QoE-Critical Learning-based Stream Scheduling in Multipath-QUIC

Submitted on: **DD-MM-YYYY**

Mees Apeldoorn
Mees.Apeldoorn@student.uva.nl
University of Amsterdam
Amsterdam, The Netherlands

Hongyun Liu
h.liu@uva.n
University of Amsterdam
Amsterdam, The Netherlands

1 INTRODUCTION

The quality of experience (QoE) of streaming services has drawn a lot of attention in the age of rapidly increasing online content consumption. Efficient stream scheduling is a key determinant of QoE, prompting a deeper exploration into the underlying transport protocols. This thesis addresses the fundamental optimization of stream scheduling by examining the transport protocol, with a particular focus on the emerging protocol, QUIC [9]. A cutting-edge transport protocol that has garnered attention for its high performance and unique architecture. Operating at the application layer and relying on the User Datagram Protocol (UDP) [2], QUIC distinguishes itself by reducing latency in connection instantiation and ensuring security by default. This protocol serves as the foundation for multipath-QUIC (MPQUIC) [5]. MPQUIC builds upon the QUIC protocol by introducing multipath capabilities. This means that it can utilize multiple network paths simultaneously, potentially leading to improved performance, robustness, and resilience against network failures. The idea is to distribute the data across multiple paths, allowing for more efficient data transmission.

The current multipath schedulers adhere to predetermined policies or rely on machine learning (ML) approaches like reinforcement learning. While schedulers without learning exhibit rapid adaptation to network conditions, the accuracy of these adaptations may fall short of desired levels, resulting in a reduced Quality of Experience (QoE). Machine learning mitigates this challenge by providing a more precise adaptation to current network conditions, albeit with a trade-off of slower adaptation speed. The objective of this thesis is to develop a multipath scheduler that combines the accuracy of an ML model with the swiftness of traditional schedulers, achieving both high precision and rapid adaptation to enhance QoE.

One promising method that could reduce the time it takes these ML models to adapt is via meta-learning [7]. Previous research [3, 8] has explored the application of meta-learning [6] for various Learning based Schedulers' tasks, and an attempt has been made to apply meta-learning to enhance MPQUIC scheduling. Still, it falls short of providing sufficient advancements to replace the current baseline [1] entirely.

the question remains: How can the adaptation time of meta-learning-enhanced MPQUIC schedulers be improved while keeping the adaptation accurate? Some additional questions that can help us answer the research questions are:

- What meta-learning approach would be most effective in enhancing the performance of MPQUIC schedulers?
- How could the implementation of a hybrid online-offline base setup contribute to the improvement of performance in meta-learning-enhanced MPQUIC scheduling?
- In the context of video streaming, what comparative analysis can be conducted to determine the most suitable meta-learning approach for optimizing MPQUIC scheduling?
- What strategies can be identified through further research to enhance the performance of meta-learning-enhanced MPQUIC schedulers beyond the limitations observed in previous studies?

2 RELATED WORK

2.1 QUIC

QUIC [2], which stands for Quick UDP Internet Connections, is a transport layer protocol designed to provide a secure and efficient communication channel over the Internet. It was developed by Google and first announced in 2013, to overcome some of the limitations and performance bottlenecks associated with traditional transport protocols like TCP (Transmission Control Protocol).

Multipath-QUIC (MPQUIC) [5] specifically addresses the use of multiple network paths simultaneously. It allows a QUIC connection to utilize multiple network paths between the client and server concurrently, which can lead to improved reliability and performance. This is particularly beneficial in situations where there may be network congestion or the quality of a single path is variable. Multipath communication can help distribute the data across different paths, making the protocol more robust to packet loss or changes in network conditions.

2.2 SAILfish

SAILfish [1], an innovative learning stream-based scheduling system designed for Multipath QUIC. SAILfish adopts a neural network approach, utilizing state-of-the-art Deep Reinforcement Learning (DRL) algorithms to acquire an efficient scheduling policy. In addition to stream-aware scheduling, SAILfish constitutes a distributed networking system comprising specific modules: (1) MPQUIC server(s), (2) Active Traffic Monitor, and (3) scheduling agent.

The authors present their system design along with inherent quality attributes. Furthermore, they implement a prototype of SAILfish and assess its performance advantages in terms of per-packet and stream-based adaptation compared to a lowest-latency baseline scheduling heuristic. To enhance user Quality of Experience (QoE), experiments are conducted on stream completion times.

SAILfish exhibits comparable performance to a per-packet scheduler, attributed to substantial differences in scheduling approaches (stream-based versus packet-based mechanisms). Conversely, when comparisons are made at the same level, SAILfish notably surpasses stream-based heuristic baselines.

2.3 Meta-Learning

Meta-learning [6] has the potential to enhance scheduler performance, but on its own, it falls short of providing sufficient improvement to supplant the existing baseline. Consequently, additional research is required to enhance other facets of the model and surpass the performance of the baseline. One avenue for improvement involves shifting from time-based scheduling to a per-packet approach or modifying the state and reward mechanisms. Exploring alternative meta-learning approaches within this environment is also worth considering. Ultimately, experimenting with a different base model, such as a hybrid online-offline setup observed in cutting-edge research, presents a promising direction for model improvement and a more comprehensive assessment of the utility of meta-learning.

One article examined the use of Meta-Learning for MPQUIC. [7]. The "FALCON" scheduler, an on-and offline learning-based reinforcement learning paradigm, is proposed in this representative work. It is predicated on the creation of a batch of previously trained models, or meta-models, using the meta-learning technique. When a network change is observed, these are then bootstrapped for a particular network situation, acting and getting feedback that is used to define other

3 METHODOLOGY

3.1 Literature review and Research Landscape

Before getting into our proposed changes, we must first conduct a thorough examination of the current literature to discover recent advances in meta-learning for MPQUIC scheduling. This review seeks to investigate any innovations, approaches, or paradigms that have emerged in recent years. This stage is critical for building on recent developments in the field while ensuring the relevance and freshness of our contributions.

3.2 setup

The foundation of our research lies in SailFish [1], a comprehensive model that comprises MPQUIC servers, a scheduling agent, and a Traffic Monitor. SailFish's use of the MPQUIC protocol and reinforcement learning algorithms for scheduler optimization makes it an ideal starting point. Our enhancements will be integrated into this base model to improve the efficiency of stream transmission paths.

SailFish is made up out of the following components:

- **MPQUIC Server:** The MPQUIC server, following the MPQUIC protocol, delegates scheduling responsibilities to the agent.
- **Scheduling Agent:** Utilizing reinforcement learning algorithms, the agent optimizes stream transmission paths based on real-time feedback and network conditions.
- **Traffic Monitor:** This component provides network insights to the agent, facilitating informed scheduling decisions by offering valuable data and enabling policy updates.

In a standard scenario, SailFish collaborates with the scheduling agent to enhance the transmission of streams, aiming to elevate the user Quality of Experience (QoE) when delivering content over MPQUIC.

3.3 evaluating

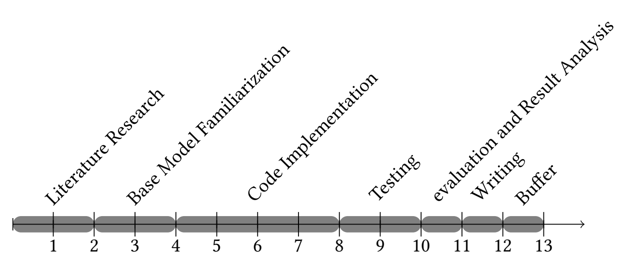
SailFish [1] incorporates an evaluation system, which is built upon the innovative MPQUIC scheduling work known as FStream [4]. This framework provides a foundation for assessing the performance of the scheduling algorithms, and we will leverage it to evaluate our enhancements. To measure the effectiveness of our proposed improvements, we will consider key metrics such as latency, throughput, and overall user QoE. These metrics will be compared against the baseline SailFish model to quantify the impact of our enhancements.

4 RISK ASSESSMENT

While researching meta-learning advancements for MPQUIC schedulers, numerous possible risks must be carefully considered. First and foremost, incorporating new algorithms and modifications into the SailFish model may present unexpected technological issues, risking project timeframes and necessitating considerable debugging efforts. Furthermore, relying on real-world data for validation increases the danger of data inconsistencies or biases, which may limit the generalizability of the proposed improvements. Furthermore, given the dynamic nature of networking protocols and technologies, there is a possibility that external influences will influence the relevance and applicability of our innovations over time. To mitigate these risks, a robust testing and validation plan, ongoing data quality monitoring, and a flexible approach to adapting to emergent technology advances will be used throughout the study process.

5 PROJECT PLAN

The timeline shown below indicates an estimated time per section. during the first weeks, we will Conduct a literature review focusing on recent advancements in meta-learning for MPQUIC schedulers. during the weeks after we will focus on understanding the fundamentals of SailFish, and implementing our improvements. This should be done in a timely manner and leave us with enough time for testing and evaluating the results.



REFERENCES

[1] Marios Evangelos Kanakis and H Bal. 2020. *A Learning-based Approach for Stream Scheduling in Multipath-QUIC.* Ph. D. Dissertation. Universiteit van Amsterdam.

[2] Adam Langley, Alistair Riddoch, Alyssa Wilk, Antonio Vicente, Charles Krasic, Dan Zhang, Fan Yang, Fedor Kouranov, Ian Swett, Janardhan Iyengar, Jeff Bailey, Jeremy Dorfman, Jim Roskind, Joanna Kulik, Patrik Westin, Raman Tenneti, Robbie Shade, Ryan Hamilton, Victor Vasiliev, Wan-Teh Chang, and Zhongyi Shi. 2017. The QUIC Transport Protocol: Design and Internet-Scale Deployment. (2017), 183–196. https://doi.org/10.1145/3098822.3098842

[3] Hongyun Liu, Peng Chen, and Zhiming Zhao. 2021. Towards a robust meta-reinforcement learning-based scheduling framework for time critical tasks in cloud environments. (2021), 637–647.

[4] Xiang Shi, Lin Wang, Fa Zhang, and Zhiyong Liu. 2019. FStream: Flexible Stream Scheduling and Prioritizing in Multipath-QUIC. (2019), 921–924. https://doi.org/10.1109/ICPADS47876.2019.00136

[5] Raza Ul Mustafa, Md Tariqul Islam, Christian Rothenberg, Simone Ferlin, Darijo Raca, and Jason J. Quinlan. 2020. DASH QoE Performance Evaluation Framework with 5G Datasets. (2020), 1–6. https://doi.org/10.23919/CNSM50824.2020.9269111

[6] Jane X Wang. 2021. Meta-learning in natural and artificial intelligence. *Current Opinion in Behavioral Sciences* 38 (2021), 90–95. https://doi.org/10.1016/j.cobeha.2021.01.002 Computational cognitive neuroscience.

[7] Hongjia Wu, Ozgu Alay, Anna Brunström, Giuseppe Caso, and Simone Ferlin. 2022. FALCON: Fast and Accurate Multipath Scheduling using Offline and Online Learning. *CoRR* abs/2201.08969 (2022). arXiv:2201.08969 https://arxiv.org/abs/2201.08969

[8] Luolin Xiong, Yang Tang, Chensheng Liu, Shuai Mao, Ke Meng, Zhaoyang Dong, and Feng Qian. 2023. Meta-Reinforcement Learning-Based Transferable Scheduling Strategy for Energy Management. *IEEE Transactions on Circuits and Systems I: Regular Papers* 70, 4 (2023), 1685–1695.

[9] Zhilong Zheng, Yunfei Ma, Yanmei Liu, Furong Yang, Zhenyu Li, Yuanbo Zhang, Jiuhai Zhang, Wei Shi, Wentao Chen, Ding Li, Qing An, Hai Hong, Hongqiang Harry Liu, and Ming Zhang. 2021. XLINK: QoE-driven multi-path QUIC transport in large-scale video services. (2021), 418–432. https://doi.org/10.1145/3452296.3472893