

# Explorando o overhead de telemetria em redes programáveis na estimativa de QoS com aprendizado de máquina

Rebeca Dantas Cabral<sup>1</sup>, Gabriel Santos de Andrade<sup>2</sup>, Luis Kilmer da Silva Bernardo<sup>1</sup>,  
Pedro Batista de Carvalho Filho<sup>1</sup>, Leandro C. de Almeida<sup>1</sup> e Fábio L. Verdi<sup>2</sup>

<sup>1</sup>Unidade Acadêmica de Informática – Instituto Federal da Paraíba (IFPB)  
João Pessoa – PB – Brasil

<sup>2</sup>Departamento de Computação (Dcomp) – Universidade Federal de São Carlos (UFSCar)  
Sorocaba – SP – Brasil.

rebeca.cabral@academico.ifpb.edu.br, gabrielsa@estudante.ufscar.br,  
luis.kilmer@academico.ifpb.edu.br, pedro.carvalho@ifpb.edu.br,  
leandro.almeida@ifpb.edu.br, verdi@ufscar.br

**Abstract.** *Monitoring the quality of service (QoS) of real-time applications is challenging for network service providers due to limited access to metrics within the user domain. This study evaluates the impact of INT (In-Band Network Telemetry) and ONT (Out-of-Band Network Telemetry) telemetry strategies on the performance of machine learning algorithms for QoS prediction in video services. Experiments conducted on P4-programmable switches showed that the INT approach reduces prediction error by up to 7 times compared to ONT, although it introduces an overhead of 25% compared to the original throughput of the network without INT. The study also defines an overhead metric and makes the implementations available in a public repository.*

**Resumo.** *O monitoramento da qualidade de serviço (QoS) de aplicações em tempo real é desafiador para provedores de serviços de rede devido o acesso limitado às métricas no domínio do usuário. Este estudo avalia o impacto das estratégias de telemetria INT (In-Band Network Telemetry) e ONT (Out-of-Band Network Telemetry) no desempenho de algoritmos de aprendizado de máquina para predição de QoS em serviços de vídeo. Experimentos em switches programáveis com P4 mostraram que a abordagem INT reduz o erro de predição em até 7 vezes em relação à ONT, embora gere um overhead de 25% em relação ao throughput original da rede sem INT. O trabalho também define uma métrica de overhead e disponibiliza as implementações em repositório público.*

## 1. Introdução

O monitoramento da qualidade de serviço (QoS - *Quality of Service*) de aplicações em tempo real é um desafio constante, mobilizando esforços da comunidade científica e da indústria na busca por soluções cada vez mais eficientes. Sob a perspectiva de um provedor de serviços de rede (ISP - *Internet Service Provider*), a tarefa de monitorar, em tempo real, o nível de QoS das aplicações que utilizam sua infraestrutura de rede (como switches e roteadores) apresenta desafios significativos. Isso ocorre porque o operador

da rede não possui acesso às métricas de serviço das aplicações, uma vez que elas estão disponíveis apenas para os usuários finais da rede. Por exemplo, em um cenário de transmissão de vídeo, o operador da rede não consegue obter informações em tempo real sobre a qualidade do vídeo que está sendo reproduzido na aplicação do usuário.

Uma vez que o operador da rede possui acesso apenas às medições realizadas no domínio de sua infraestrutura, surge a hipótese de que, a partir dessas observações (métricas da infraestrutura), seja possível estimar o estado das aplicações (métricas de serviço) que executam no domínio do usuário. Nesse contexto, esforços no campo da predição, utilizando algoritmos de aprendizado de máquina (AM), têm sido explorados como uma solução potencial para esse desafio [Stadler et al. 2017, Calasans 2020].

Dentre os trabalhos recentes, destacamos os estudos de [de Almeida et al. 2021], que utiliza telemetria (*INT - In-band Network Telemetry*) em plano de dados programável em conjunto com modelos de AM para a estimativa de métricas de QoS. O INT é uma estratégia de monitoramento na qual medições sobre o estado da rede são anexadas a todos os pacotes da aplicação que atravessam os dispositivos de rede [P4 2021]. Tais métricas são então utilizadas pelos operadores de redes como dados de entrada para que os algoritmos de AM possam realizar as predições sobre como está a QoS de determinadas aplicações. Como resultado, pode-se então realizar ações na rede a fim de manter a QoS desejada das aplicações.

Embora inovadora, a coleta de telemetria INT implica em uma carga adicional (*overhead*) para realizar a tarefa de monitoramento, já que cada pacote transporta dados extras. Além disso, há um atraso adicional introduzido pelo tempo necessário para que um switch ou roteador execute essa tarefa durante o encaminhamento dos pacotes. Na literatura, encontramos algumas estratégias que visam minimizar o *overhead* causado pelo INT, tais como: DIS-TRIBUI/CONCENTRA [Marques and Gasparly 2018], PINT [Ben Basat et al. 2020] e DLINT/PLINT [Papadopoulos et al. 2023, Dimoglis et al. 2024]. Além dessas propostas, também existem abordagens alternativas para a coleta de métricas em redes programáveis. Uma dessas estratégias consiste na utilização de um “fluxo de telemetria exclusivo” para monitorar o estado da rede, técnica que, neste trabalho, é denominada ONT (*Out-of-band Networking Telemetry* [de Almeida et al. 2021]).

Claramente, sob o ponto de vista dos algoritmos de AM, espera-se que, no geral, quanto mais dados de telemetria forem obtidos, melhor será o desempenho na predição da QoS. Entretanto, conforme mencionado, a carga gerada por esta coleta fez com que alternativas fossem desenvolvidas de forma a reduzir a quantidade de dados coletados. Neste caso, há um *tradeoff* entre o desempenho dos algoritmos de AM e a quantidade de telemetria INT coletada, porém, não se tem conhecimento de algum estudo que analisou este aspecto.

Neste contexto, este estudo tem como objetivo avaliar o impacto das estratégias de telemetria (INT e ONT) no desempenho de algoritmos de AM. Em particular, busca-se investigar como a quantidade de dados de telemetria influencia a estimativa de métricas de qualidade de um serviço de vídeo MPEG-DASH [ISO 2014], utilizando algoritmos de AM. A avaliação de desempenho foi realizada em um ambiente virtual, empregando switches BMv2 configurados com as abordagens INT e ONT, codificadas na linguagem P4

(P4 - *Programming Protocol-independent Packet Processors*) [Bosshart et al. 2014]. A escolha desse ambiente deve-se à viabilidade prática, garantindo um cenário controlado e reprodutível. Apesar das limitações, fornece uma base consistente para a análise dos métodos e futuras investigações em hardware real. Os dados de telemetria, representando o estado da rede, juntamente com os dados de qualidade de execução do vídeo, que refletem o estado da aplicação, foram utilizados no treinamento e na avaliação de modelos de AM em um problema de regressão.

Variando a frequência de coleta da telemetria, analisamos o impacto dessa variação na predição dos quadros por segundo (*FPS - Frames Per Second*) processados pela aplicação de vídeo, utilizando o algoritmo Floresta Aleatória (*RF - Random Forest*). O ambiente experimental foi cuidadosamente configurado, incluindo servidores de vídeo DASH, clientes consumidores e geradores de tráfego, com o objetivo de criar diferentes cargas e cenários para a análise.

Os resultados indicam que a utilização da telemetria INT para estimar métricas de um serviço de vídeo DASH pode reduzir o erro de predição em até 7 vezes em comparação com a abordagem ONT. Por outro lado, a abordagem ONT resultou em um aumento de 1.5 vezes em relação ao *throughput*, além de uma redução de 1.82 vezes no tempo de ida e volta (*RTT - Round Trip Time*) dos pacotes na rede. Adicionalmente, observou-se que a abordagem INT gerou uma carga adicional (*overhead*) de aproximadamente 25% na infraestrutura de experimentação.

Neste contexto, destacamos as principais contribuições deste estudo:

1. Realização de uma análise detalhada do impacto das abordagens de telemetria INT e ONT na predição de métricas de QoS em um serviço de vídeo DASH, utilizando modelos de aprendizagem de máquina.
2. Proposição e definição de uma métrica de *overhead* para quantificar a carga adicional gerada pelo uso da telemetria INT em redes programáveis.
3. Disponibilização das implementações das abordagens INT e ONT, desenvolvidas em P4, em um repositório público<sup>1</sup> para fomentar a replicabilidade deste estudo.

O artigo está organizado da seguinte forma: a Seção 2 fornece uma breve descrição dos conceitos fundamentais relacionados à telemetria e ao aprendizado de máquina. Os trabalhos relacionados são discutidos na Seção 3. A Seção 4 descreve a metodologia de avaliação empregada neste estudo, incluindo todos os componentes utilizados na experimentação. Os resultados obtidos são detalhados na Seção 5. Por fim, as conclusões e perspectivas futuras estão apresentadas na Seção 6.

## 2. Conceitos fundamentais

Nesta seção, encontra-se uma breve descrição dos conceitos necessários para uma melhor compreensão deste estudo. Nas próximas subseções estão detalhados os conceitos de telemetria de rede em plano de dados programável. Além disso, ao final da seção encontra-se uma breve apresentação sobre as técnicas e os modelos de aprendizado de máquina utilizados neste estudo.

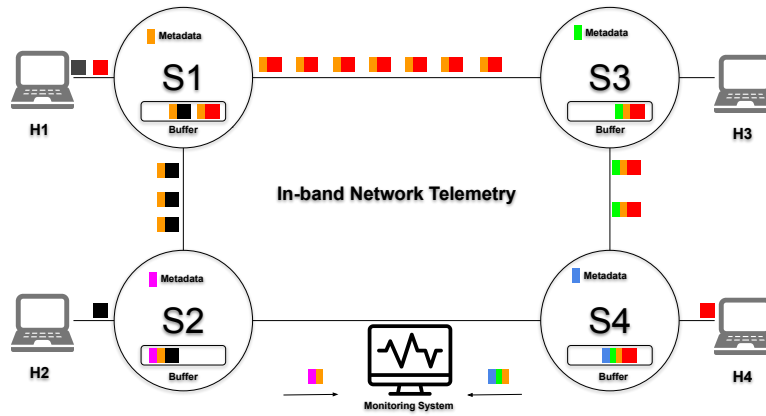
---

<sup>1</sup><https://github.com/ifpb/overhead-telemetry-int>

## 2.1. Telemetria de rede

Avanços em planos de dados programáveis [Bosshart et al. 2014] permitiram que dispositivos de rede informem, de forma autônoma, o estado da rede, eliminando a necessidade de intervenção direta do plano de controle [Arslan and McKeown 2019]. Nesse cenário, os pacotes possuem instruções de telemetria em seus campos de cabeçalho, facilitando a coleta e o registro detalhado de dados da rede. As instruções de telemetria são definidas na especificação INT (*In-band Networking Telemetry*) [P4 2021].

A Figura 1 ilustra a operação do INT dentro de uma rede arbitrária. A rede é composta por quatro hosts ( $H1$ ,  $H2$ ,  $H3$  e  $H4$ ), juntamente com quatro dispositivos de rede equipados com suporte a P4 e INT, identificados como  $S1$ ,  $S2$ ,  $S3$  e  $S4$ . Cada nó da rede possui um conjunto de metadados, definidos na especificação da arquitetura, representados pelos retângulos laranja ( $S1$ ), magenta ( $S2$ ), verde ( $S3$ ) e azul ( $S4$ ).



**Figura 1. Operação de Telemetria de Rede INT. Metadados INT são anexados aos pacotes em cada salto. No ponto de coleta específico, o sistema de monitoramento recebe metadados INT.**

Ainda na Figura 1, há dois fluxos distintos representados: um com pacotes vermelhos e outro com pacotes pretos. O fluxo vermelho deve seguir o caminho de rede  $f1=H1, S1, S3, S4, H4$ , enquanto o fluxo preto deve atravessar o caminho  $f2=H1, S1, S2, H2$ .

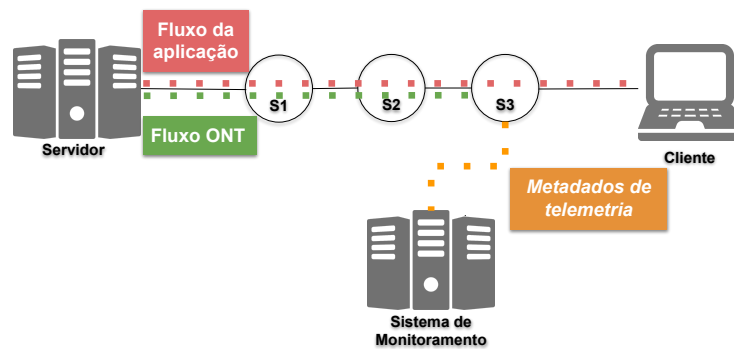
Em cada salto de rede ao longo desses caminhos, o plano de dados dos dispositivos de rede utiliza instruções de telemetria para coletar e incluir metadados nos pacotes à medida que eles atravessam cada nó. Esse processo ocorre iterativamente, iniciando no primeiro nó após a origem e sendo concluído no último nó antes de alcançar o destino. Ao chegar ao nó de destino, os metadados são extraídos do pacote e enviados para o sistema de monitoramento. Posteriormente, o pacote original é encaminhado ao seu destino final.

A principal vantagem desta abordagem consiste na grande quantidade (granularidade fina) de dados de telemetria, uma vez que todos os pacotes que atravessam a rede carregam informações extras de todos os nós. Se do ponto de vista do monitoramento isto é positivo, do desempenho não podemos afirmar com certeza. Acontece que existe uma carga adicional que representa o custo do *overhead* causado pelo INT. Neste trabalho nós definimos uma métrica para representar esta carga, conforme Equação  $O_{rede} = (n * M)/T$ , onde:  $n$  é o número de nós (switches) pelos quais o pacote passa;  $M$

é o tamanho dos metadados INT adicionados por nó (em bits); e  $T$  é o *throughput* original (sem INT) da rede (em bits por segundo).

Além dos modos detalhados na especificação INT, existem abordagens alternativas para a coleta de metadados em redes programáveis. Conforme mencionado anteriormente, uma dessas estratégias consiste na utilização de um “fluxo de telemetria exclusivo” para monitorar o estado da rede, técnica que, neste trabalho é denominada ONT.

No cenário ONT, pacotes de sondagem (“fluxo de telemetria exclusivo”) são empregados para realizar as medições diretamente no plano de dados, eliminando a necessidade de qualquer modificação nos pacotes de dados associados aos serviços que operam dentro da rede, conforme descrito na Figura 2. De forma semelhante ao INT, em cada salto na rede ao longo do caminho, o plano de dados dos dispositivos (S1, S2 e S3) utiliza instruções de telemetria para coletar e incluir metadados apenas nos pacotes do fluxo exclusivo ONT à medida que eles atravessam cada nó. Neste caso o fluxo da aplicação permanece inalterado desde a origem até o destino.



**Figura 2. Operação da telemetria ONT. Um fluxo exclusivo de telemetria realiza a coleta das medições diretamente no plano de dados programável.**

A principal vantagem dessa abordagem reside em sua capacidade de preservar a integridade do tráfego das aplicações, permitindo que ele atravesse a rede sem modificações. Dessa forma, problemas relacionados ao aumento do tamanho dos pacotes, como fragmentação, são mitigados. Por outro lado, o uso de um fluxo de telemetria exclusivo pode introduzir uma sobrecarga adicional ao tráfego total da rede, devido à necessidade de um fluxo de monitoramento dedicado ao ONT para cada serviço em execução dentro da rede.

O uso da telemetria possibilita o rastreamento individual de cada pacote, permitindo a coleta de uma ampla quantidade de informações (metadados) com um nível de granularidade nunca alcançado. De acordo com [Góes de Castro et al. 2019], as informações obtidas no processo de telemetria podem ser úteis para alimentar algoritmos de AM.

## **2.2. Aprendizado de máquina aplicado ao problema de estimativa de QoS**

Conforme discutido na seção anterior, a telemetria baseada em INT é capaz de gerar um grande volume de dados da rede com uma granularidade sem precedentes. Sabendo

que os algoritmos de AM são altamente dependentes de grandes volumes de dados para aprendizado e análise, este estudo possui uma hipótese de que o uso de dados obtidos por meio de telemetria podem ser utilizados como entrada para modelos de AM em problemas de previsão do estado da rede e dos serviços que executam sob ela.

Essa hipótese é suportada pelo fato de que o uso de métricas de telemetria, coletadas em uma taxa nunca antes alcançada e relacionadas aos *buffers* das interfaces dos elementos de rede, pode fornecer uma visão detalhada do nível de congestionamento da rede em um determinado instante. Esse estado é dinâmico, variando de acordo com as condições de carga e as flutuações no tráfego, que, por sua vez, podem influenciar diretamente as métricas de QoS de vídeo DASH executado em uma rede programável.

Neste sentido, este tópico pode ser modelado como uma série temporal, na qual avaliamos como as métricas de telemetria (denominadas por conjunto de dados  $X$ ) se correlacionam no tempo com as métricas serviço (denominadas por conjunto de dados  $Y$ ). Além disso, consideramos um relógio global (*global clock*), que pode ser lido por todos os dispositivos da rede, de modo que todos os relógios estejam sincronizados corretamente. Sendo assim, a evolução das métricas  $X$  e  $Y$  é a série temporal  $\{X_t\}$ ,  $\{Y_t\}$  e  $\{(X_t, Y_t)\}$ , na qual cada instante de tempo  $t$  representa o estado da rede e do serviço DASH. O problema de estimar métricas de serviço  $Y_t$  no tempo  $t$  de um cliente, baseado no conhecimento das métricas de telemetria  $X_t$ , pode ser modelado como  $M : X_t \rightarrow \hat{Y}_t$ , onde  $\hat{Y}_t$  é uma função aproximadora da função  $Y_t$ , para um dado  $X_t$ . Este é um problema de regressão que pode ser resolvido com aprendizado supervisionado [James et al. 2013].

Assim como em outros estudos [Stadler et al. 2017, Calasans 2020, de Almeida et al. 2021], o modelo de AM escolhido para os experimentos neste trabalho foi o *Random Forest* (RF). O RF é um algoritmo que expande o modelo de árvore de decisão (*DT - Decision Tree*) ao combinar várias árvores por meio de uma estratégia de agregação. A estimativa final é calculada a partir da média das estimativas resultantes para cada árvore. Cada árvore é construída usando uma fração dos atributos  $X$  de entrada aleatoriamente, mudando o formato de cada árvore [Stadler et al. 2017]. O objetivo do algoritmo da DT é minimizar a soma dos quadrados residuais, descrito na Equação  $\sum_{j=1}^J \sum_{i \in R_k} (y_i - \hat{y}_{R_j})^2$ , onde  $\hat{y}_{R_j} = \sum_{i \in R_j} \frac{Y_i}{|R_j|}$ .

Nesse caso, o espaço  $X$  de  $n$  características é dividido em  $J$  regiões distintas e não sobrepostas,  $R_1, R_2, \dots, R_j$ . Para cada observação que cair numa dada região  $R_j$ , uma predição é realizada, a qual consiste na média de valores de resposta para as observações de treino em  $R_j$ . Além disso,  $\hat{y}_{R_j}$  é a resposta média para as  $|R_j|$  observações da  $j$ -ésima região [Calasans 2020].

A métrica utilizada para avaliar o desempenho do modelo preditivo no experimento é o Erro Absoluto Médio Normalizado (NMAE). Essa métrica calcula a diferença média entre os valores reais e os valores previstos pelo modelo, normalizada pela média dos valores reais. Em termos percentuais, o NMAE indica o quão distantes, em média, estão as previsões do modelo em relação aos dados reais [Stadler et al. 2017].

Uma característica importante do NMAE é que sua normalização o torna independente da escala das métricas analisadas. Isso permite comparar a precisão das estimativas em diferentes cenários, mesmo quando as métricas possuem magnitudes muito distintas. Essa propriedade é especialmente útil em contextos de alta granularidade, onde os valo-

res podem variar amplamente em termos de magnitude. A fórmula do NMAE é dada por  $NMAE(y, \hat{y}) = \frac{1}{\hat{y}} (\frac{1}{m} \sum_{i=0}^{m-1} |y_i - \hat{y}_i|)$ .

### 3. Trabalhos relacionados

Pesquisas recentes apontam para uma tendência crescente no uso de técnicas de AM para estimar a qualidade de experiência do usuário (QoE) em transmissões de vídeo. Nesta seção, está descrito um resumo de duas linhas de pesquisa: a primeira utiliza exclusivamente métricas da rede como entrada para os algoritmos de AM, enquanto a segunda combina métricas tanto da aplicação quanto da rede. A Tabela 1 contém um breve resumo dos trabalhos avaliados.

Estudos	Objetivo	Ambiente	Entrada	Métricas
Orsolic e Seufert 2021	Avaliar o AM em diferentes datasets	Real	Aplicação e rede	MOS
Jalilvand e Diyanat 2024	Predizer QoE a partir de QoS	Emulado	Rede	MOS
Miranda et al. 2022	Estimar QoE com ICMP probes	Virtual	Aplicação e rede	MOS
<b>Este Artigo</b>	<b>Estimar QoS com INT e ONT</b>	<b>Virtual</b>	<b>Rede (Telemetria)</b>	<b>FPS</b>

**Tabela 1. Comparação dos trabalhos relacionados.**

No estudo de [Orsolic and Seufert 2021], o objetivo foi avaliar o nível de acurácia dos algoritmos de AM na estimativa da QoE correlacionando conjuntos de dados de regiões geográficas distintas e similares. O experimento utilizou dados capturados no dispositivo final, abrangendo métricas de aplicação e rede. O conjunto de dados gerado incluiu informações de 2.000 vídeos do YouTube, transmitidos em diversas sessões, totalizando 24.000 registros.

Os resultados revelaram que, embora a metodologia para consumo do fluxo de vídeo e captura de métricas fosse iguais nas localidades, o uso de um conjunto de dados de uma região distinta como entrada para o algoritmo resultou em baixa acurácia. Por outro lado, ao utilizar o conjunto de dados da mesma região, a estimativa da QoE mostrou boa precisão, tendo como referência o valor de MOS (*Mean Opinion Score*).

Já em [Jalilvand and Diyanat 2024] uma abordagem similar foi adotada, capturando dados de aplicação e rede no dispositivo final para estimar a QoE com o uso de AM. O estudo contudo, apresentou uma metodologia prática para captura e predição automática da QoE, permitindo que operadores de rede implementem soluções dinâmicas e em tempo real, visando alocar recursos de rede de forma eficiente e garantir altos níveis de satisfação do usuário.

Pesquisas menos intrusivas, como a de [Miranda et al. 2022], propõem estimar a QoE utilizando exclusivamente informações da rede, com destaque para o protocolo ICMP. Nesse estudo, a ferramenta fping foi empregada para emitir pacotes ICMP com frequência dinamicamente ajustada, capturando métricas da rede, como RTT, jitter e perda de pacotes. Esses dados foram utilizados como entrada para um algoritmo baseado em árvore de decisão. A abordagem alcançou um erro médio quadrático (RMSE) de 1.05 para um catálogo de 25 vídeos distintos.

Em geral, tais pesquisas capturam dados diretamente nos dispositivos finais da rede e utilizam como referência a métrica de MOS, conforme o modelo ITU-T P1203, para validar a acurácia dos modelos de AM.

No presente estudo, as métricas são obtidas nos dispositivos intermediários, localizados no núcleo da rede, por meio de instruções de telemetria de rede INT [P4 2021], a qual permite a coleta de informações em um nível de granularidade nunca antes alcançado. Neste estudo pretende-se estimar o FPS da aplicação DASH analisando exclusivamente os parâmetros da rede para as abordagens INT e ONT.

## 4. Avaliação de desempenho

Nesta seção estão descritos todos os detalhes da avaliação de desempenho realizada, contendo informações sobre os componentes e a metodologia experimental utilizada.

### 4.1. Descrição dos Componentes

As avaliações experimentais foram conduzidas em um ambiente virtualizado, utilizando um servidor Dell EMC PowerEdge equipado com 2 processadores Intel Xeon® E5-2609 0 @ 2.40GHz (8 vCPUs), 32 GB de RAM e 1 TB de HDD, operando com o sistema Debian GNU/Linux 11 (bullseye). A infraestrutura foi provisionada por meio das ferramentas VirtualBox (6.1.8), Vagrant (2.2.13) e Ansible (2.9.15).

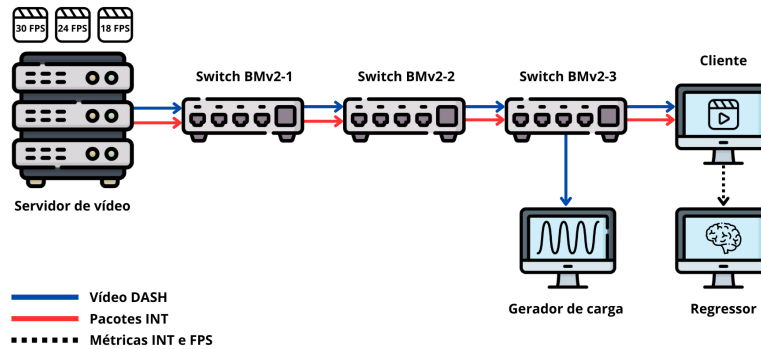


Figura 3. Topologia do experimento.

A topologia experimental foi composta por seis máquinas virtuais interconectadas através de switches BMv2 compatíveis com a linguagem P4, conforme Figura 3. O Servidor de Vídeo foi responsável por disponibilizar o *streaming* de vídeo para os outros dispositivos e por enviar o fluxo de telemetria<sup>2</sup> ao Cliente. O Cliente consumiu o *streaming* de vídeo por meio do VLC (3.0.8), modificado para capturar métricas de qualidade do serviço, armazenando-os no formato apropriado para algoritmos de aprendizado de máquina. O Gerador de Carga WAVE [Almeida et al. 2023] simulou tráfego adicional na rede, utilizando uma função senoidal para emular cargas dinâmicas típicas de datacenters. A função senoidal utilizada pelo Gerador de Carga é definida pela Equação  $f(y) = A \sin(F + \lambda)$ , onde  $A$  representa a amplitude,  $F$  a frequência, e  $\lambda$  a fase. Os Switches BMv2 (1, 2 e 3) foram programados em P4 para adicionar metadados INT aos pacotes que trafegam pela rede.

<sup>2</sup>Para a abordagem ONT.



## 4.2. Descrição dos Experimentos

Foram realizados dois experimentos distintos, um com a abordagem INT e outro com a abordagem ONT. Cada um com duração aproximada de uma hora. Na Figura 4, encontra-se um esboço da execução dos experimentos.

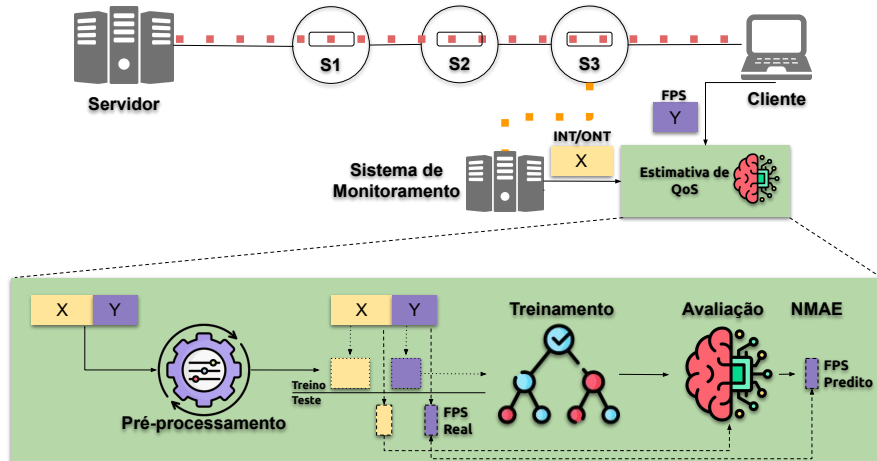


Figura 4. Descrição do experimento.

O servidor de vídeo DASH serviu um *streaming* com três níveis distintos de qualidade (18, 24 e 30 FPS). O cliente VLC consumiu esse *streaming* utilizando um algoritmo de adaptação que realizou transições entre os níveis de qualidade conforme o congestionamento (carga) aplicado na rede. Além disso, no cliente, foram coletadas métricas de qualidade de serviço, representadas pelo FPS (representado pelo conjunto Y na Figura 4).

Simultaneamente, medições foram realizadas nos switches e anexadas aos pacotes da aplicação na abordagem INT ou associadas ao fluxo exclusivo na abordagem ONT (representado pelo conjunto X na Figura 4). Esses pacotes acumularam metadados à medida que atravessavam os switches programáveis da rede, permitindo uma análise detalhada das condições de tráfego em diferentes cenários. Em *background* geradores de carga criaram um congestionamento não-estacionário na infraestrutura de rede.

Após a transmissão do *streaming* de vídeo nas abordagens INT e ONT, os conjuntos de dados X e Y foram direcionados para um módulo de AM para realizar as estimativas de QoS, conforme destacado na Figura 4. No caso da abordagem ONT, os pacotes do fluxo exclusivo de telemetria foram enviados o mais rápido possível, ou seja, sem controle de frequência de envio das sondas. Chamaremos esta abordagem de ONT (0s) nas análises abaixo. A partir deste conjunto de medições, foram derivados três conjuntos adicionais de dados para avaliar o desempenho do modelo de AM em diferentes frequências de coleta de medições. Especificamente, o primeiro conjunto (ONT (1s)) foi gerado com amostras coletadas a cada 1 segundo, o segundo conjunto (ONT (5s)) utilizou um intervalo de 5 segundos, e o terceiro conjunto (ONT (10s)) foi construído com uma frequência de coleta de 10 segundos entre as amostras. Essa estratégia permitiu analisar o impacto de diferentes intervalos de medição no desempenho do modelo de AM. No caso do INT, a frequência da coleta se dá em função da intensidade do tráfego de dados da aplicação DASH, já que as medições são carregadas em todos os pacotes da aplicação.

Inicialmente, os dados passaram por uma etapa de pré-processamento, na qual objetivou-se melhorar a qualidade e a representação dos dados. Nesta etapa foram realizadas os seguintes passos: *i) Tratamento de dados (NaN - Not a Number)*: dados deste tipo podem acarretar em problemas na execução dos métodos. Por este motivo, inicialmente, realizou-se um processo de busca e tratamento desses dados. Neste caso, as amostras com valores do tipo NaN foram removidas; *ii) Remoção de outliers*: a remoção de outliers (valores anômalos ou extremos) tem como principal objetivo melhorar a qualidade dos dados e, consequentemente, o desempenho do modelo; *iii) Remoção de atributos com o mesmo valor*: atributos com valor único não contêm informações que ajudem a distinguir os objetos, portanto são considerados irrelevantes. Por este motivo, atributos com valor único foram removidos; *iv) Normalização dos atributos*: atributos que possuem escalas muito diferentes podem acarretar em problemas nos métodos de aprendizado de máquina. Por este motivo, realizou-se o processo de normalização *z-score* dos atributos com a função `StandardScaler` do pacote Scikit-learn. Neste caso, a média de cada atributo ficou igual a zero e desvio padrão igual a um.

Após a etapa de pré-processamento, os dados foram divididos em partições (80% para treino e 20% para teste) utilizando a função `train_test_split` do pacote Scikit-learn do python. Cada partição foi submetida ao método RF através da função `RandomForestRegressor` do pacote Scikit-learn. Além disso, realizou-se uma busca em grade com validação cruzada e hiperparâmetros, com objetivo de encontrar os melhores modelos. Ao total, foram realizadas 50 (5 x 10) análises de configurações para a abordagem INT e 50 (5 x 10) para cada uma das quatro abordagens ONT avaliadas.

Na etapa de avaliação, o modelo treinado foi avaliado utilizando a partição de teste, correspondente aos 20% do conjunto de dados inicial, que não foram utilizados durante a fase de treinamento. O objetivo dessa etapa foi, com base nos dados da rede nunca antes utilizados, realizar previsões das métricas da aplicação de vídeo ( $Y$ ), destacadas como “FPS Predito” na Figura 4. De posse dos valores de FPS Real ( $Y$ ) e FPS Predito ( $\hat{Y}$ ), foi computado o NMAE, que representa uma métrica de erro, ou seja, um valor menor indica um modelo mais assertivo.

Além dos experimentos mencionados, foram realizados testes adicionais com o objetivo de avaliar o *overhead* introduzido pela telemetria na rede. Especificamente, analisamos o impacto da carga causada pela telemetria por meio de medições de RTT e *throughput* nas abordagens INT e ONT. Utilizando a mesma infraestrutura experimental descrita nesta seção, geramos tráfego TCP sintético com a ferramenta IPERF para medir o *throughput* disponível em ambas as abordagens de telemetria, no caminho entre o servidor de vídeo e o cliente. Simultaneamente, utilizamos a ferramenta PING para medir o tempo de ida e volta (RTT) dos pacotes na rede para cada abordagem (INT e ONT) na mesma direção. Na próxima seção, os resultados obtidos serão detalhadamente apresentados e uma breve discussão sobre as causas, impactos e descobertas será realizada.

## 5. Resultados e discussão

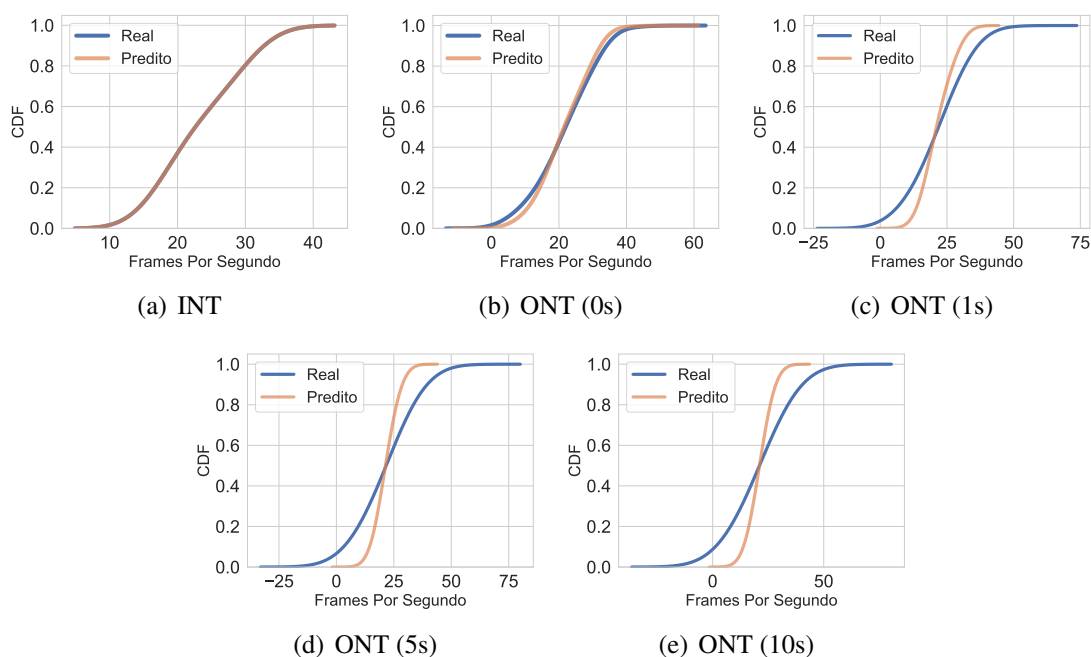
Conforme descrito e detalhado anteriormente, um dos objetivos do experimento é avaliar o desempenho de modelos de AM na estimativas de métricas de QoS de um serviço de vídeo DASH quando a rede de interconexão está sob uma estratégia de telemetria INT ou ONT. Neste trabalho, a métrica de desempenho escolhida foi o NMAE, que é uma

métrica que representa um erro de predição, ou seja, os valores menores indicam um melhor desempenho dos modelos.

Abordagem	INT	ONT (0s)	ONT (1s)	ONT (5s)	ONT (10s)
NMAE	1.28%	9.38%	26.06%	29.82%	30.39%

**Tabela 2. Resultado da predição realizada no conjunto de validação.**

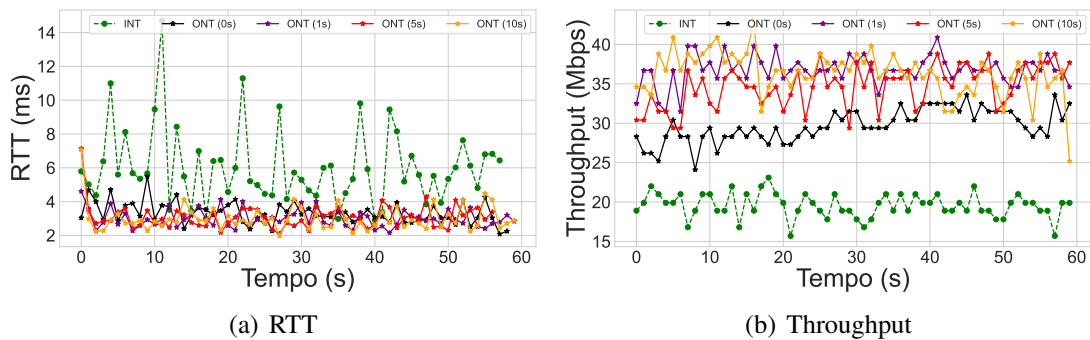
Na Tabela 2, observa-se que o menor erro de predição foi alcançado quando o modelo de AM foi treinado e avaliado utilizando dados da rede obtidos pela abordagem INT, com um erro de 1,28%. Esse valor é 7x menor que o melhor resultado obtido na avaliação da abordagem ONT (0s), que registrou um erro de 9,38%. Além disso, observa-se que o resultado para as abordagens ONT com diferentes frequências de amostras possuem o mesmo comportamento, ou seja, quanto maior a frequência das amostras, menor o erro. Essa análise inicial reforça a hipótese de que os modelos de AM, no geral, apresentam um desempenho superior quando expostos a um volume maior de dados de entrada, permitindo uma melhor capacidade de generalização e predição.



**Figura 5. Predição vs realidade no conjunto de validação.**

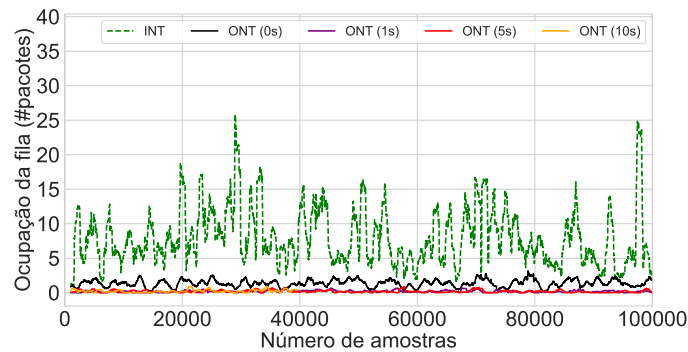
Embora o NMAE seja uma métrica eficaz para resumir numericamente o desempenho dos modelos, também optamos por realizar uma avaliação gráfica dos regressores. Nesse contexto, a Figura 5 ilustra a Função de Distribuição Acumulada (*CDF* - *Cumulative Distribution Function*) dos valores reais de FPS, provenientes do conjunto de validação (não utilizados no treinamento), e dos valores de FPS preditos pelo modelo treinado. Observa-se na Figura 5 que uma menor distância entre as CDFs reflete uma maior similaridade entre os dados, indicando uma menor discrepância entre os valores reais e os valores preditos de FPS. Esses resultados complementam e corroboram os valores de erro NMAE apresentados anteriormente, fortalecendo a análise do desempenho do modelo.

Outro ponto interessante a ser destacado na Figura 5 é o comportamento das curvas. Observa-se claramente que, à medida que o intervalo de frequência entre as amostras aumenta, a distância entre as curvas também cresce, resultando em um erro de predição maior. Esse comportamento evidencia um *tradeoff* entre a assertividade do modelo e a frequência de monitoramento. Embora seja tentador assumir que a melhor abordagem seja sempre a que apresenta o menor erro, é importante considerar, conforme discutido anteriormente, a carga adicional gerada pela abordagem INT. Nesse caso, todos os pacotes do fluxo da aplicação precisam transportar dados de monitoramento provenientes dos nós da rede, o que pode impactar negativamente o desempenho geral do sistema. Neste sentido, os próximos resultados avaliam justamente esta carga.



**Figura 6. Overhead da telemetria em termos de RTT e Throughput.**

A Figura 6 apresenta uma análise comparativa do *overhead* das abordagens INT e ONT (0s, 1s, 5s, 10s), avaliadas em termos de RTT e *throughput*. Na Figura 6(a), observa-se que o RTT na abordagem INT foi maior em comparação a todas abordagens ONT, sendo inclusive 1.82x maior em relação a ONT (0s). Isto indica que os pacotes na abordagem INT levam mais tempo para atravessar os nós da rede. Uma possível explicação para esse resultado é que, na abordagem INT, os switches programáveis realizam operações adicionais ao criar cabeçalhos de rede extras para inserir metadados de telemetria em todos os pacotes da aplicação. Esse processamento adicional não introduz apenas maior atraso, mas também reduz o *throughput* para a abordagem INT em 1.5x em relação a ONT (0s), conforme evidenciado na Figura 6(b).



**Figura 7. Ocupação da fila do switch S1 para as abordagens INT e ONT (0s). A telemetria INT possui uma maior ocupação devido as etapas adicionais.**

Na Figura 7, observa-se a ocupação da fila no primeiro switch da topologia. É

evidente que a ocupação da fila é menor em todas as abordagens ONT em comparação à abordagem INT. Esse resultado está alinhado com as observações apresentadas na Figura 6, pois, na abordagem INT, a criação do cabeçalho extra e a inserção dos metadados de telemetria são realizadas no bloco egresso do switch programável P4, após o *Traffic Manager*, onde as filas estão localizadas. Essa tarefa adicional introduz um atraso no encaminhamento dos pacotes, resultando em um aumento na ocupação média das filas. Isto acarreta um aumento no RTT e pode ser prejudicial para aplicações sensíveis a latência, tais como *streaming* de vídeo.

## 6. Conclusões

Este estudo teve como objetivo avaliar o *tradeoff* entre a frequência de coleta de telemetria em uma rede programável P4 e o desempenho de modelos de AM na estimativa de métricas de QoS de um serviço de vídeo DASH. Os resultados demonstraram que a abordagem de telemetria INT apresentou o menor erro nas estimativas de FPS, reforçando a hipótese de que os modelos de AM alcançam melhor desempenho quando expostos a um maior volume de dados de entrada. Além disso, foi analisada a carga adicional imposta pela atividade de telemetria na infraestrutura de rede. Constatou-se que a abordagem INT gerou um *overhead* de aproximadamente 25% na rede, evidenciando o impacto dessa estratégia de monitoramento.

As conclusões deste estudo indicam direções para trabalhos futuros, incluindo a exploração de amostras sintéticas geradas por Redes Adversárias Generativas (GANs - *Generative Adversarial Networks*) na abordagem ONT, com o objetivo de reduzir o *overhead* imposto pela telemetria INT, mantendo ao mesmo tempo a eficácia das estimativas de QoS. Além disso, outro passo é o avançar o estudo em hardware programável, tais como switches Tofino (Intel) e Tomahawk (Broadcom).

## Referências

- [Almeida et al. 2023] Almeida, L., Silva, J., Lins, R., Jr., P. M., Pasquini, R., and Verdi, F. (2023). Wave - um gerador de cargas múltiplas para experimentação em redes de computadores. In *Anais Estendidos do XLI Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos*, pages 9–16, Porto Alegre, RS, Brasil. SBC.
- [Arslan and McKeown 2019] Arslan, S. and McKeown, N. (2019). Switches know the exact amount of congestion. In *Proceedings of the 2019 Workshop on Buffer Sizing*, BS '19, New York, NY, USA. Association for Computing Machinery.
- [Ben Basat et al. 2020] Ben Basat, R., Ramanathan, S., Li, Y., Antichi, G., Yu, M., and Mitzenmacher, M. (2020). Pint: Probabilistic in-band network telemetry. In *Proceedings of the Annual Conference of the ACM Special Interest Group on Data Communication on the Applications, Technologies, Architectures, and Protocols for Computer Communication*, SIGCOMM '20, page 662–680, New York, NY, USA. Association for Computing Machinery.
- [Bosshart et al. 2014] Bosshart, P., Daly, D., Gibb, G., Izzard, M., McKeown, N., Rexford, J., Schlesinger, C., Talayco, D., Vahdat, A., Varghese, G., and Walker, D. (2014). P4: Programming protocol-independent packet processors. *SIGCOMM Comput. Commun. Rev.*, 44(3):87–95.

- [Calasans 2020] Calasans, M. (2020). *DASH sobre OpenFlow: estimando métricas de QoS a partir da rede*. PhD thesis, Universidade Federal de Uberlândia, Uberlândia - MG - Brazil.
- [de Almeida et al. 2021] de Almeida, L. C., Pasquini, R., and Verdi, F. L. (2021). Using machine learning and in-band network telemetry for service metrics estimation. In *2021 IEEE 10th International Conference on Cloud Networking (CloudNet)*, pages 33–39.
- [Dimoglis et al. 2024] Dimoglis, A., de Almeida, L. C., Papadopoulos, K., Papagianni, C., Papadimitriou, P., and Grosso, P. (2024). Lightweight int on the tofino programmable switch. In *Proceedings of the 30th Annual International Conference on Mobile Computing and Networking*, ACM MobiCom '24, page 2430–2436, New York, NY, USA. Association for Computing Machinery.
- [Góes de Castro et al. 2019] Góes de Castro, A., Rossi, F., Lorenzon, A., Caggiani Luizelli, M., Vogt, F., Hohemberger, R., and Mansilha, R. (2019). Orchestrating in-band data plane telemetry with machine learning. *IEEE Communications Letters*, 23:20.
- [ISO 2014] ISO (2014). Dynamic adaptive streaming over http (dash)-part 1: Media presentation description and segment formats. *ISO/IEC*, pages 23009–1.
- [Jalilvand and Diyanat 2024] Jalilvand, A. and Diyanat, A. (2024). A new approach for predicting the quality of experience in multimedia services using machine learning.
- [James et al. 2013] James, G., Witten, D., Hastie, T., and Tibshirani, R. (2013). *An Introduction to Statistical Learning*. Springer Texts in Statistics. Springer, New York, NY.
- [Marques and Gaspary 2018] Marques, J. A. and Gaspary, L. (2018). Explorando estratégias de orquestração de telemetria em planos de dados programáveis. In *Anais do XXXVI Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos*, pages 1299–1312, Porto Alegre, RS, Brasil. SBC.
- [Miranda et al. 2022] Miranda, G., Macedo, D. F., and Marquez-Barja, J. M. (2022). Estimating video on demand qoe from network qos through icmp probes. *IEEE Transactions on Network and Service Management*, 19(2):1890–1902.
- [Orsolice and Seufert 2021] Orsolice, I. and Seufert, M. (2021). On machine learning based video qoe estimation across different networks. In *2021 16th International Conference on Telecommunications (ConTEL)*, pages 62–69.
- [P4 2021] P4 (2021). In-band network telemetry (int) dataplane specification. Technical report, P4 Consortium.
- [Papadopoulos et al. 2023] Papadopoulos, K., Papadimitriou, P., and Papagianni, C. (2023). Deterministic and probabilistic p4-enabled lightweight in-band network telemetry. *IEEE Trans. on Netw. and Serv. Manag.*, 20(4):4909–4922.
- [Stadler et al. 2017] Stadler, R., Pasquini, R., and Fodor, V. (2017). Learning from network device statistics. *J. Netw. Syst. Manag.*, 25(4):672–698.