

# NDIST - Um Mecanismo de Filtragem de RSSI Eficiente para Algoritmos de *Handoff* em Redes Densas

Helga D. Balbi<sup>1</sup>, Diego Passos<sup>2,3</sup>, Luiz C. S. Magalhães<sup>2</sup>, Célio V. N. Albuquerque<sup>2</sup>

<sup>1</sup>Colegiado de Sistemas de Informação  
Centro Federal de Educação Tecnológica Celso Suckow da Fonseca (CEFET/RJ)  
Nova Friburgo – RJ – Brazil

<sup>2</sup>Laboratório MídiaCom  
Universidade Federal Fluminense (UFF) – Niterói, RJ – Brazil

<sup>3</sup>Instituto Superior de Engenharia de Lisboa (ISEL) – Lisboa – Portugal

helga.balbi@cefet-rj.br; {diego, schara, celio}@midia.com.uff.br

**Abstract.** *In dense wireless networks, handoffs are common. The criteria to trigger them are not defined by the IEEE 802.11 standard, being specific to each manufacturer's implementation. Current handoff implementations often use RSSI (Received Signal Strength Indicator) as a performance metric and commonly result in association instability, a known issue referred to as the "ping-pong effect". To address this effect, we propose NDIST, a RSSI filtering mechanism designed to enhance association stability with minimal delay in handoff triggering. Comparative tests demonstrate that NDIST improves stability without causing a significant increase in delay.*

**Resumo.** *Em redes sem fio densas, handoffs são comuns. Os critérios para acioná-los não são definidos pelo padrão IEEE 802.11, sendo específicos para a implementação de cada fabricante. Implementações atuais de handoff geralmente usam o RSSI (Indicador de Intensidade do Sinal Recebido) como métrica de desempenho e, comumente, causam instabilidade na associação, um problema conhecido como o efeito ping-pong. Para prevenir este efeito, será proposto o NDIST, um mecanismo de filtragem de RSSI, cujo objetivo é aumentar a estabilidade de associação ocasionando baixo atraso no acionamento do handoff. Testes comparativos demonstram que o NDIST melhora a estabilidade sem gerar aumento expressivo do atraso.*

## 1. Introdução

O padrão IEEE 802.11 [IEEE 2012] é amplamente utilizado em ambientes residenciais e corporativos para fornecer conectividade sem fio a dispositivos móveis. A crescente popularidade de dispositivos IoT impulsiona a demanda por maior cobertura e capacidade de rede. Com custos de implantação reduzidos, as redes IEEE 802.11 infraestruturadas densas<sup>1</sup> tornaram-se comuns em ambientes *indoor*. Nessas redes, uma estação cliente (STA) pode estar ao alcance de vários pontos de acesso (APs) que formam um único ESS (*Extended Service Set*), possibilitando a migração entre eles em busca de melhor qualidade de comunicação, um processo conhecido como *handoff* [Vasudevan et al. 2005].

---

<sup>1</sup>Densidade refere-se ao número de dispositivos que compartilham o meio sem fio. Neste contexto, uma rede é considerada densa quando uma STA está no alcance de dois ou mais APs do mesmo ESS.

O padrão IEEE 802.11 não estabelece critérios para acionar um *handoff* ou para escolha de um novo AP, resultando em algoritmos variados entre diferentes dispositivos e plataformas. Como evidenciado em trabalhos anteriores [Balbi et al. 2019, Raghavendra et al. 2007], implementações existentes muitas vezes são ineficientes, resultando em *handoffs* frequentes prejudiciais para os usuários. Idealmente, um *handoff* para um novo AP só deveria ocorrer quando houver ganho potencial em termos de qualidade de comunicação. Contudo, especialmente em ambientes densos, é comum que mesmo STAs estáticas realizem *handoffs* para APs que oferecem qualidade de comunicação semelhante ou até pior do que a atual. Em alguns casos, uma STA realiza *handoffs* sucessivos entre dois ou mais APs em curtos períodos, um comportamento conhecido como *efeito ping-pong* [Mhatre and Papagiannaki 2006], causando interrupções na conectividade, consumo adicional de energia e aumento na sobrecarga de quadros de gerenciamento.

Algoritmos tradicionais de *handoff* usam métricas baseadas no Indicador de Intensidade do Sinal Recebido (RSSI), de modo que APs com RSSI mais alto são preferíveis [Weng and Xie 2024]. O uso do RSSI na decisão de *handoff* pode ser explicado por sua alta disponibilidade, pois o RSSI pode ser obtido de qualquer quadro recebido. Além disso, o RSSI pode refletir bem o movimento de uma STA e a qualidade do canal de comunicação [Vallati et al. 2016, Levis 2006]. No entanto, para proporcionar maior precisão, é necessário usar mecanismos para mitigar a grande variabilidade nas amostras de RSSI causada, principalmente, por efeitos de propagação de sinal em ambientes *indoor*. Caso contrário, a variabilidade do RSSI continuará causando *handoffs* incorretos e o efeito ping-pong. Trabalhos relacionados que consideram a análise do efeito ping-pong, geralmente, propõem o uso de limiares em conjunto com outras técnicas de forma preventiva, como filtragem [Mhatre and Papagiannaki 2006, Montavont et al. 2015]. No entanto, conforme será discutido, o uso de filtros inadequados, que não consideram o comportamento do RSSI, pode aumentar o atraso na ativação de *handoffs* desejados.

Considerando essas questões, o objetivo do presente trabalho é propor o filtro NDIST, que visa mitigar o efeito ping-pong sem ocasionar atrasos expressivos na ativação de *handoffs*. Simulações com base em *traces* de RSSI coletados em redes reais foram realizadas para comparar o NDIST com outros cinco mecanismos comuns de prevenção de efeito ping-pong encontrados na literatura. Os resultados mostram que o NDIST é capaz de oferecer uma melhor estabilidade de associação em cenários com mobilidade, mantendo seu compromisso com a redução do atraso.

O restante do artigo está organizado da seguinte forma: na Seção 2 serão apresentados trabalhos relacionados ao problema do efeito ping-pong; na Seção 3 será apresentado o filtro NDIST; na Seção 4 será apresentada uma avaliação do filtro proposto; e, finalmente, na Seção 5, serão apresentadas conclusões e ideias para trabalhos futuros.

## 2. Trabalhos relacionados

O trabalho de [Mishra et al. 2003] caracteriza o *handoff* em termos do tempo necessário para sua execução e do comportamento de diferentes dispositivos. Ao realizar um *handoff*, primeiro a STA deve se desassociar do AP original e procurar por um novo AP para associação. Nesse ponto, a conexão é interrompida e pacotes são perdidos. Esse intervalo de interrupção é comumente dividido em 3 etapas: a etapa de sondagem, na qual a STA realiza a descoberta de APs vizinhos, um processo conhecido como *scan*; a etapa

de autenticação; e a etapa de reassociação. Em alguns casos, a etapa de *scan* pode ser executada antes da desassociação, reduzindo o intervalo de interrupção. Diferentes placas de rede sem fio seguem uma sequência diferente de mensagens ao realizar o *handoff*, podendo assim apresentar diferentes tempos de interrupção na conexão. Ainda, conforme relatado por [Bangolae et al. 2006], a autenticação de controle de acesso da rede Wi-Fi é outra etapa do *handoff* que pode causar tempos de interrupção mais longos.

Muitos trabalhos buscam amenizar as desvantagens dos *handoffs* empregando técnicas para reduzir o tempo necessário para realizar as várias etapas do procedimento. Alguns deles, como o de [Kim et al. 2006], visam reduzir o tempo necessário para realizar um *scan*. De fato, a redução do tempo necessário para o *handoff* ameniza as desvantagens causadas pelo efeito ping-pong. No entanto, não elimina o problema, pois a conectividade ainda é momentaneamente interrompida, ocasionando quebras de conexões e desperdício de energia para realizar *handoffs* desnecessários.

Trabalhos que abordam o efeito ping-pong comumente propõem o uso de uma ou mais das três técnicas a seguir:

1. *Limiares de qualidade de conexão* [Mhatre and Papagiannaki 2006, Montavont et al. 2015, Hu et al. 2015, Huang et al. 2006, Kim et al. 2012];
2. *Margem de histerese* [Ramani and Savage 2005, Mhatre and Papagiannaki 2006, Montavont et al. 2015, Rangiseti et al. 2014, Hu et al. 2015, Bellavista et al. 2006]; e
3. *Filtragem* [Mhatre and Papagiannaki 2006, Montavont et al. 2015, Rangiseti et al. 2014, Vallati et al. 2016, Pu and Chung 2008, Bellavista et al. 2006].

Quando uma STA utiliza um *limiar de qualidade de conexão*, ela aciona um *handoff* somente se a qualidade de conexão atual, medida por uma métrica específica, estiver abaixo de um determinado limiar. Essa qualidade é comumente calculada utilizando o RSSI dos quadros enviados pelo AP com o qual a STA está atualmente associada (*i.e.*, o AP atual). Mhatre *et al.* [Mhatre and Papagiannaki 2006] concluem que os limiares de qualidade de conexão não têm impacto no efeito ping-pong quando técnicas de filtragem de RSSI são utilizadas em conjunto com esses limiares.

Quando um STA utiliza um algoritmo de *handoff* com uma margem de histerese, o *handoff* é acionado somente se a *diferença* na qualidade de conexão entre os APs atual e candidato estiver acima de um valor pré-determinado. Implementações comerciais comumente utilizam essa técnica [Ramani and Savage 2005]. Embora novas técnicas tenham sido propostas ao longo dos anos, a margem de histerese continua sendo uma das abordagens mais utilizadas hoje em dia. Por exemplo, dispositivos Android costumam utilizar o algoritmo de *handoff* do WPA Supplicant, no qual a margem de histerese varia de acordo com o RSSI amostrado do AP atual [Balbi et al. 2020].

Quando uma STA utiliza técnicas de filtragem, as oscilações rápidas na série temporal do RSSI podem ser removidas, evitando o efeito ping-pong. O trabalho de [Mhatre and Papagiannaki 2006], por exemplo, propõem o uso da Média Móvel Exponencialmente Ponderada (EWMA), com  $\alpha = 0.9$  — o peso atribuído às amostras mais antigas. O uso da EWMA proporciona uma maior estabilidade em relação à margem de histerese. No entanto, o tempo de resposta da EWMA pode aumentar o atraso na ativação

dos *handoffs* desejados, principalmente para valores elevados de  $\alpha$ , que se traduzem em frequências de corte mais baixas do filtro.

Por ser um filtro passa-baixas, a EWMA remove frequências altas do sinal, fazendo com que as mudanças no sinal de saída ocorram mais lentamente em comparação com o sinal original que entra no filtro. Para que o sinal de saída mude mais rapidamente e convirja rapidamente em direção ao sinal de entrada, frequências mais altas devem passar, ou seja, o filtro deve ser ajustado com uma frequência de corte mais alta, o que se traduz em valores menores de  $\alpha$ . Por outro lado, para obter um sinal de saída mais estável, ou seja, com menos ruído, devem ser usadas frequências de corte mais baixas, o que se traduz em valores mais altos de  $\alpha$ . Esse atraso de convergência entre o sinal de saída em direção ao sinal de entrada é chamado de *tempo de resposta do filtro*. Ao filtrar uma série temporal de RSSI, é desejável que o filtro tenha a maior capacidade possível de filtrar o ruído, de modo que o efeito ping-pong seja reduzido, e, ao mesmo tempo, apresente um tempo de resposta baixo, para que não haja atraso na ativação de um *handoff* desejado.

Embora a filtragem seja um tópico bem estudado, com muitos filtros clássicos disponíveis na literatura, para empregar um filtro adequado é importante considerar o comportamento específico da série temporal do RSSI. Como essa série temporal pode apresentar um comportamento peculiar, filtros comumente aplicados em outras áreas podem não ser adequados. Intuitivamente, pode-se acreditar que a série temporal do RSSI tem uma distribuição normal de acordo com os modelos de propagação em larga escala [Sklar 1997, Smailagic et al. 2000]. Dessa forma, uma ideia inicial é considerar essa característica na criação de um mecanismo de filtragem, como ocorre na proposta do presente trabalho que será apresentada na seção a seguir.

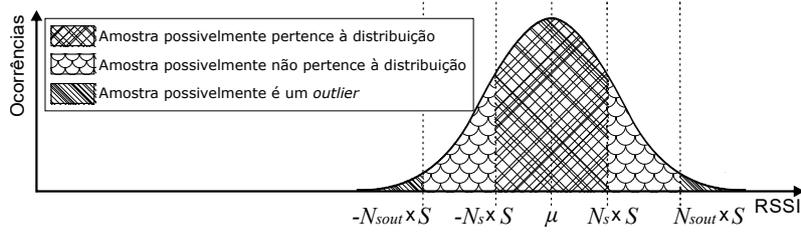
### 3. NDIST

Este estudo introduz o NDIST, um algoritmo de filtragem que pressupõe que a série temporal de RSSI de um AP específico segue uma distribuição normal. O objetivo do NDIST é verificar se uma nova amostra de RSSI pertence ou não à distribuição atual, estimada pela média ( $\mu$ ) e desvio padrão ( $S$ ) de amostras previamente guardadas em uma janela deslizante de tamanho  $W_s$ . Para a tomada de decisões de *handoff*, a média  $\mu$  dos valores contidos na janela deverá ser utilizada.

Após a execução do *scan*, quando uma nova amostra de RSSI é recebida de um Ponto de Acesso (AP) pela Estação (STA), um teste é realizado para determinar se ela pertence ou não à distribuição atual. Após o teste, se o resultado for positivo, a amostra será armazenada na janela deslizante do AP. Se o resultado for negativo, primeiro o algoritmo verifica se a amostra em questão é ou não um *outlier*, ou seja, se seu valor é atípico. Caso isso seja verdade, para assegurar a confiabilidade das análises, a amostra será filtrada. Caso a amostra esteja fora da distribuição e não seja um *outlier*, pode-se inferir que a distribuição está variando. Portanto, a amostra deverá ser armazenada na janela deslizante e a média  $\mu$  e o desvio  $S$  deverão ser atualizados. Desta forma, a distribuição é ajustada à medida que novas amostras chegam.

Para testar se a amostra de RSSI pertence ou não à distribuição atual, ou se ela é um *outlier*, o NDIST utiliza dois multiplicadores configuráveis ilustrados na Figura 1:

1.  $N_s$ : utilizado para definir os limiares da região no entorno da média  $\mu$  que engloba amostras consideradas pertencentes à distribuição. Esses limiares são dados pela



**Figura 1. Limiares para classificação de amostras de RSSI utilizados pelo NDIST.**

- multiplicação de  $N_s$  pelo desvio padrão  $S$ , positivo e negativo, definindo a região que engloba amostras com valores entre  $-N_s \times S$  e  $N_s \times S$ , estando estes incluídos.
2.  $N_{sout}$ : utilizado para definir os limiares das regiões que englobam *outliers*. Esses limiares são dados pela multiplicação de  $N_{sout}$  pelo desvio padrão  $S$ , positivo e negativo, definindo as regiões que englobam amostras de RSSI com valores maiores do que  $N_{sout} \times S$  e menores do que  $-N_{sout} \times S$ .

As demais amostras de RSSI serão classificadas como não pertencentes à distribuição.

Em situações em que a série temporal do RSSI sofre uma alteração brusca e permanente, pode ocorrer uma classificação incorreta de *outliers*. Isso significa que amostras que deveriam estar de fato causando alterações na distribuição podem ser, erroneamente, consideradas *outliers*, evitando a devida alteração da distribuição. Para lidar com essa possibilidade, o algoritmo trata as amostras classificadas como *outliers* da seguinte forma. Se a amostra em questão for considerada um *outlier*, ela será armazenada em uma nova janela deslizante que contém apenas amostras classificadas desta forma. Se o número de *outliers* consecutivos atingir o limiar configurável  $Max_{out}$ , uma nova média é iniciada com base nesses valores consecutivos armazenados.

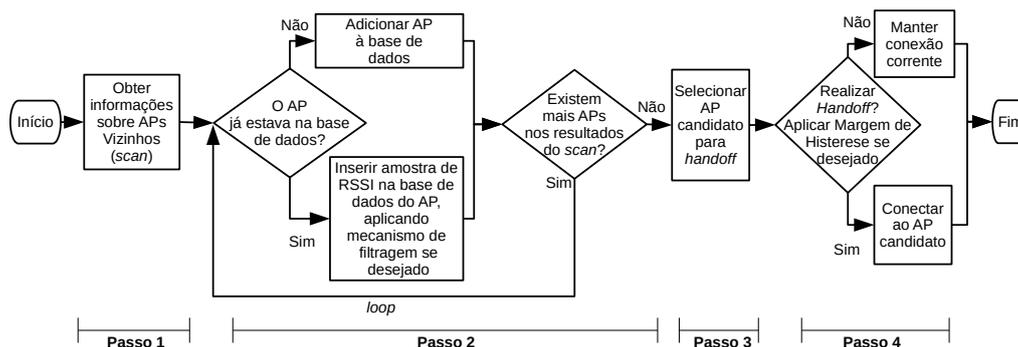
Portanto, o NDIST requer a configuração de quatro parâmetros para operar, que são: 1) tamanho da janela deslizante, denominado  $W_s$ ; 2) multiplicador do desvio padrão  $N_s$ , utilizado para definir o limiar do teste para verificar se a amostra pertence ou não à distribuição de RSSI; 3) multiplicador do desvio padrão  $N_{sout}$ , utilizado para configurar o limiar de detecção de *outliers*; 4) número máximo de *outliers* consecutivos  $Max_{out}$  que deve ocorrer para que a média da distribuição seja alterada.

## 4. Avaliação e resultados

Nesta seção, serão apresentados os resultados dos testes de avaliação do NDIST, comparando-o com outros cinco mecanismos anti ping-pong baseados em RSSI encontrados na literatura. Os testes foram realizados por meio de simulações utilizando séries temporais de RSSI (*traces*) obtidas a partir de um cenário de rede real com mobilidade. A utilização de *traces* de RSSI foi preferida em relação a testes em redes reais para garantir uma comparação justa entre os algoritmos. Ao utilizar *traces* pré-capturados, todos os algoritmos puderam ser testados com a mesma amostra de RSSI, como se estivessem sendo executados simultaneamente no mesmo espaço/tempo.

### 4.1. Descrição do simulador

O simulador, desenvolvido como parte deste trabalho, baseia-se na estrutura de um algoritmo amplamente utilizado atualmente, que é o mecanismo de *handoff* do WPA Supplicant v2.1 descrito em [Balbi et al. 2020]. O simulador recebe *traces* de amostras de RSSI



**Figura 2. Fluxograma genérico de operação dos algoritmos de *handoff*.**

e simula uma STA que realiza periodicamente *scans* para avaliar APs próximos. Após cada *scan*, a STA deve escolher entre permanecer associada ao seu AP atual ou realizar o *handoff* para um novo AP.

A Figura 2 apresenta um fluxograma ilustrando como funciona um ciclo, desde o *scan* até a decisão de *handoff*, no simulador. No *Passo 1*, as informações de RSSI são obtidas a partir de um *scan*. No *Passo 2*, essas informações são adicionadas ao banco de dados de APs, após o processamento da amostra por meio de um mecanismo de filtragem, se desejado. Este passo é executado em um *loop* que itera por cada AP encontrado durante o *scan*. No *Passo 3*, o algoritmo escolhe um AP candidato para um possível *handoff*, com base em seu RSSI. No *Passo 4*, o algoritmo decide se o *handoff* deve ser realizado ou não. Neste passo, os RSSIs dos APs atual e candidato são comparados, de modo que aquele com o valor mais alto será escolhido. Uma margem de histerese pode ser aplicada, como é o caso do WPA Supplicant e de outros mecanismos que utilizam essa característica.

Para a implementação do NDIST, assim como a de outros mecanismos de filtragem, a função que armazena o RSSI na base de dados do programa, para a tomada de decisão de *handoff*, indicada no Passo 2 da figura, foi alterada. No WPA Supplicant, o RSSI é armazenado diretamente na base de dados, sem a utilização de mecanismos de filtragem. Já no NDIST, caso o AP não esteja na base de dados, uma janela deslizante é criada para ele e a informação de RSSI é adicionada à janela. Caso o AP já esteja na base de dados, a nova amostra de RSSI será adicionada à janela deslizante deste AP de acordo com as regras do algoritmo. Por fim, no Passo 4, o NDIST não utiliza mecanismo de margem de histerese para evitar ping-pongs, como ocorre no WPA Supplicant.

Os *traces* de RSSI utilizados nos testes foram obtidos a partir de *beacons* coletados no ambiente de teste FIBRE<sup>2</sup>, localizado na UFF, que possui um trem capaz de percorrer um corredor reto de 60 metros, transportando um *laptop*, em aproximadamente 50 segundos, ou seja, a uma velocidade de  $1,2m/s$ , equivalente à de uma pessoa caminhando rapidamente. O sistema operacional do *laptop* era o Linux Ubuntu 14.04 e a interface Airpcap-NX<sup>3</sup> foi utilizada para a captura de pacotes. O trem foi programado para percorrer a rota 45 vezes entre 0h e 6h da manhã.

O simulador possui vários parâmetros, entre os quais os mais importantes são<sup>4</sup>:

<sup>2</sup><https://fibre.org.br/start-using-fibre/register/uff-island/>

<sup>3</sup>[https://wikidevi.com/wiki/CACE\\_Technologies\\_AirPcap\\_Nx](https://wikidevi.com/wiki/CACE_Technologies_AirPcap_Nx)

<sup>4</sup>Mais detalhes em <https://github.com/helgadb/MAXsimulator>

- Intervalo de *Scan*: intervalo de tempo entre o início das varreduras realizadas pela STA. *Beacons* são amostrados periodicamente neste intervalo. Nos testes, este intervalo foi configurado para 0,1024 segundos, ou seja, o intervalo de *beacon* comumente utilizado pelos APs;
- Deslocamento (*Offset*): este parâmetro permite a criação de diferentes instâncias de teste a partir do mesmo *trace*. Isso é possível adicionando-se um pequeno valor aleatório ao tempo de início da varredura, assim como uma pequena variação ao intervalo de varredura.

Por fim, o simulador gera estatísticas para cada algoritmo e parâmetro testado, incluindo o número de *handoffs* e o momento do último *handoff*.

## 4.2. Métricas de avaliação e cenários

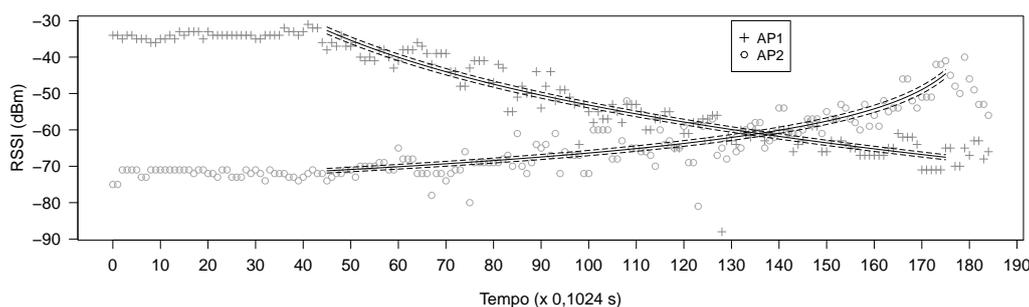
A Figura 3 ilustra um dos cenários de mobilidade utilizados nesta avaliação. Nela, podem ser vistos dois dos *traces* de RSSI usados nos testes, referentes a dois APs intitulados AP1 e AP2. O eixo *Y* indica o RSSI de cada amostra e o eixo *X* indica o tempo, dado em intervalos de *scan* (0,1024 s). Este exemplo inclui aproximadamente 190 intervalos, o que leva a aproximadamente 190 decisões de *handoff* para cada simulação. Assim, cada simulação durou aproximadamente 19 segundos. Este intervalo de tempo de simulação foi escolhido para fornecer um cenário de mobilidade no qual os *traces* de RSSI dos dois APs se cruzam apenas uma vez, facilitando a análise. Neste cenário, a STA se associa inicialmente ao AP1 e move-se em direção ao AP2. Assim, espera-se que apenas um *handoff* seja realizado do AP1 para o AP2.

As seguintes métricas de desempenho foram consideradas na análise:

- Percentual de *erros de estabilização*, ou seja, percentual de casos, em relação ao número total de instâncias simuladas, em que ocorreu um erro de estabilização. Ambos os casos de não estabilização e estabilização precoce são considerados erros de estabilização. Uma estabilização precoce ocorre quando a STA realiza o último *handoff* antes do momento ideal para *handoff*, e uma não estabilização ocorre quando a STA termina a simulação associada ao AP1 e não ao AP2.
- Número médio de ping-pongs, considerando apenas instâncias de testes em que não ocorreram erros de estabilização.
- Atraso médio de *handoff* contado a partir do momento ideal para *handoff*. Aqui, novamente são desconsideradas as instâncias com erros de estabilização. Esse atraso é a soma do atraso de acionamento (*triggering*) do *handoff* com o atraso de estabilização da STA que ocorrerá apenas se a STA sofrer ping-pongs.

Para todas as métricas de desempenho envolvendo médias, foram calculados intervalos de confiança de 95% como uma estimativa das margens de erro. O momento ideal para o *handoff* foi calculado estimando-se o ponto de interseção entre os *traces* de RSSI dos dois APs, AP1 e AP2. A estimativa foi realizada utilizando-se o modelo de perda *log-distance path loss* [Alim et al. 2010], que descreve a degradação do sinal com o logaritmo da distância. O método dos mínimos quadrados foi usado para ajustar esse modelo para cada *trace* de RSSI, e a interseção entre as curvas resultantes foi calculada.

Para cada curva, também foram calculados os limites superior e inferior de confiança de 95%. O ponto de interseção entre o limite inferior da curva do AP1 e o



**Figura 3. Exemplo de *traces* usados nos testes, incluindo as curvas modeladas para a estimativa do momento ideal de *handoff* (em preto sólido) e as curvas dadas pelas margens de erro do modelo (em preto pontilhado).**

limite superior da curva do AP2 é usado como limite inferior para a estimativa, a fim de considerar as incertezas das medições. Da mesma forma, a interseção entre o limite superior da curva do AP1 e o limite inferior da curva do AP2 é usada como limite superior. Essas curvas também estão representadas na Figura 3: as curvas em preto sólido denotam o melhor ajuste das amostras de RSSI de cada AP ao modelo *log-distance path loss*, enquanto as curvas em preto pontilhado denotam seus respectivos limites de confiança de 95%, superior e inferior. Neste exemplo, o momento ideal para o *handoff* é de 135,73 intervalos de *scan* (interseção das linhas pretas), enquanto os limites inferior e superior para a estimativa são 131,63 e 139,77 intervalos de *scan*, respectivamente. Nesta figura, pode-se observar que, no início da simulação, a STA permaneceu estática por um tempo, e, no final da simulação, a STA passou pelo AP2, o que pode ser percebido pela queda do RSSI deste AP.

O valor absoluto do atraso de *handoff* foi calculado considerando o último *handoff* detectado durante a simulação<sup>5</sup>. Este *handoff* pode ocorrer antes ou depois do momento ideal. Se o *handoff* ocorrer antes do limite inferior da estimativa, a instância é contada como um erro de estabilização precoce.

Considerando que as simulações em cenários móveis duram cerca de 19 segundos — um intervalo de tempo que pode ser considerado muito curto para mais de um *handoff* entre AP1 e AP2 — e que os *traces* de RSSI cruzaram apenas uma vez, o número de ping-pongs foi calculado como o número de *handoffs* extras realizados nesses cenários, além do único *handoff* esperado.

Filtros que geram séries temporais mais estáveis geralmente geram um atraso maior para acionar um *handoff* desejado devido ao maior tempo de resposta do filtro, e vice-versa. Dessa forma, as métricas de atraso e o número de ping-pongs podem ser consideradas critérios de avaliação conflitantes, ou seja, geralmente, uma métrica não pode ser melhorada sem que a outra piore. Devido a esse fato, a fronteira de Pareto [Lotov et al. 2013] foi escolhida para detectar os melhores resultados.

Finalmente, para decidir o melhor mecanismo entre os selecionados como Paretos ótimos, uma regra arbitrária baseada na distância entre cada um deles e o ponto de origem (0,0) do espaço de critérios será usada, de modo que aquele com a menor distância será considerado o melhor. Nos testes, este espaço, que será usado para definir o melhor

<sup>5</sup>Na análise, o atraso é dado em número de intervalos de *scan*, com cada intervalo tendo aproximadamente 0,1024 segundos.

mecanismo de *handoff*, é dado pelos resultados médios de atraso de *handoff* vs. número médio de ping-pongs obtidos para cada algoritmo/parâmetro testado.

### 4.3. Descrição dos algoritmos testados

Os seguintes mecanismos de prevenção de ping-pong baseados em RSSI foram considerados nas avaliações:

- NDIST - Testado com todas as combinações de parâmetros nas seguintes faixas:  $N_s = \{0,5; 0,7; 1,0; 1,2; 1,5\}$ ;  $N_{sout} = \{4; 5\}$ ;  $Max_{out} = \{4; 6; 8; 10\}$ ;  $W_s = \{4; 6; 8; 10; 12; 14\}$ .
- WPA - O algoritmo original do WPA Supplicant (versão 2.1), descrito em [Balbi et al. 2020], que busca evitar ping-pongs utilizando uma margem de histerese que varia de acordo com o RSSI do AP atual;
- EWMA - Filtro de média móvel exponencialmente ponderada proposto em [Mhatre and Papagiannaki 2006], com valores de  $\alpha$  variando de 0,01 a 0,99 em incrementos de 0,01;
- HM - Mecanismo de Margem de Histerese, também descrito em [Mhatre and Papagiannaki 2006] com valores de margem entre 1 e 20 variando em incrementos de 1;
- MODE - Filtro baseado na moda<sup>6</sup>, ou seja, o valor mais frequente em uma amostra. Nesta implementação, o valor da moda das amostras de RSSI é escolhido a partir de uma janela deslizante. Tamanhos de janela  $W_s$  entre 3 e 22 foram usados nos testes com valores variando em incrementos de 1. Um filtro baseado na moda foi usado anteriormente em [Pu and Chung 2008];
- MEDIAN - Filtro baseado na mediana calculada a partir de uma janela deslizante. Tamanhos de janela ímpares  $W_s$  entre 3 e 31 foram usados nos testes com valores variando em incrementos de 1. Um filtro baseado na mediana foi usado anteriormente em [Kim et al. 2006].

Esses mecanismos foram selecionados considerando-se a baixa complexidade de implementação e operação em dispositivos móveis.

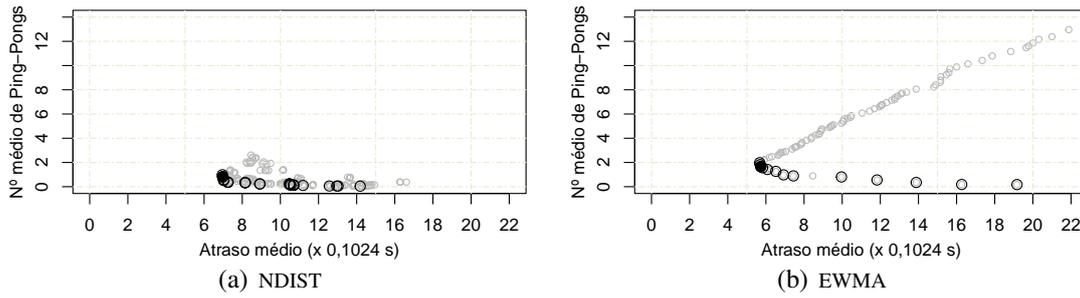
#### 4.3.1. Pré-seleção de parâmetros dos algoritmos

Devido ao grande número de parâmetros dos algoritmos selecionados para esta avaliação, uma pré-seleção de parâmetros ótimos foi realizada. O objetivo foi escolher os parâmetros considerados Pareto ótimos em relação ao número de ping-pongs e atraso de *handoff* para cada um dos algoritmos testados em um cenário com mobilidade. Esse tipo de cenário é considerado o mais importante aqui, pois a mobilidade está relacionada a uma maior ocorrência de ping-pongs, como concluído em [Balbi et al. 2020].

Para realizar essa seleção, instâncias de um cenário de mobilidade semelhante à apresentada na Figura 3 foram simuladas. Essas instâncias foram geradas a partir de 45 *traces* de RSSI para pares de APs. Para cada um desses pares, 45 *offsets* diferentes foram simulados, gerando  $45 \times 45 = 2025$  instâncias de teste.

A Figura 4(a) exemplifica a fronteira de Pareto obtida para o algoritmo NDIST. O eixo X representa o atraso médio de *handoff*, enquanto o eixo Y representa o número médio de ping-pongs. Cada círculo cinza representa o resultado obtido para uma combinação de parâmetros diferentes do NDIST. Os círculos pretos são os Pareto ótimos, e o ponto preto é o melhor resultado de acordo com o critério arbitrário escolhido, ou seja, a menor distância para o ponto de origem (0,0) entre os pontos na fronteira de Pareto. Nesta figura, pode-se observar um vértice na fronteira de Pareto. Para pontos com um número de ping-pongs acima desse vértice, o atraso de *handoff* tende a aumentar. Isso

<sup>6</sup><https://itl.nist.gov/div898/handbook/eda/section3/eda351.htm>



**Figura 4. Espaços de critérios obtidos no primeiro cenário de mobilidade para (a) NDIST, (b) EWMA. Parâmetros que são Paretos ótimos estão marcados com um círculo preto, formando a fronteira de Pareto.**

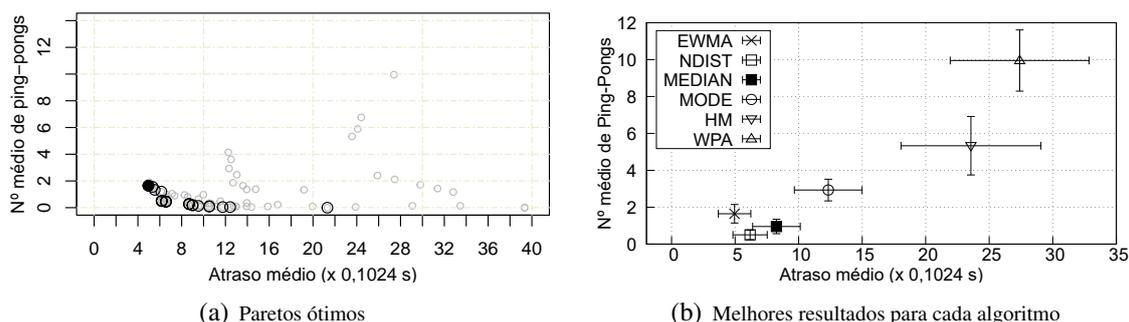
**Tabela 1. Algoritmos de *handoff* e parâmetros pré-selecionados para testes.**

Algoritmo	Descrição
WPA	Algoritmo Original do WPA Supplicant.
EWMA	Filtro EWMA com $\alpha = \{0,76; 0,78 \text{ até } 0,81; 0,83 \text{ até } 0,85; 0,87 \text{ até } 0,94; 0,98; 0,99\}$
HM	Margem de histerese de RSSI igual a 9 até 11 e 16 até 20.
MODE	Filtro baseado na moda com $W_s = 4,6,7,9,14,15,16,17 \text{ e } 22$ .
MEDIAN	Filtro baseado na mediana com $W_s = 9,11,13,15,17,19,21,23 \text{ e } 31$ .
NDIST	Filtro baseado na distribuição normal com os seguintes parâmetros combinados: $N_s = \{0,5\}; N_{sout} = \{4\}; W_s = \{12\}; Max_{out} = \{4; 6\}$ $N_s = \{0,5\}; N_{sout} = \{5\}; W_s = \{8\}; Max_{out} = \{8; 10\}$ $N_s = \{0,5\}; N_{sout} = \{5\}; W_s = \{10\}; Max_{out} = \{4; 6; 8; 10\}$ $N_s = \{0,5\}; N_{sout} = \{5\}; W_s = \{12\}; Max_{out} = \{4; 6; 10\}$ $N_s = \{0,7\}; N_{sout} = \{4\}; W_s = \{6\}; Max_{out} = \{4\}$ $N_s = \{0,7\}; N_{sout} = \{5\}; W_s = \{10\}; Max_{out} = \{4; 6; 8; 10\}$ $N_s = \{0,7\}; N_{sout} = \{5\}; W_s = \{14\}; Max_{out} = \{4; 6; 8; 10\}$ $N_s = \{1,0\}; N_{sout} = \{4\}; W_s = \{12\}; Max_{out} = \{4; 6\}$ $N_s = \{1,2\}; N_{sout} = \{4\}; W_s = \{10\}; Max_{out} = \{6\}$ $N_s = \{1,2\}; N_{sout} = \{4\}; W_s = \{12\}; Max_{out} = \{4\}$ $N_s = \{1,2\}; N_{sout} = \{4\}; W_s = \{14\}; Max_{out} = \{8\}$ $N_s = \{1,2\}; N_{sout} = \{5\}; W_s = \{10\}; Max_{out} = \{4; 6; 8; 10\}$ $N_s = \{1,5\}; N_{sout} = \{4\}; W_s = \{10\}; Max_{out} = \{6; 10\}$ $N_s = \{1,5\}; N_{sout} = \{4\}; W_s = \{12\}; Max_{out} = \{4\}$

pode ser explicado pela ocorrência de ping-pongs que são realizados após o momento ideal para o *handoff*. Como o atraso de *handoff* é calculado com base no último *handoff* realizado (ou seja, após a estabilização da STA), esse atraso cresce com o número de ping-pongs. Por outro lado, para pontos com um número de ping-pongs abaixo desse vértice, os atrasos também são tipicamente mais longos devido ao aumento do tempo de resposta do filtro. Esse comportamento pode ser observado nos outros algoritmos testados, como exemplificado na Figura 4(b), que mostra a fronteira de Pareto para EWMA.

Além disso, observando a dispersão dos resultados, podemos notar que o NDIST apresenta menos sensibilidade aos parâmetros em relação aos outros algoritmos, ou seja, o NDIST mostrou menos variabilidade no atraso de *handoff* e ping-pongs para os diferentes parâmetros testados. Isso pode ser observado pela localização dos pontos nos gráficos da Figura 4. Os resultados do NDIST estão acumulados em uma faixa menor do gráfico em relação ao EWMA, por exemplo, que alcançou valores mais altos tanto no eixo X quanto no eixo Y. Isso indica que o NDIST apresentou menos variabilidade nos resultados em relação aos outros algoritmos para seus diferentes parâmetros testados. Devido a isso, o NDIST será menos afetado por uma decisão ruim ao escolher o valor do parâmetro.

A Tabela 1 apresenta os algoritmos e seus parâmetros considerados Paretos



**Figura 5. Espaço de critérios com (a) Paretos ótimos marcados em círculos pretos e melhor resultado geral marcado como um ponto preto e (b) melhores resultados para cada algoritmo com intervalo de confiança de 95%.**

ótimos. No total, considerando todos os algoritmos, 77 parâmetros foram classificados como Paretos ótimos e selecionados para uma avaliação mais detalhada que será apresentada a seguir.

#### 4.4. Resultados dos Testes em Cenários com Mobilidade

Para avaliar cenários de mobilidade, 45 instâncias semelhantes àquela mostrada na Figura 3 e descrita na Seção 4.2 foram utilizadas. Para cada instância, os seis algoritmos com todos os parâmetros listados na Tabela 1 foram executados.

A Figura 5(a) apresenta o espaço de critérios em relação ao atraso médio de *handoff* e ao número médio de ping-pongs obtidos nas 45 instâncias de teste, incluindo todos os resultados obtidos por todos os algoritmos/parâmetros. Os círculos cinzas representam os resultados obtidos para cada algoritmo/parâmetro, e os círculos pretos representam os resultados que são Paretos ótimos. O ponto preto representa o Pareto ótimo com a menor distância ao ponto (0,0) do gráfico. Neste cenário, o melhor resultado, segundo os critérios estabelecidos, foi obtido pelo EWMA com  $\alpha = 0,79$ , com um atraso médio de *handoff* de  $4,95 \pm 1,27$  intervalos de *scans* e um número médio de ping-pongs de  $1,64 \pm 0,50$ .

A Tabela 2 apresenta os Paretos ótimos obtidos nos testes com mobilidade em termos de atraso e número de ping-pongs, ordenados pela distância até o ponto de origem (0,0) do espaço de critérios. As colunas *Erros* e *Dist* relatam as porcentagens de erros de estabilização e a distância até o ponto de origem (0,0) do espaço de critérios, respectivamente. Nota-se que, apesar de ter obtido um atraso médio ligeiramente maior em relação ao EWMA, o NDIST obteve resultados mais estáveis no geral, com números de ping-pongs menores. O algoritmo MEDIAN obteve os Paretos ótimos com maior estabilidade, porém, apresentando também os maiores atrasos. Em relação aos erros de estabilização, dentre os Paretos ótimos, apenas o NDIST apresentou tais casos, embora com uma baixa porcentagem de ocorrência. Esses erros foram referentes a estabilizações precoces. Ainda ao se observar a Tabela 2, pode-se notar que diferentes valores do parâmetro  $Max_{out}$  do NDIST geraram os mesmos resultados.

A Figura 5(b) mostra um gráfico comparativo entre os melhores resultados (sem considerar se eram Paretos ótimos ou não) obtidos por cada algoritmo através do gráfico da média do atraso vs. o número médio de ping-pongs. Os parâmetros com melhores resultados foram: EWMA com  $\alpha = 0,79$ ; NDIST  $N_s=0,5$   $N_{sout} = 5$   $W_s = 10$   $Max_{out} = \{4;6;8;10\}$ ; MEDIAN com  $W_s=9$ ; MODE com  $W_s=7$ ; e HM=11. Pode-

**Tabela 2. Paretos ótimos gerais ordenados pela distância.**

Algoritmo	Atraso	Ping-Pongs	Erros (%)	Dist.
EWMA $\alpha = 0,79$	$4,95 \pm 1,27$	$1,64 \pm 0,50$	0	5,22
EWMA $\alpha = 0,8$	$5,33 \pm 1,30$	$1,55 \pm 0,46$	0	5,55
EWMA $\alpha = 0,81$	$5,53 \pm 1,38$	$1,33 \pm 0,48$	0	5,69
NDIST $N_s = 0,5; N_{sout} = 5; W_s = 10; Max_{out} = \{4; 6; 8; 10\}$	$6,17 \pm 1,35$	$0,50 \pm 0,25$	0	6,19
EWMA $\alpha = 0,83$	$6,14 \pm 1,54$	$1,20 \pm 0,45$	0	6,25
NDIST $N_s = 0,7; N_{sout} = 5; W_s = 10; Max_{out} = \{4; 6; 8; 10\}$	$6,56 \pm 1,39$	$0,45 \pm 0,25$	0,02	6,57
NDIST $N_s = 0,5; N_{sout} = 5; W_s = 12; Max_{out} = \{4; 6; 10\}$	$8,67 \pm 1,72$	$0,27 \pm 0,20$	0,02	8,67
NDIST $N_s = 0,5; N_{sout} = 4; W_s = 12; Max_{out} = \{4; 6\}$	$8,99 \pm 1,63$	$0,18 \pm 0,17$	0,02	8,99
NDIST $N_s = 1,2; N_{sout} = 4; W_s = 10; Max_{out} = 6$	$9,52 \pm 2,08$	$0,13 \pm 0,15$	0,02	9,52
NDIST $N_s = 1,0; N_{sout} = 4; W_s = 12; Max_{out} = \{4; 6\}$	$10,52 \pm 1,58$	$0,09 \pm 0,12$	0,02	10,52
MEDIAN $W_s = 17$	$11,74 \pm 1,87$	$0,04 \pm 0,08$	0,02	11,74
MEDIAN $W_s = 19$	$12,42 \pm 1,89$	$0,04 \pm 0,08$	0	12,42
MEDIAN $W_s = 31$	$21,31 \pm 1,90$	0	0	21,31

**Tabela 3. Dados relacionados à porcentagem de ocorrências de erros de estabilização para todos os parâmetros testados de cada algoritmo.**

Algoritmo	Mínimo	Máximo	Mediana	1° Quartil	3° Quartil
EWMA	0	93,33	0	0	0
HM	17,77	22,22	21,11	18,88	22,22
NDIST	0	2,22	2,22	2,22	2,22
MODE	0	8,88	2,22	0	4,44
WPA	6,66	6,66	6,66	6,66	6,66
MEDIAN	0	2,22	0	0	2,22

se notar que, o atraso obtido pelo NDIST encontra-se dentro da margem de erro do EWMA, porém o NDIST apresentou maior estabilidade de forma expressiva em relação ao EWMA. O WPA obteve o pior resultado, com atraso médio de  $27,39 \pm 5,46$  e número médio de ping-pongs de  $9,95 \pm 1,66$ .

A Tabela 3 apresenta valores mínimo, máximo, mediana e primeiro e terceiro quartis da porcentagem de ocorrência de erros de estabilização para todos os parâmetros testados de cada algoritmo. Cada porcentagem de ocorrência foi calculada a partir de um total de 45 instâncias de teste. Pode-se observar que todos os algoritmos apresentaram casos de erros de estabilização em proporções diferentes para alguns dos parâmetros avaliados. O NDIST obteve baixas porcentagens de ocorrência, apresentando apenas estabilizações precoces em 2,22% das instâncias de teste para 29 das 32 combinações de parâmetros testados. Os demais algoritmos, exceto MEDIAN, apresentaram desempenho pior nesse aspecto. Entre essas instâncias, o NDIST nunca resultou em não estabilização, sofrendo apenas de estabilização precoce. No entanto, tanto EWMA quanto HM foram afetados pela não estabilização. Para o EWMA, a não estabilização ocorreu para  $\alpha \in \{0,93; 0,94; 0,98; 0,99\}$ . O EWMA foi particularmente suscetível a esse problema para  $\alpha = 0,98$  e  $\alpha = 0,99$ , ambos resultando em não estabilização em 42 das 45 instâncias de teste. Para o HM, a não estabilização ocorreu para margens de 9, 10, 11, 16, 17, 18 e 19 dB, sendo os piores casos ocorrendo para 16, 17 e 18 dB, que sofreram não estabilização em 4,44% das instâncias de teste. Deve-se notar que a não estabilização pode ser considerada um erro mais grave, pois a STA falha em migrar para o AP2, enquanto na estabilização precoce, o *handoff* ocorre corretamente, porém antes do momento ideal definido neste trabalho.

## 5. Conclusões

Este trabalho apresentou o NDIST, um mecanismo de filtragem eficiente para mitigar o problema da instabilidade de associação em redes densas IEEE 802.11. Embora este seja

um problema conhecido e estudado por mais de uma década, os mecanismos de *handoff* baseados em RSSI utilizados em dispositivos atuais ainda sofrem com o efeito ping-pong.

O desempenho do NDIST foi comparado a outros cinco mecanismos de filtragem encontrados na literatura, utilizando um simulador alimentado por *traces* de redes reais. Os resultados mostraram que, dentre os mecanismos testados, o NDIST foi capaz de proporcionar uma maior estabilidade de associação em cenários com mobilidade sem ocasionar aumento expressivo no atraso para acionamento do *handoff*. O atraso obtido pelo NDIST foi próximo ao do filtro EWMA (dentro da margem de erro do mesmo), porém, o NDIST apresentou expressiva redução de 69,5% na ocorrência de ping-pongs. O MEDIAN obteve um número de ping-pongs próximo ao do NDIST, mas com a desvantagem de apresentar um atraso 33,7% maior.

É importante observar que o NDIST foi criado considerando-se que a série temporal de RSSI possui uma distribuição normal. Uma ideia para trabalhos futuros é uma melhor avaliação acerca das características da distribuição do RSSI e ajuste dos parâmetros do algoritmo para suportar uma possível distribuição mais adequada.

## Referências

- Alim, A., Rahman, M. M., Hossain, M. M., and Al-Nahid, A. (2010). Analysis of large-scale propagation models for mobile communications in urban area. *International Journal of Computer Science and Information Security*, 7(1):135–139.
- Balbi, H., Passos, D. G., Carrano, R., Magalhães, L., and Albuquerque, C. V. N. (2019). A Case Study of Association Instability in Dense IEEE 802.11 Networks. In *IEEE Symposium on Computers and Communications (ISCC)*, pages 1–6. IEEE.
- Balbi, H. D., Passos, D., Carrano, R. C., Magalhães, L. C., and Albuquerque, C. V. (2020). Association stability and handoff latency tradeoff in dense IEEE 802.11 networks: A case study. *Computer Communications*, 159:175 – 185.
- Bangolae, S., Bell, C., and Qi, E. (2006). Performance Study of Fast BSS Transition Using IEEE 802.11r. In *International conference on Wireless communications and mobile computing (IWCMC)*, page 737–742, New York, NY, USA. Association for Computing Machinery.
- Bellavista, P., Corradi, A., and Giannelli, C. (2006). Evaluating filtering strategies for decentralized handover prediction in the wireless internet. In *Symposium on Computers and Communications (ISCC)*, pages 167–174. IEEE.
- Hu, T., Xue, K., Wei, W., and Jiang, W. (2015). LENV: A new light-weighted edge network virtualization framework in software-defined wireless networks. In *International Conference on Wireless Communications & Signal Processing*, pages 1–6.
- Huang, P.-J., Tseng, Y.-C., and Tsai, K.-C. (2006). A Fast Handoff Mechanism for IEEE 802.11 and IAPP Networks. In *Vehicular Technology Conference*, volume 2, pages 966–970.
- IEEE (2012). 802.11-2012 IEEE standard for information technology – LAN/MAN – specific requirements – part 11: Wireless LAN medium access control (MAC) and physical layer (PHY) specification.

- Kim, H.-S., Park, S.-H., Park, C.-S., Kim, J.-W., and Ko, S.-J. (2006). Fast handoff scheme for seamless multimedia service in wireless LAN. In *International Conference on Research in Networking*, pages 942–953. Springer.
- Kim, M., Liu, Z., Parthasarathy, S., Pendarakis, D., and Yang, H. (2012). Association Control Algorithms for Handoff Frequency Minimization in Mobile Wireless Networks. *Wireless Networks*, 18(5):535–550.
- Levis, K. (2006). RSSI is under appreciated. In *Proceedings of the Third Workshop on Embedded Networked Sensors*, pages 239–242.
- Lotov, A. V., Bushenkov, V. A., and Kamenev, G. K. (2013). *Interactive decision maps: Approximation and visualization of Pareto frontier*, volume 89. Springer Science & Business Media.
- Mhatre, V. and Papagiannaki, K. (2006). Using Smart Triggers for Improved User Performance in 802.11 Wireless Networks. In *Proceedings of MobiSys*, pages 246–259.
- Mishra, A., Shin, M., and Arbaugh, W. (2003). An Empirical Analysis of the IEEE 802.11 MAC Layer Handoff Process. *SIGCOMM Comput. Commun. Rev.*, 33(2):93–102.
- Montavont, N., Blanc, A., Navas, R., Kerdoncuff, T., and Castignani, G. (2015). Handover triggering in IEEE 802.11 networks. In *WoWMoM*, pages 1–9.
- Pu, C. and Chung, W. (2008). Mitigation of Multipath Fading Effects to Improve Indoor RSSI Performance. *IEEE Sensors Journal*, 8(11):1884–1886.
- Raghavendra, R., Belding, E. M., Papagiannaki, K., and Almeroth, K. C. (2007). Understanding Handoffs in Large IEEE 802.11 Wireless Networks. In *Proceedings of IMC*, pages 333–338.
- Ramani, I. and Savage, S. (2005). SyncScan: practical fast handoff for 802.11 infrastructure networks. In *INFOCOM*, volume 1, pages 675–684.
- Rangiseti, A. K., Baldaniya, H. B., B, P. K., and Tamma, B. R. (2014). Load-aware hand-offs in software defined wireless LANs. In *10th IEEE WiMob*, pages 685–690.
- Sklar, B. (1997). Rayleigh fading channels in mobile digital communication systems .I. Characterization. *IEEE Communications Magazine*, 35(7):90–100.
- Smailagic, A., Small, J., and Siewiorek, D. P. (2000). Determining user location for context aware computing through the use of a wireless LAN infrastructure. Technical report, Institute for Complex Engineered Systems Carnegie Mellon University, Pittsburgh, PA.
- Vallati, C., Mingozzi, E., and Benedetto, C. (2016). Efficient handoff based on link quality prediction for video streaming in urban transport systems. *Wireless Communications and Mobile Computing*, 16(15):2298–2314.
- Vasudevan, S., Papagiannaki, K., Diot, C., Kurose, J., and Towsley, D. (2005). Facilitating Access Point Selection in IEEE 802.11 Wireless Networks. In *Proceedings of IMC*, pages 293–298.
- Weng, Z. and Xie, Z. (2024). Ap selection game in dense ieee 802.11 wlans. *Wireless Networks*, pages 1–16.