Explorando Estratégias de Orquestração de Telemetria em Planos de Dados Programáveis

Jonatas Adilson Marques[1], **Luciano Paschoal Gaspary**[1]

[1] Instituto de Informática
Universidade Federal do Rio Grande do Sul (UFRGS)

{jonatas.marques, paschoal}@inf.ufrgs.br

Abstract. *As networks and their applications evolve, new monitoring aspects and finer granularity become important for correctly identifying and characterizing situations related to, for example, load imbalance and micro-bursts. Data Plane Telemetry emerges in this context as a promising approach to meet this demand, enabling the insertion of information about the network state directly into production packets traversing the network. This kind of telemetry enables unprecedented monitoring accuracy and precision, but may also lead to degradation of traffic if applied naively. In this paper we propose and evaluate strategies for the orchestration of data plane telemetry that explore the effects of concentrating the monitoring load on a small number of packet flows or of distributing it among a larger number of them. The evaluation results demonstrate that it is possible to obtain good quality in terms of coverage, freshness and consistency while considerably reducing production traffic degradation.*

Resumo. *À medida que as redes de computadores e as próprias aplicações evoluem, novos aspectos de monitoração e uma granularidade mais fina tornam-se importantes para identificar e caracterizar corretamente situações ligadas, por exemplo, ao balanceamento inadequado de carga e a micro-bursts. A telemetria no plano de dados surge, nesse contexto, como uma abordagem promissora para satisfazer essa demanda, permitindo a inserção de informações sobre o estado da rede diretamente em pacotes de produção trafegando pela rede. A telemetria habilita a monitoração com acurácia e precisão sem precedentes, mas também pode levar a uma degradação significativa do tráfego caso aplicada ingenuamente. Neste trabalho são propostas e avaliadas estratégias para a orquestração de telemetria no plano de dados que exploram os efeitos de se concentrar a carga de monitoração em um número pequeno de fluxos ou de se distribuí-la entre um número maior deles. Os resultados da avaliação demonstram que é possível obter boa qualidade em termos de cobertura, freshness e consistência com redução considerável de degradação do tráfego de produção.*

1. Introdução

Monitoração é um componente essencial da "disciplina" de gerência e operação de redes. Mais recentemente, com a evolução das redes pela introdução de paradigmas e tecnologias inovadores (*e.g.*, Redes Definidas por Software, virtualização de redes), novos aspectos de monitoração e uma maior granularidade passaram a ser de interesse e, por vezes, imprescindíveis para identificar e caracterizar corretamente situações ligadas a, por

exemplo, falhas, desempenho e segurança. O ferramental de monitoração clássico ou não aborda esses novos aspectos adequadamente ou não é capaz de prover a granularidade demandada.

Na direção de prover maior visibilidade sobre o estado da rede, alguns trabalhos foram desenvolvidos explorando novas possibilidades materializadas por Redes Definidas por Software. NetSight [Handigol et al. 2014], Everflow [Zhu et al. 2015] e Adaptive-Sampling [Cheng and Yu 2017] exploram o espelhamento de pacotes transitando na rede para a criação de *packet histories* contendo o caminho cursado e estimativas do atraso salto-a-salto experimentado por cada pacote. Payless [Chowdhury et al. 2014] explora o ajuste dinâmico da frequência de consultas por estatísticas dos fluxos transitando na rede. DecentralizedMonitoring [Tangari et al. 2017] explora a descentralização do plano de controle de monitoração. Esse trabalho propõe o uso de módulos independentes, cada um realizando tarefas de monitoração de interesse específico para um subconjunto de switches da rede. As três propostas mencionadas apresentam limitações importantes. A primeira limitação é referente ao grande volume de dados de monitoração. Por exemplo, o espelhamento de todos os pacotes transitando na rede por todos os switches gera sobrecarga significativa de transmissão e processamento. A segunda está relacionada ao nível máximo de acurácia e granularidade atingível. Por exemplo, para Payless, mesmo que a frequência de consultas seja muito elevada, a granularidade dos dados não seria suficiente para identificar algumas situações importantes.

Diante da problemática apresentada, um novo mecanismo de monitoração tem sido explorado: a telemetria[1] no plano de dados. Esse modo de telemetria vale-se dos avanços em planos de dados programáveis [Bosshart et al. 2013, Bosshart et al. 2014] para inserir informações sobre os switches da rede (*e.g.*, identificação, ocupação de filas, tempo de processamento) em pacotes do "tráfego de produção". Essas informações são acumuladas em um pacote ao longo de sua rota e subsequentemente extraídas e enviadas para análise. A telemetria no plano de dados provê acurácia e granularidade sem precedentes [Kim et al. 2015, Zhang et al. 2017]. Ao invés de sondas ativas (que são passíveis de experimentar encaminhamento e roteamento inconsistentes com os reais), o próprio tráfego de produção é usado para medir o desempenho da rede. Além disso, o uso do tráfego de produção torna a medição de *estatísticas instantâneas* coerente e útil.

Apesar de a telemetria no plano de dados ser capaz de prover alta qualidade de dados, os *trade-offs* entre qualidade e custos envolvidos em sua execução são, tanto quanto sabemos, ainda pouco explorados. A realização de telemetria envolve a modificação de pacotes de produção transitando na rede. Essa modificação pode contribuir para degradação significativa de desempenho das aplicações dos usuários finais. O impacto dessa modificação é ditado por fatores como: a redução do espaço para carga útil de aplicação, a variação *in-flight* do tamanho dos pacotes, o processamento de pacotes de controle para reportar informações coletadas, e a demanda desses pacotes de controle por largura de banda dos enlaces da rede. Portanto, a mitigação desses fatores é importante quando da realização de telemetria *in-band*.

Neste trabalho propõe-se e avalia-se estratégias algorítmicas para a orquestração

[1]Neste trabalho emprega-se o termo *telemetria* em conformidade com vários trabalhos relacionados [Kim et al. 2015, Thomas and Laupkhov 2016]. Outro termo que poderia ser utilizado para expressar a finalidade desse mecanismo seria *metrologia* [Rocha et al. 2016].

de telemetria no plano de dados que abordam diretamente a mitigação dos fatores que podem contribuir para degradação de desempenho. As estratégias exploram as alternativas de se concentrar a carga de monitoração em um número pequeno de fluxos ou de se distribuí-la uniformemente entre um número maior deles. Introduz-se também, aos algoritmos, um parâmetro K, que permite obter soluções com diferentes níveis de compromisso entre os dois extremos de solução. Por avaliação experimental conclui-se que as estratégias propostas são capazes de gerar soluções de qualidade competitiva (em termos de cobertura, *freshness* e consistência) em relação à estratégia clássica de telemetria com redução considerável de degradação do tráfego de rede e economia de recursos alocados nos switches.

O restante deste documento está organizado como segue. Na Seção 2.1 discute-se os trabalhos relacionados. Na Seção 3 apresenta-se as estratégias de orquestração de telemetria propostas. Na Seção 4 avalia-se a qualidade e os custos das estratégias propostas. Na Seção 5, ressaltada-se as principais conclusões do trabalho, bem como identifica-se vias potenciais para continuidade da pesquisa.

2. Estado da Arte

Nesta seção são discutidos os trabalhos relacionados ao contexto de monitoração em Redes Definidas por Software e Planos de Dados Programáveis.

2.1. Trabalhos Relacionados

NetSight [Handigol et al. 2014] tem como peça fundamental de monitoração o mecanismo de espelhamento de pacotes, que é presente tanto em switches OpenFlow [McKeown et al. 2008] quanto tradicionais. Nesse trabalho, cada switch na rota de um pacote de dados cria uma cópia (*postcard*) deste e a envia a um plano de controle logicamente centralizado. Os múltiplos *postcards* referentes a um único pacote transitando na rede são combinados para formar uma *packet history* que indica a rota percorrida e as modificações sofridas pelo pacote. Uma lacuna deixada em aberto por NetSight é a questão do grande volume de dados produzido por monitoração. Para preencher essa lacuna, em Everflow [Zhu et al. 2015] é proposto o uso do mecanismo de *match-action* para filtrar os pacotes sobre os quais deve-se fazer o espelhamento, reduzindo os custos de banda em detrimento da granularidade e acurácia de monitoração.

PayLess [Chowdhury et al. 2014] e AdaptiveSampling [Cheng and Yu 2017] vão em uma direção um pouco diferente dos trabalhos anteriores. Aproveitam as funcionalidades de switches OpenFlow de (*i*) armazenar estatísticas a respeito das entradas de suas tabelas (que podem representar fluxos) e (*ii*) reportar essas estatísticas mediante requisições enviadas por um controlador. Nesse cenário, a qualidade (*e.g.*, *freshness*) dos dados de monitoração é influenciada pela frequência com que esses dados são consultados. Ambos os trabalhos ajustam dinâmica e independentemente a frequência de consulta de cada fluxo buscando alcançar um bom *trade-off* geral entre qualidade da informação e custo de monitoração. AdaptiveSampling [Cheng and Yu 2017] vai além das propostas anteriores [Handigol et al. 2014, McKeown et al. 2008, Chowdhury et al. 2014] realizando também a amostragem de pacotes nos switches OpenFlow de forma ajustável para cada fluxo. Ambos os trabalhos discutidos neste parágrafo possibilitam reduzir os custos de monitoração, mas obtém informações referentes apenas aos fluxos transitando na rede e não sobre o estado da rede (*e.g.*, ocupação de filas dos switches).

DecentralizedMonitoring [Tangari et al. 2017] aborda o desafio de prover qualidade aceitável e custos baixos através da descentralização da monitoração. O plano de controle da monitoração é composto por múltiplos *módulos de monitoração* espalhados pela rede, sendo cada um capaz de realizar tarefas de monitoração independentemente de uma entidade de controle centralizada e sem a necessidade de se ter uma visão global da rede. Essa abordagem divide logicamente redes em contextos de monitoração. Em cada contexto lógico são realizadas somente as tarefas de monitoração necessárias e, de forma análoga, os dados de monitoração são limitados aos seus próprios contextos. DecentralizedMonitoring reduz os custos de monitoração significativamente em relação aos trabalhos anteriores, mas apresenta as mesmas limitações de granularidade discutidas.

2.2. Rumo à Próxima Geração de Mecanismos de Monitoração

Um paradigma inovador que tem emergido da evolução de Redes Definidas por Software é o de planos de dados programáveis [Bosshart et al. 2013]. Esse novo paradigma introduz a switches a possibilidade de serem configurados em um nível superior de flexibilidade em relação ao OpenFlow [McKeown et al. 2008], por exemplo. A "programabilidade" de planos de dados permite a operadores de rede especificar – através de linguagens como P4 [Bosshart et al. 2014] – como os switches de uma rede devem analisar cabeçalhos (padronizados ou personalizados) e processar seus pacotes. Esse novo nível de flexibilidade desacopla o desenvolvimento e implantação de protocolos inovadores do projeto de hardware de switches [Jeyakumar et al. 2014, Bosshart et al. 2014], permitindo maior agilidade no oferecimento de novos serviços.

A telemetria no plano de dados é um exemplo de serviço inovador que tem sido possibilitado pela programabilidade de planos de dados [Kim et al. 2015, Jeyakumar et al. 2014]. Ela aproveita a oportunidade de definir cabeçalhos e processamento personalizados para inserir informações sobre o estado da rede (*e.g.*, utilização de enlaces, ocupação de *buffers* em switches, atraso salto-a-salto) em pacotes do tráfego de produção de uma infraestrutura [Kim et al. 2015]. Esses dados de telemetria são transparentemente extraídos e reportados por um switch a um controlador, enquanto o pacote com seu conteúdo original é entregue ao destinatário [Thomas and Laupkhov 2016].

Como uma nova abordagem para a monitoração de rede, a telemetria *in-band* permite monitorar redes com alta acurácia e granularidade [Cordeiro et al. 2017]. Apesar desses benefícios, como ela atua sobre o tráfego de produção de rede, é importante materializá-la de forma consciente de potenciais fatores de degradação para minimizar seu impacto sobre os fluxos de pacotes e, ainda assim, manter nível adequado de qualidade de monitoração. A seguir, são enumerados os fatores que podem contribuir para a degradação de desempenho dos fluxos relacionados à realização de telemetria no plano de dados.

(a) **Redução da carga útil para aplicação.** Como os dados de telemetria compartilham o MTU dos pacotes com os dados de aplicação, a carga útil máxima de cada pacote é reduzida à medida que novos itens de telemetria necessitam ser carregadas por um fluxo.

(b) **Competição pela largura de banda dos enlaces da rede.** Os bytes de dados de telemetria são transmitidos *in-band*, potencialmente atrasando a transmissão dos bytes de dados de aplicação e aumentando a latência fim-a-fim experimentada em caminhos com alto nível de utilização.

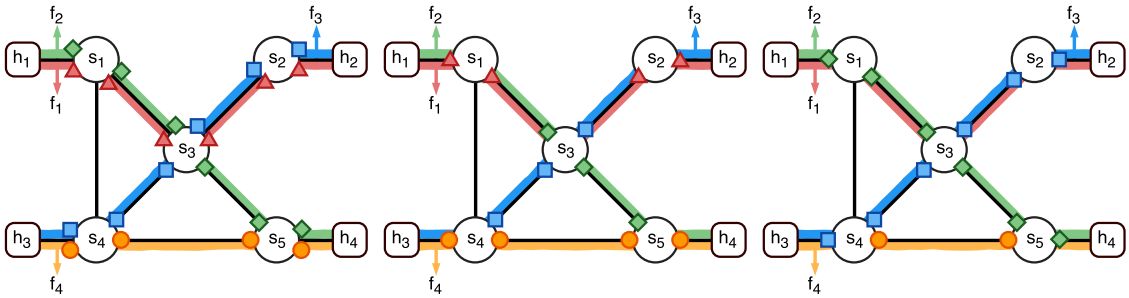
(c) **Variação *in-flight* do tamanho dos pacotes.** À medida que um pacote é roteado pela rede, novos itens de telemetrias são inseridos neste, aumentando seu tamanho. Essa variação no tamanho dos pacotes pode impactar no atraso e *jitter* de transmissão desses, o que pode prejudicar, principalmente, a qualidade de serviço de aplicações não elásticas (*e.g.*, VoIP).

(d) **Processamento de pacotes de controle de telemetria.** Dependendo da implementação da telemetria no plano de dados e da arquitetura do switch programável, a geração e o processamento de pacotes de controle de telemetria podem competir pela largura de banda de encaminhamento do switch, causando o atraso no processamento de pacotes de produção.

Com vistas a explorar os *trade-offs* entre qualidade de medição e custos de implementação nesse novo cenário de monitoração, neste trabalho propõe-se e avalia-se estratégias algorítmicas de orquestração de telemetria em planos de dados que buscam minimizar os custos enquanto mantêm qualidade de informações e visibilidade adequadas sobre a rede. Tanto quanto sabemos, este é o primeiro trabalho a explorar esses aspectos na realização de telemetria no plano de dados.

3. Aprimorando a Telemetria em Planos de Dados Programáveis

Como discutido nas seções anteriores, a telemetria no plano de dados oferece alta acurácia e granularidade de monitoração, mas também apresenta alguns fatores que contribuem para a degradação do tráfego de rede. O nível do impacto causado por esses fatores está intrinsecamente relacionado à atribuição de tarefas de telemetria aos switches que compõem a rede. A Figura 1(a) ilustra a atribuição dessas tarefas tal como sugerida nos trabalhos pioneiros [Jeyakumar et al. 2014, Kim et al. 2015] sobre telemetria no plano de dados. A figura apresenta uma topologia de rede composta por 5 switches e 4 hospedeiros na qual transitam 4 fluxos bidirecionais: $f_1 = (h_1, h_2)$, $f_2 = (h_1, h_4)$, $f_3 = (h_2, h_3)$, e $f_4 = (h_3, h_4)$. Essa estratégia de atribuição, denominada neste trabalho "INT Clássico", configura a rede de forma que cada fluxo colete (e carregue) informações referentes a todas as portas de switches em sua rota. Por exemplo, os pacotes do fluxo f_4 coletam informações das portas do switch s_4 que o conectam a h_3 e s_5 e as do switch s_5 que o conectam a s_4 e h_4, como indicado pelos círculos em laranja na figura.



(a) INT Clássica (b) DISTRIBUI (c) CONCENTRA

Figura 1. Estratégias para orquestração de Telemetria no Plano de Dados.

A estratégia de orquestração "INT Clássico" apresenta algumas deficiências e limitações. Primeiramente, todos os fluxos são expostos aos fatores de degradação mencionados na Seção 2.2. Além disso, toda a carga das tarefas de suporte a telemetria (*e.g.*,

criação de cabeçalhos, geração de relatórios, e remoção de cabeçalhos e dados de telemetria) e os custos de largura de banda para pacotes de controle (*i.e.*, pacotes de relatório direcionados ao controlador de monitoração) são acumulados nos switches de borda da rede. Por fim, as portas dos switches podem ser monitoradas por mais de um fluxo. Tal redundância aumenta com o número de fluxos transitando nas portas e pode não ser necessária [Kim et al. 2015, Zhang et al. 2017], além de aumentar o volume de dados de monitoração gerados.

No presente trabalho propõe-se duas estratégias de orquestração de telemetria (DISTRIBUI e CONCENTRA) que abordam diretamente a mitigação de fatores que contribuem para a degradação de desempenho da rede. As estratégias propostas buscam explorar a influência do número de fluxos usados para telemetria e o número de informações de telemetria carregadas por cada fluxo sobre a qualidade e os custos de monitoração. Essas estratégias atribuem aos fluxos a responsabilidade de monitorar algumas das portas em suas rotas. Cada porta ativa na rede é monitorada por um único fluxo dentre os transitando nesta. Tal procedimento elimina redundâncias e reduz o volume total de dados de monitoração a serem reportados e processados. Além disso, não apenas os switches de borda são responsáveis pelas tarefas de suporte à telemetria. Essa propriedade das soluções das estratégias evita sobrecarregar um subconjunto de switches e enlaces com o processamento e transmissão (respectivamente) de pacotes de controle.

As Figuras 1(b) e (c) ilustram a atribuição de tarefas de telemetria pelas estratégias propostas. A estratégia DISTRIBUI busca distribuir o número de portas monitoradas entre os fluxos, uniformemente, através de um processo de atribuição *round-robin*. Por consequência, DISTRIBUI minimiza o número máximo de informações carregadas por cada fluxo, mas tende a maximizar o número de fluxos que realizam alguma telemetria. Para o exemplo da Figura 1(b), o número máximo de informações carregadas por um fluxo em um de seus sentidos é 2. Por exemplo, f_2 monitora a porta que liga s_3 a s_5 na ida de h_1 para h_4 e as portas que ligam s_5 a s_3 e s_3 a s_1 na volta. CONCENTRA vale-se de uma heurística gulosa para minimizar o número de fluxos usados para telemetria. No exemplo da Figura 1(c), somente três dos quatros fluxos são usados. As soluções de atribuição de tarefas de telemetria de CONCENTRA costumam apresentar alta variabilidade no número de informações carregadas por cada fluxo, com alguns fluxos monitorando quase todas as portas em suas rotas e outros monitorando pouquíssimas portas. Voltando ao exemplo, aos fluxos f_2 e f_3 é atribuída a responsabilidade de monitorar todas as portas de suas rotas (em ambos os sentidos), enquanto nenhuma responsabilidade é atribuída ao fluxo f_1.

A seguir, as Seções 3.1 e 3.2 formalizam as estratégias propostas para orquestração de telemetria no plano de dados. Como essas estratégias representam objetivos de minimização opostos, na Seção 3.3, introduz-se um parâmetro aos seus algoritmos que permite explorar o espaço de soluções entre os extremos.

3.1. DISTRIBUI – Distribuindo Uniformemente a Carga de Telemetria

Dado o objetivo de orquestrar a telemetria, uma possível estratégia é distribuir o máximo possível as tarefas de telemetria. O Algoritmo 1 ilustra o pseudo-código da estratégia DISTRIBUI, que tem tal objetivo. Cada um dos procedimentos do algoritmo é detalhado a seguir.

Algoritmo 1 Pseudo-código da estratégia DISTRIBUI.

Entrada: P: conjunto de portas ativas da rede. F: conjunto de fluxos transitando na rede.

1: $\text{Monitorada\_Por}(p) \leftarrow \text{Null}, \forall p \in P$
2: $\text{Monitorar}(f) \leftarrow \{\}, \forall f \in F$
3: **for** $f \in F$ **do**
4: ORDENAR($p \in$ ROTA(f), ordem decrescente, $|$FLUXOS$(p)|$)
5: **while** existirem portas não monitoradas **do**
6: **for** $f \in F$ **do**
7: **for** $p \in$ ROTA(f) **do**
8: **if** $\text{Monitorada\_Por}(p) = \text{Null}$ **then**
9: $\text{Monitorada\_Por}(p) \leftarrow f$
10: $\text{Monitorar}(p) \leftarrow \text{Monitorar}(f) \cup \{p\}$
11: **break**

Saída: Monitorada_Por, Monitorar

O algoritmo recebe como parâmetros o conjunto P de todas as portas ativas na rede e o conjunto F de todos os fluxos transitando na rede. F é necessário, pois neste trabalho assume-se que os fluxos são conhecidos *a priori*. A análise da relação entre a dinâmica do tráfego e a orquestração de telemetria é prevista como um trabalho futuro. Durante sua execução, o algoritmo mantém duas estruturas de dados – Monitorada_Por e Monitorar – que indicam, respectivamente, qual o fluxo que monitorará cada porta ativa e quais as portas a serem monitoradas por cada fluxo. Essas estruturas de dados são também a saída do algoritmo e são usadas para configurar a rede. As linhas 1 e 2 representam a inicialização dessas estruturas, já que a nenhum fluxo foi atribuída, ainda, a tarefa de monitorar alguma porta. Nas linhas 3 e 4, as portas de cada conjunto ROTA(f) (*i.e.*, conjunto das portas na rota do fluxo f) são ordenadas em ordem decrescente do número de fluxos transitantes (*i.e.*, $|$FLUXOS$(p)|$). As linhas 5-11 contém o laço de repetição central do algoritmo. A cada iteração do laço mais externo, o algoritmo considera cada um dos fluxos $f \in F$ (linha 6) buscando alguma porta em sua rota (respeitando a ordenação) que ainda não esteja monitorada (linhas 7-8). Quando uma porta que atende a tal condição é encontrada, a responsabilidade de monitorar a porta é atribuída ao fluxo (linhas 9-10). Após a atribuição, o algoritmo passa a procurar uma porta ainda não monitorada para o próximo fluxo do conjunto F (linha 11). Esse processo se repete até que um fluxo seja atribuído para monitorar cada uma das portas ativas da rede. A complexidade computacional deste algoritmo (no pior caso) é dada por $\mathcal{O}(n)+\mathcal{O}(m)+\mathcal{O}(mnlog(n))+\mathcal{O}(n^2m) = \mathcal{O}(n^2m)$, sendo n e m o número de portas em P e de fluxos em F, respectivamente.

3.2. CONCENTRA – Minimizando o Número de Fluxos Usados para Telemetria

Nesta subseção, formaliza-se a estratégia de orquestração CONCENTRA , que busca minimizar o número de fluxos usados para telemetria de informações sobre a rede. O Algoritmo 2 apresenta o pseudo-código da estratégia. Esse algoritmo recebe como parâmetros e retorna como solução as mesmas estruturas de dados do Algoritmo 1.

Na linha 5, é criada uma fila de prioridade para os fluxos de F. A fila é ordenada de forma que o primeiro elemento, f_{max}, seja o fluxo com o maior número de portas em sua rota ainda não monitoradas. As linhas 6-13 contém o laço de repetição central do algoritmo. A cada iteração, o algoritmo extrai de FP o fluxo f_{max} do início da fila

Algoritmo 2 Pseudo-código da estratégia CONCENTRA.

Entrada: P: conjunto de portas ativas da rede. F: conjunto de fluxos transitando na rede.

1: $\text{Monitorada\_Por}(p) \leftarrow \text{Null}, \forall p \in P$
2: $\text{Monitora}(f) \leftarrow \{\}, \forall f \in F$
3: **for** $f \in F$ **do**
4: ORDENAR($p \in$ ROTA(f), ordem decrescente, |FLUXOS(p)|)
5: FP ←FILADEPRIORIDADE(F, ordem decrescente, número de portas (ainda) não monitoradas)
6: **while** existirem portas não monitoradas **and** FP não estiver vazia **do**
7: $f_{max} \leftarrow$ FP.EXTRAIRMÁXIMO()
8: **for** $p \in \text{Rota}(f_{max})$ **do**
9: **if** $\text{Monitorada\_Por}(p) = \text{Null}$ **then**
10: $\text{Monitorada\_Por}(p) \leftarrow f_{max}$
11: $\text{Monitora}(f_{max}) \leftarrow \text{Monitora}(f_{max}) \cup \{p\}$
12: **for** $f_p \in$ FLUXOS(p) **do**
13: FP.REDUZIRPRIORIDADE(f_p)

Saída: Monitorada_Por, Monitora

e atribui a este a responsabilidade de monitorar todas as portas em sua rota ainda não monitoradas (linhas 7-11). Para cada uma das novas atribuições fluxo-porta, o algoritmo decrementa a prioridade de todos os outros fluxos atravessando a porta em uma unidade (linhas 12-13), visto que todos esses fluxos possuem agora uma porta a menos ainda não monitorada. Esse ajuste de prioridades causa a reordenação dos fluxos em FP. O algoritmo repete esse procedimento iterativo até que todas as portas sejam monitoradas ou não hajam mais elementos em FP. A complexidade computacional no pior caso deste algoritmo, caso a fila de prioridade seja implementada por um *heap* máximo binário, é dada por $\mathcal{O}(n) + \mathcal{O}(m) + \mathcal{O}(mn \log n) + \mathcal{O}(m \log m) + \mathcal{O}(m) \cdot (\mathcal{O}(\log m) + \mathcal{O}(nm \log m)) = \mathcal{O}(nm^2 \log m)$, sendo n e m o número de portas em P e de fluxos em F (respectivamente); $\mathcal{O}(m \log m)$, a complexidade de construção do *heap* (linha 5); e $\mathcal{O}(\log m)$, a complexidade de ambas as operações de extração do elemento máximo (linha 7) e de atualização da prioridade de um elemento (linha 13) do *heap*.

3.3. Explorando o Espaço de Soluções via Parâmetro K

As duas estratégias de orquestração propostas exploram nos extremos os dois possíveis objetivos (*i.e.*, minimizar o número de fluxos monitorando ou minimizar o número de informações por fluxo) que podem ser traçados para mitigar o efeito dos fatores que contribuem para degradação de desempenho dos fluxos e da rede. No contexto deste trabalho, também é de interesse investigar estratégias que conciliem os dois objetivos e resultem em soluções intermediárias no espaço de soluções. Para atender a esses casos, introduz-se um parâmetro K aos algoritmos propostos. K expressa a proporção máxima desejada entre o número de portas monitoradas por um fluxo e o número total de portas em sua rota, assumindo um valor real no intervalo $[0, 1]$. Por exemplo, para $K = 0.5$, cada fluxo poderia monitorar no máximo metade das portas em sua rota. Já $K = 1$ representa a implementação original dos algoritmos, em que todos os fluxos podem monitorar todas as portas em suas rotas, caso necessário. Note que a introdução do parâmetro K pode impossibilitar que todas as portas ativas da rede sejam monitoradas, mas também permite que operadores de rede estabeleçam um nível máximo de degradação para cada fluxo.

Para a estratégia DISTRIBUI, o Algoritmo 1 é modificado para verificar se tarefas de telemetria ainda podem ser atribuídas para o fluxo f sendo considerado, *i.e.*, se $|\text{Monitorar}(f)| < (K \times |\text{ROTA}(f)|)$. Caso o fluxo f ainda tiver capacidade de monitoração livre, uma nova porta é atribuída normalmente. Caso contrário, o algoritmo segue para a consideração do próximo fluxo de F. O critério de parada do laço de repetição mais externo (linha 5) é modificado de forma que este seja repetido até que todas as portas estejam monitoradas ou não existirem mais fluxos com capacidade livre. A introdução do parâmetro K nesse algoritmo pode limitar a cobertura das soluções (vide Seção 4).

Para a estratégia CONCENTRA, o Algoritmo 2 é modificado para ir atribuindo novas portas ao fluxo f_{max} até que esse não tenha mais capacidade de monitoração livre, *i.e.*, até que $|\text{Monitorar}(f)| = (K \times |\text{ROTA}(f)|)$. Os critérios de parada não sofrem modificação nesse algoritmo. A introdução do parâmetro K limita a eficácia da estratégia em usar o número mínimo de fluxos para carregar dados de telemetria, além de poder limitar a cobertura de suas soluções. A relação entre os níveis de K, a eficácia e a cobertura das estratégias é explorada em detalhe na Seção 4 a seguir.

4. Avaliação

Nesta seção são apresentados e discutidos os resultados obtidos por experimentação para avaliar as propostas. Para o propósito da avaliação, foi desenvolvido um programa P4 para realizar telemetria no plano de dados em linha com a especificação de INT [Kim et al. 2015]. Um switch com este programa instalado pode ser configurado para: (*i*) criar cabeçalhos de telemetria de 4 bytes, que informam o número de itens e o espaço usado por telemetria no pacote; (*ii*) inserir itens de telemetria de 8 bytes, que seria o suficiente, por exemplo, para carregar o ID do switch, o instante em que o pacote foi recebido, e a ocupação do *buffer* de egresso; e (*iii*) extrair as informações armazenadas nos pacotes e reportá-las a um controlador de monitoração. Os switches foram configurados para realizar as tarefas de telemetria de acordo com a solução gerada por cada estratégia de orquestração. A título de comparação, é estudada – além de DISTRIBUI e CONCENTRA – a estratégia do INT Clássico.

Dois grupos complementares de experimentos foram realizados para analisar tanto a qualidade quanto os custos envolvidos na aplicação das estratégias. O primeiro grupo estudou métricas que expressam diretamente a qualidade e os custos das soluções geradas pelas estratégias. Para esse grupo, foi utilizada uma topologia de provedor de Internet (ISP) com 1.000 nós e cerca de 2.800 enlaces gerada através da ferramenta BRITE[2] seguindo os modelos Waxman e Barabasi-Albert conjuntamente. Nessa topologia, há uma média de $11{,}7$ nós entre quaisquer dois hospedeiros e cada nó possui em média $6{,}3$ conexões. O nível de atividade da rede – *i.e.*, o número de fluxos transitando – foi variado entre 200 e 4.200. O valor do parâmetro K (apresentado na Seção 3.3), foi variado entre $0{,}2$ (*i.e.*, cada fluxo pode monitorar até 20% das portas em sua rota); $0{,}3$; $0{,}4$; $0{,}5$; e $1{,}0$ (*i.e.*, cada fluxo pode monitorar todas as portas em sua rota). Para cada combinação de estratégia de orquestração e níveis dos fatores descritos acima foram realizados 32 experimentos explorando diferentes padrões de comunicação entre os hospedeiros da rede. Ao total foram analisados 2.112 cenários distintos. Para as estratégias DISTRIBUI e CON-

[2]http://www.cs.bu.edu/brite/

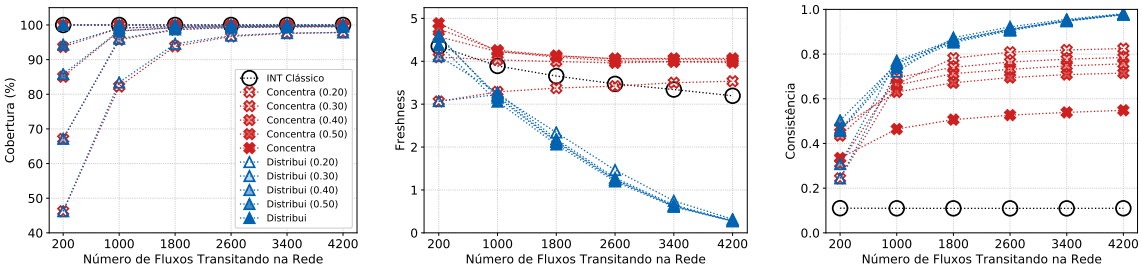
CENTRA foram considerados seis níveis de atividade, cinco valores de K e 32 diferentes combinações de comunicação entre hospedeiros, resultando em $2 \cdot (6 \cdot 5 \cdot 32) = 1.920$ cenários. Para o INT clássico, K é sempre 1, o que resulta em $6 \cdot 1 \cdot 32 = 192$ cenários adicionais. O segundo grupo de experimentos teve por objetivo caracterizar os efeitos das soluções providas pelas estratégias sobre métricas de desempenho de aplicação. Para esses experimentos foi gerada outra topologia (nos moldes da anterior) com 20 nós, 42 enlaces e 80 fluxos. O ambiente de experimentação consistiu em uma instalação *bare-metal* do emulador Mininet[3] com suporte a $P4_{16}$ através do switch de software BMV2[4]. Os fluxos foram gerados por conexões TCP usando iPerf3[5]. Todas as medições foram realizadas em um servidor com quatro processadores AMD Opteron 6276 de 16 núcleos e 64 GB de RAM usando o sistema operacional Ubuntu GNU/Linux Server 16.04 x86_64. A seguir, descreve-se as métricas de avaliação empregadas, bem como apresenta-se e discute-se os resultados obtidos.

4.1. Qualidade das Soluções

Cobertura. A primeira métrica de qualidade avaliada é a cobertura provida pelas soluções. A cobertura é medida como o percentual de portas ativas na rede que tem suas informações coletadas por algum fluxo de pacotes. Quanto maior o valor da cobertura, melhor a solução de atribuição de tarefas de telemetria. A Figura 2(a) apresenta o percentual de cobertura médio (entre 32 casos de teste) das soluções geradas pelas estratégias variando o número de fluxos transitando na rede (eixo x) e o valor do parâmetro K (curvas). Os intervalos de confiança não são exibidos por serem bastante estreitos e para permitir melhor visualização do gráfico. Como a estratégia do INT Clássico atribui a cada fluxo a responsabilidade de monitorar todas as portas em sua rota, a cobertura de suas soluções tem sempre valor 100%. As estratégias propostas, DISTRIBUI e CONCENTRA, apresentam curvas de cobertura de comportamento semelhante para os diferentes níveis de K. Quando $K = 1{,}0$ (símbolos opacos), ambas as estratégias provêm mesma cobertura que o INT Clássico (100%) para qualquer nível de atividade da rede. Quando o valor de K é menor que $1{,}0$, há uma tendência de redução do nível de cobertura. Ainda assim, observa-se que as estratégias apresentam cobertura razoável (superior a 80%) quando o nível de atividade da rede ainda é bastante baixo (1.000 fluxos ou aproximadamente 2 fluxos por cada switch de borda). À medida que a carga da rede aumenta, ambas as estratégias aproximam-se de 100% de cobertura, igualando-se a INT Clássico.

***Freshness*.** A qualidade das soluções pode também ser medida pela "atualidade" ou *freshness* das informações obtidas das portas de switches. Essa métrica é medida como o número médio de saltos pelos quais uma informação é carregada antes de ser encaminhada (em um pacote de controle) ao controlador de telemetria. Quanto mais próximo de zero o valor de *freshness*, melhor a solução. A Figura 2(b) apresenta o valor médio de *freshness* de todas as informações coletadas pelas estratégias. Todas as variações K de DISTRIBUI apresentam comportamento semelhante. Com o aumento do número de fluxos na rede, o valor de *freshness* tende a zero, indicando que a maioria das informações é reportada ao controlador a partir do próprio switch de onde foi obtida. As variações K de CONCENTRA iniciam em patamar próximo ao de DISTRIBUI, mas param de reduzir

[3]http://mininet.org
[4]https://github.com/p4lang/behavioral-model
[5]https://iperf.fr



(a) Cobertura (b) *Freshness* (c) Consistência

Figura 2. Avaliação da qualidade das estratégias.

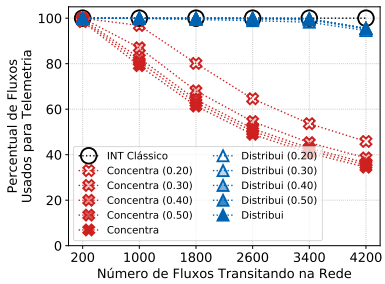
por volta de 4 saltos. A variação $K = 0{,}2$ de CONCENTRA apresenta valor inferior de *freshness*, pois monitora um número inferior de portas (cobertura) que as variações $K >= 0{,}4$. O INT Clássico apresenta valores de *freshness* pouco inferiores a CONCENTRA, mas que chegam a ser 11 vezes maiores que as soluções de DISTRIBUI.

Consistência. A terceira e última métrica de qualidade considerada foi a consistência entre si das informações coletadas. Essa métrica é calculada como a razão entre o número de fluxos monitorando as portas da rota de cada fluxo e o comprimento da rota. Quanto menor a razão, menor o número de fluxos que estão monitorando a rota do fluxo e, consequentemente, maior o número de informações sendo monitoradas conjuntamente, o que melhora a consistência entre as informações obtidas. De acordo com a Figura 2(c), INT Clássico apresenta a melhor consistência, de aproximadamente $0{,}1$, seguido das variações K de CONCENTRA, que apresentam valores piores à medida que o parâmetro K diminui. A estratégia CONCENTRA, para $K = 1{,}0$, converge para aproximadamente $0{,}55$. A estratégia DISTRIBUI apresenta os piores valores de consistência, pois busca distribuir ao máximo a carga de telemetria entre fluxos e é particularmente efetiva nesse aspecto quando o número de fluxos com rotas que compartilham portas é alta.

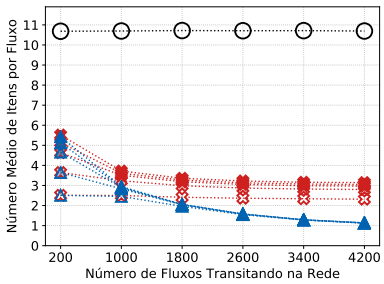
4.2. Custos das Soluções

Percentual de Fluxos Usados Para Telemetria. Quanto menor o percentual de fluxos carregando informações de telemetria, menor o número de pacotes de controle que são gerados periodicamente. Portanto, a redução do número de pacotes de controle ajuda a mitigar os efeitos dos fatores (b) e (d) apresentados na Seção 2.1. A Figura 3(a) ilustra o percentual de fluxos usados para telemetria para diferentes valores de K e níveis de atividade da rede. DISTRIBUI e INT Clássico tendem a atribuir tarefas de telemetria a todos os fluxos transitando na rede. CONCENTRA, por sua vez, reduz o percentual de fluxos que carregam alguma informação à medida que o número de fluxos transitando na rede aumenta, chegando a aproximadamente 35% quando 4.200 fluxos transitam na rede.

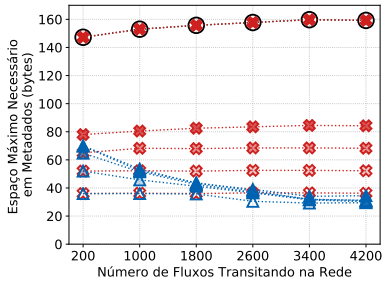
Itens de Telemetria Por Fluxo de Monitoração. Essa métrica de custo está relacionada aos fatores de degradação (a) e (c) apresentados na Seção 2.1. Quanto maior o número médio de itens de telemetria carregados pelos fluxos, maior a redução de carga útil e a variação *in-flight* do tamanho dos pacotes. A estratégia DISTRIBUI apresenta o menor número médio de itens de telemetria por fluxo, chegando a aproximadamente 1 quando o número fluxos transitando na rede é igual a 4.200 (Figura 3(b)). As variações K de CONCENTRA apresentam valores por volta de 2 e 3 itens de telemetria por fluxo, pouco



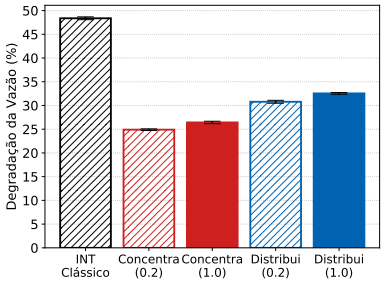
(a) Percentual de Fluxos Usados Para Telemetria



(b) Itens de Telemetria Por Fluxo de Monitoração



(c) Espaço em Memória de Metadados



(d) Degradação da Vazão de Aplicação

Figura 3. Métricas para avaliação dos custos das estratégias.

superiores a DISTRIBUI. Esse resultado mostra que apesar de CONCENTRA priorizar a minimização de fluxos realizando telemetria, essa não aumenta amplamente a carga média de telemetria sobre cada fluxo. A estratégia do INT Clássico faz com que cada fluxo monitore todas as portas em sua rota. Por essa razão, o número médio de itens por fluxo para essa estratégia é uma aproximação do comprimento médio das rotas de todos os fluxos da rede.

Espaço em Memória de Metadados. Para extrair e reportar os itens armazenados nos pacotes dos fluxos, é necessário copiá-los para a memória de metadados dos switches. Essa métrica tem por objetivo identificar o espaço em metadados necessário para que os switches na rede possam copiar todas as informações de telemetria contidas em um pacote atravessando a rede. O espaço de metadados necessário para cada pacote é calculado como o tamanho do cabeçalho de telemetria (4 bytes em nossa implementação) somado do produto entre o número máximo de itens por fluxo e o tamanho de um item de telemetria (8 bytes). O espaço máximo necessário dentre todas as estratégias é de 160 bytes para ambos o INT Clássico e a estratégia CONCENTRA $K = 1$ (Figura 3(c)). DISTRIBUI chega a apresentar valores 4 vezes menores que esse, pois ela minimiza o número de itens por fluxo.

Degradação da Vazão de Aplicação. A última métrica de custo busca avaliar os efeitos das estratégias diretamente no desempenho da rede experimentado pelo fluxos. No experimento de medição foi calculada a vazão agregada de todas as conexões TCP. A Figura 3(d) apresenta a degradação da vazão em pontos percentuais em comparação com o cenário em que não é realizada telemetria no plano de dados. A estratégia do INT Clássico resultou na pior degradação de vazão, cerca de 48%. As variações K de CONCENTRA avaliadas apresentam a menor degradação, aproximadamente metade da sofrida para o INT Clássico. Os intervalos de confiança indicam que há diferença entre a degradação

para $K = 0{,}2$ e $K = 1{,}0$. As variações de DISTRIBUI apresentam degradação por volta de 10 pontos percentuais maiores que as equivalentes em K de CONCENTRA. Essa diferença entre as estratégias é justificada pelo fato de que CONCENTRA reduz significativamente o número de fluxos monitorantes sem apresentar um número médio de itens por fluxo muito superior ao de DISTRIBUI.

5. Conclusão e Trabalhos Futuros

Neste trabalho foram propostas e avaliadas duas estratégias para realizar a atribuição de tarefas de telemetria aos fluxos transitando em uma rede. A primeira estratégia, DISTRIBUI, busca distribuir o mais igualitariamente possível as tarefas de telemetria entre os fluxos. Segundo a avaliação, DISTRIBUI supera as outras estratégias em *freshness* e na mitigação do efeito dos fatores de (a) redução de carga útil para aplicação e (c) variação *in-flight* do tamanho dos pacotes. Esses resultados a tornam particularmente interessante para redes de baixo atraso fim-a-fim ou com percentual significativo de aplicações não elásticas. A segunda estratégia, CONCENTRA, busca minimizar o número de fluxos realizando telemetria. CONCENTRA supera as outras no aspecto de minimização de tráfego de controle de telemetria. Ela dá maior ênfase à mitigação do efeito dos fatores (b) competição pela largura de banda dos enlaces da rede e (d) processamento de pacotes de controle de telemetria, mas sem apresentar grandes perdas nos outros fatores em relação a DISTRIBUI. CONCENTRA é potencialmente útil para redes com alto volume de dados ou onde pode ser necessário restringir a sobrecarga de telemetria a um grupo pequeno de fluxos. Os resultados também mostram que o parâmetro K é eficaz em prover soluções com diferentes níveis de compromisso entre as duas estratégias. De modo geral, a avaliação experimental mostrou que as estratégias propostas obtêm soluções de qualidade competitiva (em termos de cobertura, *freshness* e consistência) em relação à estratégia clássica com redução considerável da degradação do tráfego de rede e economia de recursos alocados nos switches.

Como trabalhos futuros pretende-se investigar outros aspectos da realização de telemetria em planos de dados que podem ser ajustados de forma a prover boa qualidade de monitoração com baixos custos. Exemplos desses aspectos são: monitorar apenas as *top-n* portas mais ativas da rede; aplicar técnicas de amostragem de pacotes para habilitar a monitoração de diferentes grupos de portas por pacotes de um mesmo fluxo; e distribuir o plano de controle de telemetria para diminuir a utilização da rede para escoamento do tráfego de controle.

Agradecimentos

Os autores agradecem aos revisores anônimos os comentários e sugestões. Este trabalho foi parcialmente apoiado pelo processo nº 15/24494-8, Fundação de Amparo à Pesquisa do Estado de São Paulo (FAPESP); pelo Programa de Fomento à Pesquisa da Pró-Reitoria de Pesquisa (PROPESQ) da Universidade Federal do Rio Grande do Sul (UFRGS); e pela Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES).

Referências

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.

Bosshart, P., Gibb, G., Kim, H.-S., Varghese, G., McKeown, N., Izzard, M., Mujica, F., and Horowitz, M. (2013). Forwarding metamorphosis: Fast programmable match-action processing in hardware for sdn. In *Proceedings of the ACM SIGCOMM Conference*, SIGCOMM '13, pages 99–110, New York, NY, USA. ACM.

Cheng, G. and Yu, J. (2017). Adaptive sampling for openflow network measurement methods. In *Proceedings of the International Conference on Future Internet Technologies*, CFI'17, pages 4:1–4:7, New York, NY, USA. ACM.

Chowdhury, S. R., Bari, M. F., Ahmed, R., and Boutaba, R. (2014). Payless: A low cost network monitoring framework for software defined networks. In *Proceedings of the IEEE Network Operations and Management Symposium*, NOMS'14, pages 1–9.

Cordeiro, W. L. d. C., Marques, J. A., and Gaspary, L. P. (2017). Data plane programmability beyond openflow: Opportunities and challenges for network and service operations and management. *Journal of Network and Systems Management*, 25(4):784–818.

Handigol, N., Heller, B., Jeyakumar, V., Mazieres, D., and McKeown, N. (2014). I know what your packet did last hop: Using packet histories to troubleshoot networks. In *Proceedings of the USENIX Conference on Networked Systems Design and Implementation*, NSDI'14, pages 71–85, Berkeley, CA, USA. USENIX Association.

Jeyakumar, V., Alizadeh, M., Geng, Y., Kim, C., and Mazières, D. (2014). Millions of little minions: Using packets for low latency network programming and visibility. *SIGCOMM Comput. Commun. Rev.*, 44(4):3–14.

Kim, C., Sivaraman, A., Katta, N., Bas, A., Dixit, A., and Wobker, L. J. (2015). In-band network telemetry via programmable dataplanes. In *Proceedings of the ACM SIGCOMM Symposium on SDN Research*, SOSR'15.

McKeown, N., Anderson, T., Balakrishnan, H., Parulkar, G., Peterson, L., Rexford, J., Shenker, S., and Turner, J. (2008). Openflow: Enabling innovation in campus networks. *SIGCOMM Comput. Commun. Rev.*, 38(2):69–74.

Rocha, A., Sampaio, L. N., Vieira, A. B., Wehmuth, K., and Ziviani, A. (2016). Revisitando metrologia de redes: Do passado às novas tendências. In *Simpósio Brasileiro de Redes de Computadores e Sistema Distribuídos*, SBRC 2016, pages 151–209.

Tangari, G., Tuncer, D., Charalambides, M., and Pavlou, G. (2017). Decentralized monitoring for large-scale software-defined networks. In *Proceedings of the IFIP/IEEE Symposium on Integrated Network and Service Management (IM)*, pages 289–297.

Thomas, J. and Laupkhov, P. (2016). Tracking packets' paths and latency via int (in-band network telemetry). 3rd P4 Workshop.

Zhang, Q., Liu, V., Zeng, H., and Krishnamurthy, A. (2017). High-resolution measurement of data center microbursts. In *Proceedings of the Internet Measurement Conference*, IMC '17, pages 78–85, New York, NY, USA. ACM.

Zhu, Y., Kang, N., Cao, J., Greenberg, A., Lu, G., Mahajan, R., Maltz, D., Yuan, L., Zhang, M., Zhao, B. Y., and Zheng, H. (2015). Packet-level telemetry in large datacenter networks. In *Proceedings of the ACM SIGCOMM Conference*, SIGCOMM '15, pages 479–491, New York, NY, USA. ACM.