

Uma Estratégia de Encaminhamento de Pacotes Baseada em Aprendizado por Reforço para Redes Orientadas a Conteúdo

Ian V. Bastos¹, Victor C. M. Sousa¹ e Igor M. Moraes¹

¹Laboratório MídiaCom, PGC-TCC
Instituto de Computação - Universidade Federal Fluminense
Niterói, Rio de Janeiro, Brasil

ianvilar@id.uff.br, vcmsousa@id.uff.br , igor@ic.uff.br

Abstract. *This paper proposes a packet forwarding strategy for ICN. Our proposal is based on reinforcement learning techniques and aims at balancing the exploration of new paths and data acquired from previous exploitations. The output interfaces of a node are classified according to the content retrieval time and all interests that share the same prefix with contents previously forwarded are sent through the interface with the lowest mean retrieval time. The path exploration is probabilistic. Each node sends the same interest through the best interface and through another interface chosen at random simultaneously. The goal is to retrieve the content by using the best path found until present moment and at the same time explore copies that are recently stored in the cache of nearest nodes. Simulation results shows that the proposed strategy reduces up to 28% the number of hops traversed by received contents and up to 80% the interest load per node in comparison to other forwarding strategies.*

Resumo. *Esse artigo propõe uma estratégia de encaminhamento de pacotes para redes orientadas a conteúdo. A proposta usa a técnica de aprendizado por reforço, cuja ideia principal é realizar um balanço entre explorar novos caminhos e se aproveitar da informação adquirida durante explorações anteriores. As interfaces são classificadas com base no tempo de recuperação dos conteúdos e todo interesse com o mesmo prefixo para um conteúdo já encaminhado é enviado pela interface com o menor tempo de recuperação médio. A exploração é realizada probabilisticamente, na qual cada nó envia o mesmo interesse para a interface melhor classificada e também para uma outra interface escolhida aleatoriamente. O objetivo é fazer com que o conteúdo seja entregue pelo melhor caminho encontrado até o momento e ao mesmo tempo explorar cópias que possam ter sido armazenadas em caches ainda mais próximos recentemente. Os resultados de simulação mostram que a estratégia proposta reduz o número de saltos cerca de 28% em cada nó e 80% a carga de interesses também por nó em determinados cenários quando comparada às outras estratégias de encaminhamento.*

1. Introdução

Existe um conflito conceitual entre a rápida expansão da distribuição de conteúdos e a arquitetura orientada à conexão da Internet atual. Os usuários atualmente estão mais interessados no conteúdo em si do que na localização ou na identificação de quem o envia.

Aplicações de distribuição de conteúdo já representam mais de 60% do tráfego atual da Internet [Sandvine 2014]. Para aumentar a eficiência da distribuição de conteúdo, muitas arquiteturas baseadas no paradigma das redes orientadas a conteúdo (ROCs) foram propostas [de Brito et al. 2012]. As ROCs introduzem um novo paradigma de comunicação para redes de computadores, no qual a comunicação passa a ser centrada nos conteúdos de interesse, em vez de endereços de destino. Dessa forma, todo conteúdo que faz parte do tráfego nesse novo paradigma é identificado, endereçado e recuperado através de seu nome ao invés de sua localização física. Todos os nós da rede potencialmente armazenam o conteúdo que encaminham, com o objetivo de servir futuras requisições para o mesmo conteúdo, criando uma rede distribuída de *caches*.

Uma arquitetura proposta para as ROCs é a *Content-Centric Networking* (CCN) [Jacobson et al. 2009]. Na CCN, o processo de entrega de conteúdos é iniciado através do envio de requisições pelos usuários interessados em um dado conteúdo. Essas requisições são encaminhadas em direção a uma cópia do conteúdo requisitado, que pode estar armazenada em qualquer nó da rede que previamente tenha encaminhado esse conteúdo. Como um nó pode encaminhar uma solicitação de conteúdo para um ou mais nós vizinhos, a estratégia mais eficiente para recuperar o conteúdo é enviar a solicitação para todos os vizinhos. Caso não existisse uma limitação na largura de banda, a inundação da rede certamente garantiria a entrega da réplica com o menor atraso possível. Entretanto, os enlaces que são utilizados para propagar tais solicitações de conteúdo possuem largura de banda limitada e, em sua grande maioria, heterogênea. Em vista disso, se faz necessário o desenvolvimento de uma estratégia de encaminhamento mais inteligente do que simplesmente realizar uma inundação na rede que poderá levar a um uso ineficiente dos recursos, como a própria capacidade do enlace e espaço em *cache* [Chiocchetti et al. 2012].

Na presença de uma infraestrutura altamente distribuída de *caching* que a CCN introduz, a disponibilidade e a localização dos conteúdos são dinâmicas ao longo do tempo devido às réplicas temporárias que estão distribuídas por toda extensão da rede e a falta de coordenação entre os *caches* [Rosensweig et al. 2013]. Os fatores de distribuição e volatilidade das réplicas são determinados principalmente pela popularidade e políticas de *caching*, que por sua vez estão fortemente relacionados com a estratégia de encaminhamento adotada. Considerando os fatores mencionados, esse artigo propõe uma estratégia de encaminhamento que emprega a técnica de aprendizado por reforço, chamada de MAB (*Multi-Armed Bandit*). Cada vez que uma solicitação de conteúdo precisa ser encaminhada é preciso escolher entre n opções de interfaces de saída diferentes. Ao receber pela primeira vez a solicitação por um conteúdo nunca antes encaminhado, todas as interfaces de saídas são inundadas. Com a solicitação atendida e o conteúdo recuperado, a escolha é feita aleatoriamente entre somente a interface de saída que recuperou o conteúdo mais rapidamente, ou essa mesma interface além de uma outra selecionada ao acaso com distribuição uniforme. Após determinar a interface de saída e receber a réplica do conteúdo desejado, a estratégia proposta atribui uma recompensa numérica a essa interface, nesse caso o tempo de recuperação - tempo definido como o que leva para obter o conteúdo assim que uma requisição por ele é enviada na rede - para obter a réplica, associando a interface utilizada e o conteúdo. O objetivo é fazer com que o conteúdo seja entregue pelos nós cujo conjunto de interfaces de saída resultaram na obtenção do conteúdo com o menor tempo de recuperação com base em explorações anteriores.

Este trabalho também analisa o desempenho do mecanismo proposto através da sua implementação no simulador ndnSIM, no qual cenários de simulação foram gerados para avaliar o impacto do número de consumidores assim como a presença de falhas de enlace na rede. Para tal, métricas como carga na rede, tanto de pacotes de interesse e de dados, quantidade de saltos e o tempo de recuperação dos conteúdos foram avaliados. Os resultados mostram que nossa proposta é capaz de reduzir em cerca de 28% a quantidade saltos necessários para obter um conteúdo em cada nó da rede, assim como 75% a carga de pacotes de dados quando comparada às estratégias *Smart Flooding* e *Best Route*.

O restante do artigo está organizado da seguinte forma. Os aspectos e características básicas do funcionamento da CCN são discutidos na Seção 2. A Seção 3 introduz os conceitos estudados do problema das máquinas caça-níqueis e os contextualiza com as redes CCN e a implementação da estratégia proposta. A Seção 4 discute os parâmetros e cenários de simulação e a Seção 5 os resultados obtidos. Finalmente, na Seção 6, apresenta-se os trabalhos relacionados e a Seção 7 conclui o trabalho realizado e mostra direções para trabalhos futuros.

2. Visão Geral da Arquitetura CCN

A CCN (*Content-Centric Networking* - CCN) [Jacobson et al. 2009] é uma arquitetura de rede projetada com base no paradigma das ROCs. Em vez de endereçar conteúdos através de sua localização, a CCN os referencia pelos seus nomes legíveis. O nome de um conteúdo é composto por um ou mais prefixos de tamanho variável que são opacos para a rede. Os limites de cada prefixo são explicitamente delimitados por uma barra “/”. Em um exemplo, o nome do conteúdo que referencia a página principal do Instituto de Computação da Universidade Federal Fluminense pode ser representado por “/uff.br/ic/index.html”. Conteúdos muito grandes devem ser divididos em fragmentos menores denominados *chunks*. Uma requisição através do prefixo “/uff.br/ic/redes/aula1.avi/4”, por exemplo, se remete ao quarto fragmento do conteúdo aula1.avi da disciplina redes de computadores ministrada no Instituto de Computação da Universidade Federal Fluminense.

Um usuário final na CCN realiza requisições na rede ao enviar pacotes de interesse (*Interest Packets*) para os conteúdos nomeados. Os pacotes são encaminhadas salto a salto em direção a uma cópia permanente do conteúdo requisitado. Para cada interesse recebido, a entidade CCN (roteador ou sistema final) realiza uma consulta por nomes de conteúdos na tabela de encaminhamento (*Forwarding Interest Base* - FIB), que armazena o conjunto de interfaces por qual um determinado conteúdo pode ser obtido. Como vários caminhos são possivelmente conhecidos para um certo prefixo, a camada de estratégia fica responsável por selecionar uma ou mais interfaces de próximo salto entre o conjunto de possibilidades. A camada de estratégia é o módulo responsável por realizar as retransmissões assim como selecionar quais e quantas das várias interfaces de saída serão utilizadas para encaminhar o interesse [Jacobson et al. 2009]. Nós intermediários no caminho escolhido podem dispor de cópias armazenadas em *cache* do conteúdo de interesse em seu armazém de conteúdos (*Content Store* - CS). Nesse caso, pacotes de interesse não necessitam ser encaminhados até o repositório que detém a cópia permanente, e a réplica temporária é enviada imediatamente pelo caminho inverso até o usuário que a requisitou utilizando as informações contidas nas tabelas de interesse pendente (*Pending Interest Table* - PIT). Essa tabela possui como responsabilidade manter uma indexação

entre os prefixos encaminhados pelo nó e todas as interfaces de entrada por onde pacotes de interesse de um mesmo conteúdo foram recebidos. Dessa forma, todo roteador ao encaminhar um pacote de interesse possui a garantia que o pacote de dados correspondente irá passar por ele no caminho de volta até o consumidor que o requisitou.

3. Estratégia de Encaminhamento Proposta

O objetivo da estratégia proposta, chamada de MAB, é minimizar o tempo de recuperação de conteúdos observado pelo usuário final. A ideia principal é descobrir caminhos alternativos que levem a réplicas temporárias de conteúdos sem que nenhuma informação sobre a localização desses conteúdos se encontre disponível *a priori*. A estratégia proposta modela o encaminhamento como o problema das Máquinas Caça-Níqueis e desenvolve um algoritmo em que, além de explorar probabilisticamente suas interfaces de saída, entrega o conteúdo de interesse pela interface que obteve o menor tempo de recuperação.

3.1. O Problema das Máquinas Caça-Níqueis e a Construção da Tabela de Encaminhamento

O problema das Máquinas Caça-Níqueis (*Multi-Armed Bandit*) é um clássico problema encontrado na teoria da probabilidade no qual um apostador, na presença de múltiplas máquinas caça-níqueis, precisa decidir em qual máquina irá apostar. Quando utilizada, cada máquina fornece uma recompensa aleatória a partir de uma distribuição estatística específica da máquina selecionada. O objetivo do apostador é maximizar seu ganho ao concentrar suas apostas nas máquinas que produzem melhores recompensas sobre um determinado período de tempo [Barto 1998].

No contexto da CCN, as máquinas podem ser associadas às interfaces presentes em cada nó da rede. Toda vez que um interesse é recebido por um nó e o conteúdo não se encontra presente em seu *cache*, e não possui uma entrada na PIT indicando um encaminhamento anterior para o mesmo interesse, o nó estará diante do mesmo problema que o apostador ao precisar escolher uma ou mais interfaces para encaminhá-lo.

Na estratégia proposta, que é baseada no algoritmo ε -*greedy*, cada nó constrói sua tabela de encaminhamento ao registrar o tempo de recuperação de cada prefixo de nome quando recebe o pacote de dados correspondente por uma das interfaces de próximo salto. A tabela é composta por entradas que indicam o valor de recompensa médio Q em cada nó i para todos os destinos d , $Q_i(f, d) \forall d \in interfaces(i)$, onde $Q_i(f, d)$ representa o tempo de recuperação médio para obter o conteúdo f após o nó i ter enviado um pacote de interesse via interface que alcança d . A ação de encaminhar realizada por um nó i consiste em selecionar a interface de próximo salto d com a menor recompensa média Q para um conteúdo f , ou seja, a interface que ao longo do tempo produziu o menor tempo de recuperação para que o conteúdo pudesse ser obtido. Descobrir e selecionar a interface de próximo salto d possui complexidade $O(|interfaces(i)|)$. Para cada pacote de dados recebido de volta pela interface d , o nó i atualiza a recompensa Q em sua tabela de acordo com a Equação 1, onde $rtt_{(f,d),k}$ representa o atraso desde o nó i encaminhar o pacote de interesse pelo conteúdo f até o momento em que i recebe o pacote de dados que contém f .

Cada nó atualiza sua tabela de encaminhamento independentemente e, ao serem utilizados em conjunto, conseguem identificar o “melhor” caminho dada as condições

momentâneas da rede para encaminhar e recuperar um determinado conteúdo. Para cada conteúdo $f \in F$, conjunto de todos conteúdos disponibilizados, um nó i mantém um conjunto de valores $Q_i(f, d) \forall d \in interfaces(i)$, que são computados e atualizados toda vez que o nó i recupera f , seja pela “melhor” interface ou por uma interface explorada, como será descrito a seguir.

3.2. O Processo de Encaminhamento e Exploração

A estratégia de encaminhamento proposta supõe que nenhum conhecimento prévio foi adquirido antes de receber o interesse por um conteúdo pela primeira vez. Posto isso, ao receber um pacote de interesse por um conteúdo f que não consta na tabela de encaminhamento, o roteador i irá inundar todas as suas interfaces, exceto a que recebeu o interesse, para inicializar o valor das recompensas $Q_i(f, d)$. O objetivo de tal inundação está em adquirir conhecimento sobre o estado atual da rede em relação a f , garantindo ao mesmo tempo a entrega dos pacotes de dados requisitados.

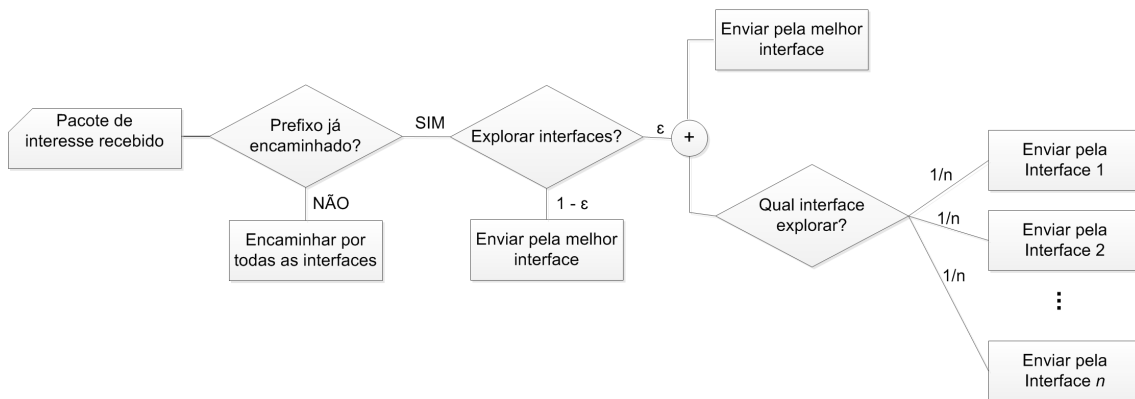


Figura 1. Processo decisório ao encaminhar um pacote de interesse.

Após essa fase inicial, as interfaces são classificadas de acordo com o tempo transcorrido para a recuperação dos primeiros pacotes de dados. A partir desse ponto, toda vez que um pacote de interesse com o prefixo de f for recebido por um nó, ele pesquisará pela interface que é considerada a melhor classificada para recuperar o conteúdo solicitado com tempo constante em sua tabela de encaminhamento através do *hash* computado a partir do prefixo. Em seguida, encaminhará o interesse por um processo aleatório, que decide com uma probabilidade $1 - \varepsilon$ propagar o pacote de interesse somente pela interface que produziu o menor valor $Q_i(f, d)$ no decorrer do tempo. Ou, com probabilidade ε , propagar o interesse para a “melhor” interface e também para outra interface escolhida ao acaso diferente da atual “melhor” seguindo uma distribuição uniforme. Através dessa abordagem, garante-se que f será entregue pelo *cache* mais “próximo” encontrado até o momento enquanto outras cópias que possam ter sido armazenadas por *caches* ainda mais “próximos” em um passado mais recente possam ser encontradas. A opção de possuir somente uma única interface para exploração foi adotada para limitar a sobrecarga de pacotes de interesse trafegando na rede [Chiocchetti et al. 2013].

Da mesma forma, para cada pacote de dados retornado passada a inundação, os nós atualizarão sua tabela de encaminhamento de acordo com a seguinte equação [Barto 1998]:

$$\begin{aligned}
Q_{i,k+1}(f, d) &= Q_{i,k}(f, d) + \alpha[r_{tt(f,d),k} - Q_{i,k}(f, d)] \\
&= \alpha r_{tt(f,d),k} + (1 - \alpha)Q_{i,k}(f, d) \\
&= \alpha r_{tt(f,d),k} + (1 - \alpha)[\alpha r_{tt(f,d),k-1} + (1 - \alpha)Q_{i,k-1}(f, d)] \\
&= \alpha r_{tt(f,d),k} + (1 - \alpha)\alpha r_{tt(f,d),k-1} + (1 - \alpha)^2 Q_{i,k-1}(f, d) \\
&= \alpha r_{tt(f,d),k} + (1 - \alpha)\alpha r_{tt(f,d),k-1} + (1 - \alpha)^2 \alpha r_{tt(f,d),k-2} + \dots \\
&\quad + (1 - \alpha)^{k-1} \alpha r_{tt(f,d),k-1} + (1 - \alpha)^k Q_{i,1}(f, d) \\
&= (1 - \alpha)^k Q_{i,1}(f, d) + \sum_{j=1}^k \alpha (1 - \alpha)^{k-j} r_{tt(f,d),j}.
\end{aligned} \tag{1}$$

Como a localização das réplicas nos *caches* muda com o passar do tempo, incorpora-se uma atualização que leva em conta a não-estacionariedade do problema de encontrar o *cache* que retorna mais rapidamente o conteúdo requisitado. Fixando o valor de α em uma constante, tal que $0 < \alpha \leq 1$, a Equação 1 mostra que, como $1 - \alpha$ é menor que 1, o peso atribuído a $r_{tt(f,d),j}$ diminui à medida que a quantidade de tempos de recuperação intervenientes coletados aumentam. Na verdade, o peso decai exponencialmente de acordo com o expoente $1 - \alpha$. Consequentemente, a recuperação de um pacote de dados no estado atual da rede sempre possuirá um peso maior no cálculo do tempo de recuperação médio, mas sem que o histórico obtido a partir de uma interface deixe de ser levado em conta, já que o estado corrente pode apresentar uma condição particular e temporária.

Outro tratamento realizado via recepção de um pacote de dados é a identificação de uma possível queda de enlace. Cada entrada da tabela de encaminhamento possui uma tupla contendo o prefixo de um conteúdo, a interface utilizada na propagação do interesse, o tempo de recuperação médio e um contador de retransmissões. Assim, sempre que um interesse é enviado, sua entrada correspondente na tabela tem o contador acrescido de uma unidade. Caso esse contador atinja um valor de limiar predefinido T , o valor do tempo de recuperação médio é definido como infinito e, consequentemente, a interface associada a essa entrada deixa de ser utilizada. Como a estratégia de encaminhamento proposta realiza explorações de forma aleatória, eventualmente tal interface que foi classificada como “fora do ar” acaba sendo sondada novamente e, no recebimento de um respectivo pacote de dados, todo processo de aquisição de recompensas associado aos prefixos previamente encaminhados por ela recomeça.

4. Simulação

Para que o desempenho da estratégia proposta pudesse ser avaliado e comparado, o simulador ndnSIM [Afanasyev et al. 2012] foi estendido para acomodar a MAB. A estratégia proposta foi comparada com duas outras estratégias de encaminhamento implementadas por padrão no ndnSIM: a *Best Route* e a *Smart Flooding*. Ambas estratégias apoiam-se em uma codificação de cor para cada interface de saída ao produzir estatísticas das informações coletadas no plano de dados, como tempo de recuperação e sobrecarga da PIT. A primeira encaminha interesses somente para a interface de cor verde (teve o interesse enviado e o dado correspondente retornado) melhor classificada se disponível. Caso

contrário, uma das interfaces de cor amarela (estado da interface desconhecido, ainda não utilizada) é selecionada. A segunda segue o mesmo princípio, entretanto no caso de não possuir uma interface de cor verde, inunda todas as interface de cor amarela. As seguintes métricas são utilizadas para avaliar o desempenho das estratégias de encaminhamento:

- **número médio de saltos:** número médio de saltos que foram necessários para obter o *chunk* de interesse para cada consumidor, incluindo o caminho percorrido pelos pacotes de interesse e pelos pacotes de dados.
- **tempo de recuperação médio:** tempo decorrido, em milissegundos, a partir do primeiro interesse enviado até que o *chunk* requisitado fosse entregue com sucesso.
- **número médio de pacotes de interesse:** média da carga de interesses inseridos na rede.
- **número médio de pacotes de dados:** média da carga de dados inseridos na rede.

A topologia de rede usada na avaliação é uma topologia real obtida através do mapeador de topologias *Rocketfuel* [Spring et al. 2002]. Essa topologia possui um total de 163 nós, dos quais 72 pertencem a borda da rede. A largura de banda dos enlaces, assim como o tamanho da fila das interfaces e o atraso de transmissão são heterogêneos. Para efeito de comparação com as estratégias já adotadas pelo simulador, os experimentos são divididos em função do número de consumidores. Além disso, testes são realizados em diferentes configurações de consumidores para avaliar o comportamento das estratégias de encaminhamento na presença de falha de enlaces.

Em todos os experimentos, existem 3 produtores fixos e, com exceção da avaliação do número de consumidores, 20 consumidores selecionados aleatoriamente entre os 72 nós de borda. Cada consumidor envia 100 interesses por segundo seguindo a distribuição de popularidade MandelBrot-Zipf [Silagadze 1999] com $\alpha = 0,7$. O *cache* dos nós tem capacidade para armazenar até 1000 *chunks* e cada *chunk* tem tamanho igual a 1024 KB. A *Least Recently Used* (LRU) é a política de descarte empregada pelos *caches*. O limiar de retransmissões T para identificação de um possível problema na rede é configurado em 50 tentativas para a estratégia de encaminhamento proposta. Os demais parâmetros do simulador foram mantidos em sua configuração padrão. Todas as simulações tiveram uma duração de 60 s e os resultados são gerados através da média de múltiplas rodadas de simulação, com os intervalos de confiança calculados, representados por barras verticais, para um nível de confiabilidade de 95%.

5. Resultados

5.1. Definição dos Parâmetros da Estratégia de Encaminhamento Proposta

O primeiro passo é investigar o impacto dos parâmetros da estratégia de encaminhamento proposta no seu desempenho. Primeiramente varia-se o parâmetro α para verificar o peso adequado de importância a ser colocado no histórico de recompensas (tempo de recuperação) coletados ao longo do tempo. Nesse experimento, para efeito de simplificação, ajustamos o parâmetro ε em 0,5, ou seja, é concedida a mesma chance de utilizar somente o melhor caminho e de explorar caminhos alternativos. Os resultados foram derivados de 20 rodadas de simulação e representam a média de cada nó consumidor.

Pode-se observar na Figura 2 que o tempo de recuperação médio, diminui conforme aumenta-se o valor de α a partir de 0,2. Esse comportamento indica que atribuir

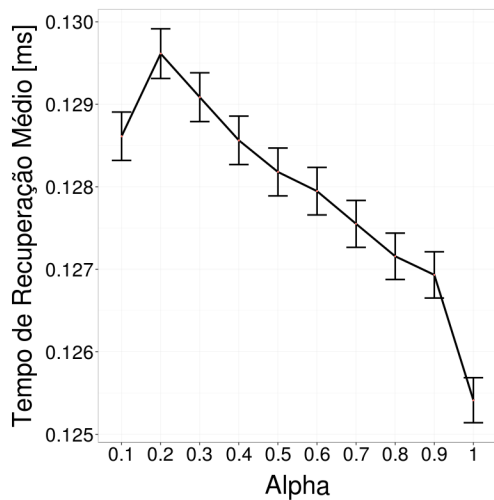


Figura 2. Atraso médio por nó em função de α .

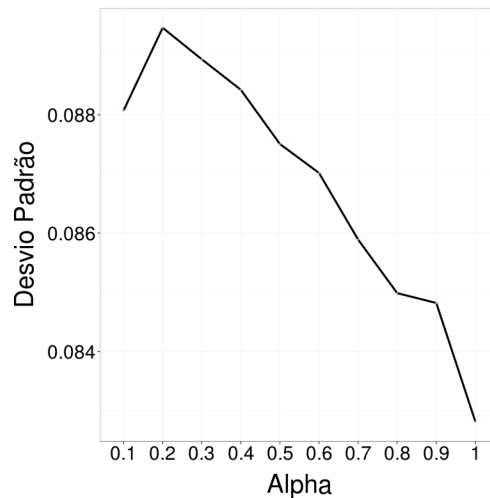


Figura 3. Desvio Padrão.

uma importância maior ao caminho que recuperou um conteúdo com menor atraso mais recentemente proporciona uma adaptação mais rápida em relação à disponibilidade dos conteúdos na rede. A Figura 3 mostra que, além de diminuir o tempo de recuperação médio com maiores valor de α , também mantém-se uma maior estabilidade nos caminhos selecionados. Portanto, assume-se $\alpha = 1$ na comparação da estratégia proposta com as demais estratégias.

Com $\alpha = 1$, investiga-se o melhor valor para ε . A Figura 4 mostra que o tempo de recuperação médio diminui conforme ε aumenta, porém o alto nível de exploração diminui a estabilidade dos caminhos selecionados, ilustrado na Figura 5.

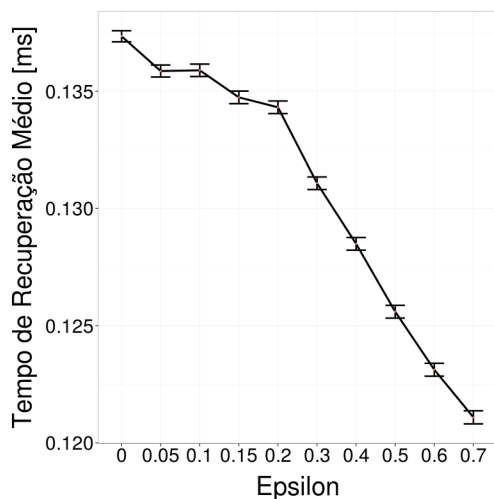


Figura 4. Atraso médio por nó em função de ε .

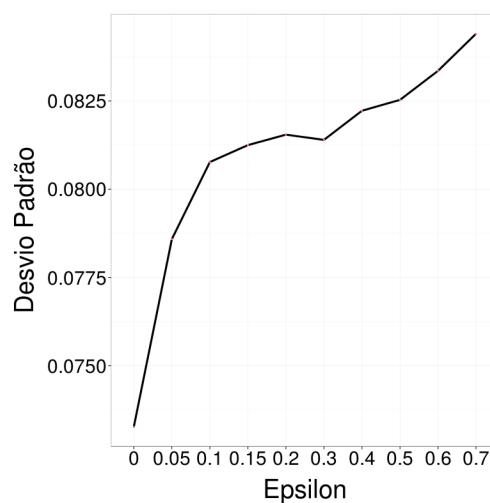


Figura 5. Desvio Padrão.

Como o tempo de recuperação médio e seu desvio padrão divergiram, analisa-se também o comportamento da quantidade de interesses satisfeitos e do número de pacotes

de interesses recebidos em média por nó da rede. A primeira métrica informa os interesses agregados na PIT de cada roteador que foram satisfeitos antes de seu temporizador estourar. A segunda informa a agregação bem sucedida de interesses na PIT. Na Figura 6, pode-se observar que a no intervalo $[0,2, 0,4]$ a quantidade média de interesses satisfeitos aumenta e a carga média de interesses recebidos diminui. Ao se avaliar a razão entre interesses satisfeitos e interesses recebidos nesse intervalo, $\varepsilon = 0,4$ proporciona o maior ganho, aproximadamente 63% em relação à configuração em que nunca novos caminhos são explorados ($\varepsilon = 0$), e uma perda de aproximadamente 14% em relação à configuração com $\varepsilon = 0,7$, mas com uma menor sobrecarga.

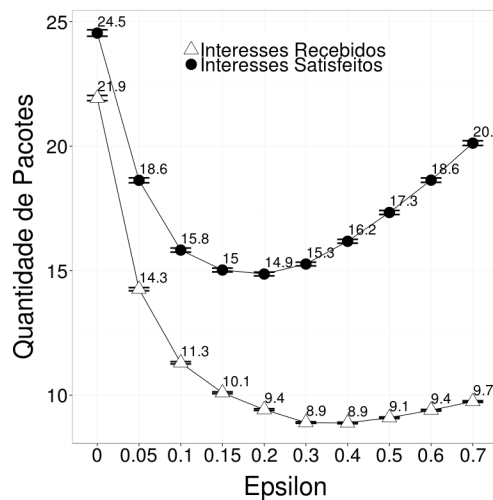


Figura 6. Média de interesses satisfeitos e interesses recebidos por nó.

5.2. Comparação das Estratégias de Encaminhamento

O objetivo dos próximos resultados é avaliar e comparar o desempenho da estratégia proposta MAB com as estratégias *Smart Flooding* (SF) e *Best Route* (BR). Avaliam-se todas as métricas descritas na Seção 4 em função do número de consumidores presentes na rede, que varia de 5 a 50 consumidores. Todos os parâmetros dos mecanismos BR e SF são mantidos do padrão e o nosso mecanismo com $\alpha = 1$ e $\varepsilon = 0,4$.

É possível observar na Figura 7 que a MAB possui um menor tempo de recuperação médio comparada ao SF e uma curva muito semelhante ao BR. Entretanto as Figuras 8 e 9 mostram que a carga inserida na rede, tanto de pacotes de interesse e de pacotes de dados, é menor com a MAB que ambos os mecanismos. Comparada ao SF, a MAB reduz de 44-46% a quantidade de dados recebidos por nó, e com o BR de 55-75%. Já em relação aos pacotes de interesse, a MAB reduz de 21-37% comparado ao SF, desconsiderando seu pico, e cerca de 51-80% quando comparado ao BR. Esse comportamento está associado à exploração esporádica de novos caminhos realizada por cada nó da rede, fazendo com que nós vizinhos aumentem a disponibilidade dos conteúdos e, conseqüentemente, diminua a necessidade dos pacotes serem encaminhados por uma quantidade maior de saltos, o que pode ser verificado na Figura 10.

A segunda avaliação realizada tem como objetivo verificar como a estratégia proposta MAB se comporta na presença de falha de enlaces. Por realizar explorações esporádicas, espera-se que a MAB mantenha aproximadamente as mesmas taxas para cada

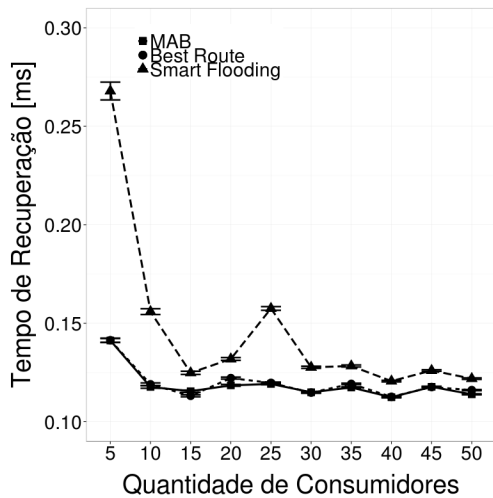


Figura 7. Atraso médio por nó em função da quantidade de consumidores.

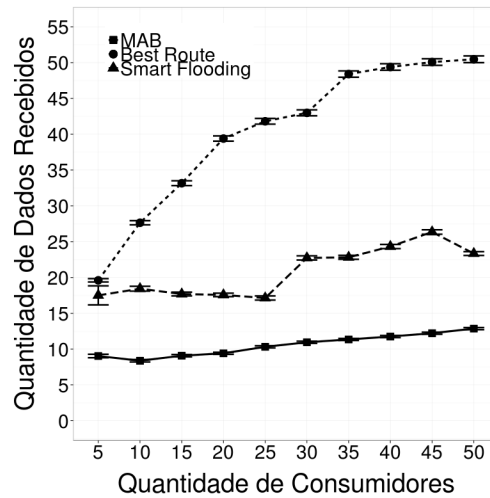


Figura 8. Média da quantidade de pacotes de dados recebidos por nó em função da quantidade de consumidores.

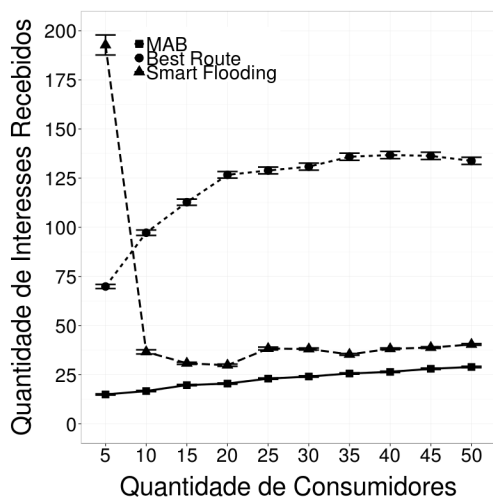


Figura 9. Média da quantidade de pacotes de interesse recebidos por nó em função da quantidade de consumidores.

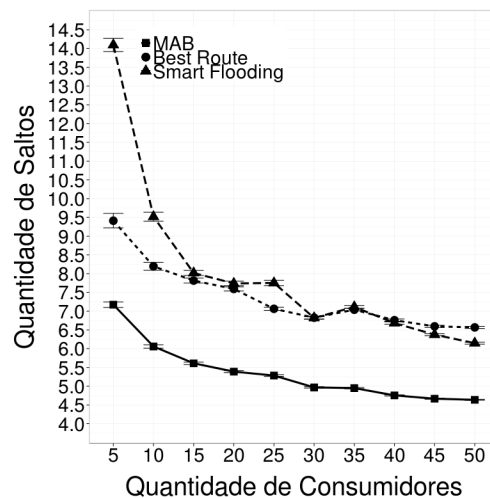


Figura 10. Média do número de saltos por nó em função da quantidade de consumidores.

métrica na presença e na ausência de pequenas falhas na rede em consequência dos caminhos alternativos encontrados durante as explorações. Para realizar a avaliação, são gerados aleatoriamente 20 cenários de falha diferentes contendo cada um 20 consumidores diferentes. Os enlaces foram quebrados em tempos escolhidos de forma aleatória e todos próximos aos produtores que se mantiveram fixos para todos os cenários, forçando a escolha de outros enlaces após as falhas.

A Figura 11 mostra que o tempo de recuperação médio com a MAB se mantém quase que idêntico aos experimentos anteriores, ficou ligeiramente maior que o do SF. Acredita-se que essa diferença seja em razão da forma como os dois mecanismos iden-

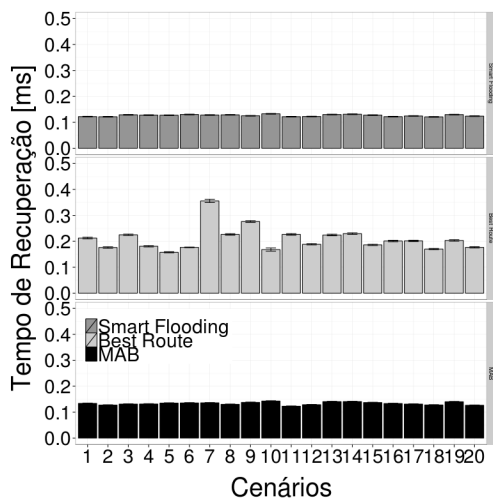


Figura 11. Atraso médio por nó para cada cenário.

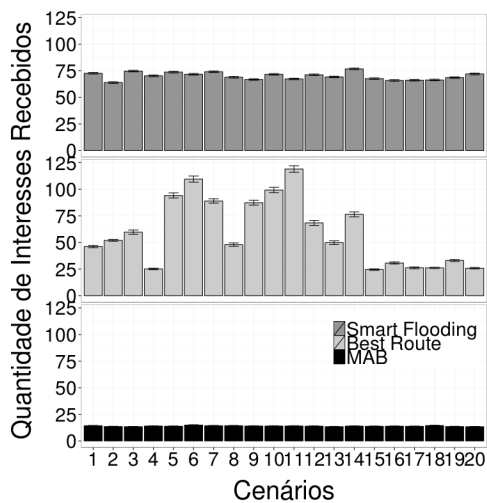


Figura 12. Média da quantidade de pacotes de interesse recebidos por nó para cada cenário.

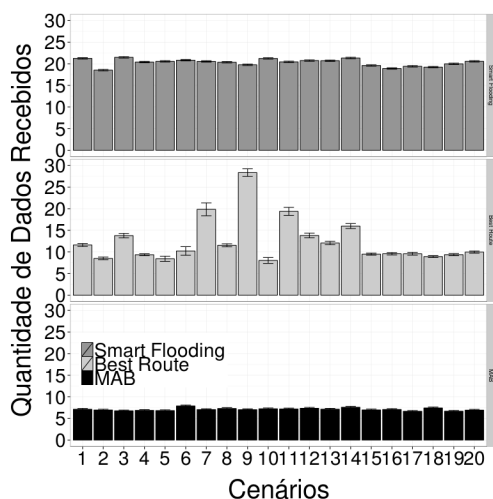


Figura 13. Média da quantidade de pacotes de dados recebidos por nó para cada cenário.

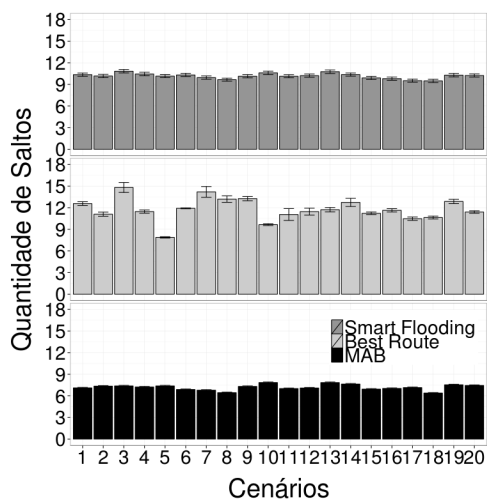


Figura 14. Média do número de saltos por nó para cada cenário.

tificam possíveis problemas na rede. O SF realiza essa identificação pelo uso de pacotes NACK, onde nós acima na hierarquia comunicam a impossibilidade de encaminhar o interesse aos nós inferiores com mais rapidez. A MAB realiza essa identificação ao perceber que depois de T retransmissões nenhum pacote de dados chegou em resposta. No caso dos experimentos, $T = 50$. Ainda assim, a MAB mantém a baixa carga média por nó de pacotes de interesse, Figura 12, e de pacotes de dados, Figura 13 comparada ao SF e ao BR. Além disso, é necessário um menor número de saltos para recuperar os conteúdos, o que mais uma vez enfatiza o aumento da disponibilidade dos conteúdos com o uso das explorações.

6. Trabalhos Relacionados

Pesquisas anteriores sobre encaminhamento em ROCs progrediram em duas direções diferentes. Por um lado protocolos de roteamento operando no plano de controle [Wang et al. 2012b, Wang et al. 2012a, Hoque et al. 2013] possuem a finalidade de disseminar as informações contidas na FIB que endereçam réplicas permanentes. O protocolo de roteamento baseado em nomes OSPFn [Wang et al. 2012a] realiza anúncio dos prefixos de nomes de conteúdos permanentes, que podem ser obtidos com os interesses encaminhados até o produtor que detém as réplicas permanentes.

Na outra direção, trabalhos com o foco em estratégias de encaminhamento no plano dos dados [Chiocchetti et al. 2013, Yi et al. 2012, Rossini e Rossi 2013] possuem como finalidade a recuperação de réplicas voláteis seguindo alguma métrica, como a mais próxima ou a com o menor tempo de recuperação. A principal vantagem dessa abordagem em relação à anterior é a possibilidade de recuperar réplicas temporárias sem a necessidade de ficar sujeito a sobrecarga de sinalização explícita para descobrir sua localização. Dessa forma, realizar uma exploração restrita na rede no intuito de ajustar o encaminhamento conforme o estado momentâneo da rede pode reduzir o tamanho das tabelas de encaminhamento [Chiocchetti et al. 2012] em troca de uma comunicação um pouco mais intensa, mas que poderá beneficiar terceiros com o aumento da disponibilidade dos conteúdo com réplicas armazenadas em nós vizinhos.

Em particular, Chiocchetti et al. 2012 evidenciam o potencial de usar somente o primeiro fragmento de um conteúdo para explorar a rede em busca de réplicas temporárias [Chiocchetti et al. 2012]. Posteriormente, partindo da ideia de explorar esporadicamente a rede, Chiocchetti et al. 2013 utiliza a técnica *Q-learning* de aprendizado por reforço distribuídamente em cada nó da rede com a troca de informações das suas recompensas ao anexá-las nos conteúdos que encaminham [Chiocchetti et al. 2013], mas não apresentam um artifício para verificar que um dado conteúdo está inacessível por alguma das interfaces de saída. A ausência de um mecanismo para identificação de falhas na rede em uma estratégia de encaminhamento que decide a melhor interface de saída com base no tempo de recuperação dos conteúdos pode prejudicar gravemente seu funcionamento. Não receber um pacote de dados implica na não atualização da entrada da tabela de encaminhamento correspondente a interface “defeituosa” e, no caso de ter sido classificada anteriormente como “melhor” interface, o nó continuará enviando interesses por ela sem que os conteúdos sejam recebidos. Por último, Yi et al. 2012 introduz em uma abordagem dinâmica na qual as interfaces são sondadas periodicamente e estatísticas são coletadas para cada uma delas [Yi et al. 2012]. Caso, para um dado conteúdo, uma interface se mostre “melhor” do que a utilizada atualmente, a estratégia de encaminhamento substitui para essa interface a ser explorada, entretanto, por somente utilizarem esse artifício em momentos de uma possível degradação da interface utilizada, deixam de investigar potenciais melhores caminhos.

A estratégia proposta realiza explorações probabilisticamente ao invés de usar a fase de exploração e esperar uma certa convergência como em Chiocchetti et al. 2013, mas aproveita a ideia dos autores de utilizar uma única interface de saída quando não está explorando e duas interfaces de saída quando está. Também realiza detecção de falhas na rede ao usar um limiar de retransmissões, diferentemente de Yi et al. 2012 que usam um pacote de controle.

7. Conclusão

Esse trabalho teve por objetivo avaliar uma estratégia de encaminhamento para a arquitetura CCN baseada na técnica de aprendizado por reforço ϵ - *greedy*. Assim sendo, foi desenvolvida a estratégia denominada MAB (*Multi-Armed Bandits*). Um nó CCN incorporado com a estratégia proposta deve primeiro classificar suas interfaces de acordo com o tempo de recuperação de um dado conteúdo. Em seguida, para cada interesse recebido com um prefixo de nome já presente na tabela de encaminhamento, o nó decide probabilisticamente se envia o interesse somente pela melhor interface classificada ou se envia também o interesse por outra interface escolhida aleatoriamente. O objetivo é aprender outras interface que levem a cópias de conteúdos em nós cada vez mais próximos e, assim, reduzir o tempo de recuperação de conteúdos.

Além do desenvolvimento da estratégia, foi realizada a comparação do seu desempenho com duas outras estratégias de encaminhamento: *Best Route* e *Smart Flooding*. A avaliação considera uma topologia de rede real chamada *Rocketfuel* e duas abordagens para análise das métricas, o número de consumidores e cenários de falha. Os resultados mostram que a estratégia proposta é a que proporciona o menor tempo de recuperação de conteúdos, pois reduz em até 28% o número médio de saltos atravessados pelos conteúdos recuperados com sucesso. Também conclui-se que a estratégia proposta é robusta contra falhas de enlace, uma vez que não aumenta significativamente o tempo de recuperação de conteúdos e a sobrecarga de controle para detectar uma falha.

Algumas avaliações não foram realizadas mas são bastante interessantes de serem feitas. É desejado investigar futuramente a influência do tamanho relativo do *cache* dado um catálogo de conteúdos fixo, assim como o efeito causado pelo tamanho da PIT na taxa de entrega de conteúdos. Além disso, deseja-se estudar como os parâmetros presentes na estratégia proposta influenciam em achar o nó mais “próximo” para conteúdos com diferentes níveis de popularidade. É interessante também analisar os prós e contras da abordagem de usar um limiar de retransmissões com a abordagem de um pacote de controle NACK para detecção de falhas.

Outras considerações importantes a serem estudadas estão relacionadas com a modelagem da simulação. Pretende-se pesquisar modelos que representem uma forma mais consistente com a realidade da Internet, como a taxa de geração de requisições por conteúdos, a distribuição da popularidade dos conteúdos e falhas de enlaces e equipamentos para que a estratégia proposta possa ser comparada com outras estratégias presentes na literatura.

Agradecimentos

Este trabalho é apoiado por CNPq, CAPES, FAPERJ, Proppi/UFF, TBE/ANEEL e CELESC/ANEEL.

Referências

- Afanasyev, A., Moiseenko, I. e Zhang, L. (2012). ndnSIM: NDN simulator for NS-3. Relatório técnico, University of California, Los Angeles.
- Barto, A. G. (1998). *Reinforcement Learning: An Introduction*. MIT press.

- Chiocchetti, R., Perino, D., Carofiglio, G., Rossi, D. e Rossini, G. (2013). INFORM: a dynamic interest forwarding mechanism for information centric networking. Em *Proceedings of the 3rd ACM SIGCOMM workshop on Information-centric networking*, páginas 9–14. ACM.
- Chiocchetti, R., Rossi, D., Rossini, G., Carofiglio, G. e Perino, D. (2012). Exploit the known or explore the unknown?: hamlet-like doubts in ICN. Em *Proceedings of the second edition of the ICN workshop on Information-centric networking*, páginas 7–12. ACM.
- de Brito, G. M., Velloso, P. B. e Moraes, I. M. (2012). Redes orientadas a conteúdo: Um novo paradigma para a Internet. *Minicursos do Simpósio Brasileiro de Redes de Computadores -SBRC*, 2012:211–264.
- Hoque, A., Amin, S. O., Alyyan, A., Zhang, B., Zhang, L. e Wang, L. (2013). NLSR: Named-data link state routing protocol. Em *Proceedings of the 3rd ACM SIGCOMM workshop on Information-centric networking*, páginas 15–20. ACM.
- Jacobson, V., Smetters, D. K., Thornton, J. D., Plass, M. F., Briggs, N. H. e Braynard, R. L. (2009). Networking named content. Em *Proceedings of the 5th international conference on Emerging networking experiments and technologies*, páginas 1–12. ACM.
- Rosensweig, E. J., Menasché, D. S. e Kurose, J. (2013). On the steady-state of cache networks. Em *INFOCOM, 2013 Proceedings IEEE*, páginas 863–871. IEEE.
- Rossini, G. e Rossi, D. (2013). Evaluating CCN multi-path interest forwarding strategies. *Computer Communications*, 36(7):771–778.
- Sandvine (2014). Global Internet phenomena report. Relatório técnico, Sandvine Incorporated ULC.
- Silagadze, Z. K. (1999). Citations and the Zipf-Mandelbrot’s law. Relatório técnico, arXiv preprint physics/9901035.
- Spring, N., Mahajan, R. e Wetherall, D. (2002). Measuring ISP topologies with Rocketfuel. Em *ACM SIGCOMM Computer Communication Review*, volume 32, páginas 133–145. ACM.
- Wang, L., Hoque, A., Yi, C., Alyyan, A. e Zhang, B. (2012a). OSPFN: An OSPF based routing protocol for named data networking. Relatório técnico, University of Memphis and University of Arizona.
- Wang, Y., Lee, K., Venkataraman, B., Shamanna, R. L., Rhee, I. e Yang, S. (2012b). Advertising cached contents in the control plane: Necessity and feasibility. Em *IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, páginas 286–291. IEEE.
- Yi, C., Afanasyev, A., Wang, L., Zhang, B. e Zhang, L. (2012). Adaptive forwarding in named data networking. *ACM SIGCOMM computer communication review*, 42(3):62–67.