcvDwAAQBAJ>
- [24] J Popoola and RA Ipinyomi. 2017. Empirical performance of weibull self-similar tele-traffic model. *International Journal of Engineering and Applied Sciences* 4, 8 (2017), 257389.
- [25] Majid Rabbani and Rajan Joshi. 2002. An overview of the JPEG 2000 still image compression standard. *Signal Processing: Image Communication* 17, 1 (2002), 3–48. [https://doi.org/10.1016/S0923-5965\(01\)00024-8](https://doi.org/10.1016/S0923-5965(01)00024-8) JPEG 2000.
- [26] Thomas G Robertazzi. 2000. *Computer networks and systems: queueing theory and performance evaluation*. Springer Science & Business Media.
- [27] Sajal Saha and Anwar Haque. 2023. Wavelet-Based Hybrid Machine Learning Model for Out-of-distribution Internet Traffic Prediction. In *NOMS 2023-2023 IEEE/IFIP Network Operations and Management Symposium*. 1–8. <https://doi.org/10.1109/NOMS56928.2023.10154337>
- [28] Sajal Saha, Moinul Islam Sayed, and Anwar Haque. 2023. Empirical Mode Decomposition and Stationary Wavelet Transformation in Internet Traffic Prediction. In *IEEE INFOCOM 2023 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*. 1–6. <https://doi.org/10.1109/INFOCOMWKSHPS57453.2023.10226162>
- [29] Mallat Stéphane. 2009. CHAPTER 7 - Wavelet Bases. In *A Wavelet Tour of Signal Processing (Third Edition)* (third edition ed.), Mallat Stéphane (Ed.). Academic Press, Boston, 263–376. <https://doi.org/10.1016/B978-0-12-374370-1.00011-2>
- [30] Andras Varga. 2019. *A Practical Introduction to the OMNeT++ Simulation Framework*. Springer International Publishing, Cham, 3–51. https://doi.org/10.1007/978-3-030-12842-5_1
- [31] Michael Welzl. 2005. *Network congestion control: managing internet traffic*. John Wiley & Sons.
- [32] Zhiyuan Xu et al. 2018. Experience-driven networking: A deep reinforcement learning based approach. In *IEEE INFOCOM*. 1871–1879.
- [33] Qingqing Yang, Xi Peng, Li Chen, Libin Liu, Jingze Zhang, Hong Xu, Baochun Li, and Gong Zhang. 2022. DeepQueueNet: Towards Scalable and Generalized Network Performance Estimation with Packet-Level Visibility. In *Proceedings of the ACM SIGCOMM 2022 Conference (Amsterdam, Netherlands) (SIGCOMM '22)*. Association for Computing Machinery, New York, NY, USA, 441–457. <https://doi.org/10.1145/3544216.3544248>
- [34] Kun Zhang, Yinping Chai, and Xing-an Fu. 2012. A network traffic prediction model based on recurrent wavelet neural network. In *Proceedings of 2012 2nd International Conference on Computer Science and Network Technology*. 1630–1633. <https://doi.org/10.1109/ICCSNT.2012.6526232>
- [35] Qinghua Zhang and Albert Benveniste. 1992. Wavelet networks. *IEEE transactions on Neural Networks* 3, 6 (1992), 889–898.