

# FS-MAC: Uma Plataforma para a Flexibilização da Subcamada MAC em Redes Sem Fio

Jefferson R. S. Cordeiro, Esthefanie Lanza, Daniel F. Macedo, Luiz F. M. Vieira

<sup>1</sup>Departamento de Ciência da Computação – Universidade Federal de Minas Gerais  
Belo Horizonte, Minas Gerais

{jeff, esthefanie.lanza, damacedo, lfvieira}@dcc.ufmg.br

**Abstract.** *Current wireless networks are very dynamic and have a wide variety of applications with very different needs. Applications such as voice and video, for example, require low latency to ensure quality in the transmission, but tolerate reasonable error rates. On the other hand, web and message services have low error tolerance, however don't require low latency. This diversity raises the need for more flexible equipment as well as networks that adapt to the context of the network. This work make the medium access control more flexible, allowing more than one MAC protocol to be used on the network. This operates by activating each one in the situation where it is most efficient. FS-MAC is also extensible, allowing the addition of new protocols. The proposal was tested in a testbed, where we varied the load and the number of connected stations.*

**Resumo.** *As redes sem fio atuais são muito dinâmicas e abrigam uma grande diversidade de aplicações com necessidades bem diferentes. Aplicações de voz e vídeo, por exemplo, exigem uma latência baixa para que haja qualidade na transmissão, mas suportam taxas razoáveis de erro. Por outro lado, serviços de mensagem e web são pouco tolerantes a erros, mas não requerem baixa latência. Essa diversidade faz surgir a necessidade por equipamentos e redes mais flexíveis, que se adaptem às características do contexto. O presente trabalho flexibiliza o controle de acesso ao meio, permitindo que mais de um protocolo MAC possa ser utilizado na rede, colocando cada um em atividade na situação em que é mais eficiente. O FS-MAC<sup>1</sup> ainda é extensível, permitindo a adição de novos protocolos. A proposta foi testada em um testbed, onde variamos a carga e o número de estações conectadas.*

## 1. Introdução

As redes sem fio atuais são muito dinâmicas e suas características possibilitam a existência de uma grande diversidade de aplicações. Consideremos, por exemplo, uma rede de celular que promove a comunicação entre diversos dispositivos, entre eles, *smartphones*. Esses aparelhos possuem aplicações que executam chamadas de voz, chamadas de vídeo, exibição de páginas web, aplicativos de troca de mensagens, etc. As aplicações de chamada de voz e vídeo possuem uma tolerância razoável a erros, pois a perda de alguns pacotes não compromete o entendimento da mensagem por parte do usuário. Por outro lado, essas aplicações exigem uma latência pequena, pois no nível do usuário, atrasos grandes prejudicam muito este tipo de comunicação. Já as aplicações

---

<sup>1</sup>Código disponível em <https://github.com/jeffRayneres/FS-MAC>

web e de mensagem de texto por exemplo, suportam latências maiores. No entanto, a tolerância a erros nesses casos é pequena, pois a falta mínima de alguns dados pode comprometer a exibição e o entendimento do conteúdo. Assim, torna-se desejável que a rede adapte o seu comportamento de acordo com as necessidades das aplicações, alterando dinamicamente parâmetros de um protocolo, por exemplo.

Na subcamada MAC há também essa demanda por flexibilidade, pois os protocolos existentes não são eficientes em todos os contextos [Sharp et al. 1995, Rhee et al. 2008]. Isto ocorre porque é impossível que um único protocolo consiga atender a requisitos ortogonais como largura de banda, latência, consumo de energia, disponibilidade e segurança. Assim, percebemos uma necessidade por redes e dispositivos mais flexíveis que possam se adaptar ao contexto da comunicação para maximizar o aproveitamento dos recursos e atender corretamente aos requisitos das aplicações.

Uma solução adaptativa poderia empregar as diversas técnicas existentes na subcamada MAC para atender às demandas da aplicação. O MAC possui diversos algoritmos e técnicas que podem ser usados de acordo com a necessidade do projetista, por exemplo: *i*) acesso com ou sem contenção; *ii*) utilização ou não controle de fluxo; *iii*) emprego de *Automatic Repeat Request* (ARQ) ou *Forward Error Correction* (FEC); *iv*) priorização do acesso ao meio; *v*) reserva de recursos; *vi*) mecanismos de segurança e privacidade.

A literatura apresenta diversas soluções para a adaptatividade. Os protocolos híbridos [Sharp et al. 1995, Rhee et al. 2008, Hu et al. 2011] ajustam parte dos seus parâmetros de operação para um conjunto limitado de situações. Plataformas programáveis [Neufeld et al. 2005, Nychis et al. 2009, Tinnirello et al. 2012] permitem o desenvolvimento de novos protocolos MAC adaptados a cada rede ou serviço, mas em geral possuem uma expressibilidade limitada. Finalmente, podemos empregar plataformas de troca de protocolos [Doerr et al. 2005], onde a plataforma escolhe qual protocolo empregar dentre uma lista de protocolos disponíveis, baseado nas demandas da rede.

Este artigo apresenta uma arquitetura híbrida e com alta flexibilidade para a subcamada MAC, chamada FS-MAC. O FS-MAC troca dinamicamente o protocolo MAC em atividade na rede, de forma a manter operando aquele que for mais eficiente para o cenário apresentado. A arquitetura é extensível, permitindo a adição de novos protocolos MAC. A troca entre protocolos é regida por um conjunto de regras, que podem ser customizadas de acordo com as necessidades da aplicação e dos administradores da rede. Implementamos um protótipo sobre o padrão IEEE 802.15.4, onde o FS-MAC escolhe entre um protocolo MAC baseado em CSMA e outro baseado em TDMA.

Os resultados mostram que o FS-MAC funciona como esperado, pois o protótipo implementado é capaz de detectar que protocolo melhor se adapta às condições da rede e quando necessário, de forma coordenada, substitui corretamente o protocolo em atividade pelo mais adequado. Os resultados mostram também que devido às mensagens de controle necessárias para o seu funcionamento, o FS-MAC apresentando uma pequena queda de desempenho em relação aos protocolos isolados.

O restante do artigo está organizado da seguinte forma: a seção 2 discute os trabalhos relacionados ao tema tratado nesse artigo; a seção 3 apresenta a arquitetura FS-MAC; a seção 4 mostra a implementação de um protótipo baseado na arquitetura proposta; a seção 5 descreve os experimentos realizados para a avaliação da arquitetura e discute os

seus resultados e por fim a seção 6 apresenta as conclusões e os trabalhos futuros.

## 2. Trabalhos Relacionados

Desde a criação de redes sem fio, diversos trabalhos foram publicados com a preocupação de melhorar o controle de acesso ao meio. Com a popularização dessas redes e a diversificação dos protocolos, várias pesquisas foram desenvolvidas no sentido de aumentar a flexibilidade e o desempenho da subcamada MAC.

Diversos trabalhos procuram otimizar os protocolos existentes. Os autores de [Puschmann et al. 2013] mostram estratégias que reduzem o parâmetro de *slot time*, mitigando os efeitos negativos da pouca capacidade de processamento de sinais. Já o OpenTDMF é inspirado no conceito de SDN [Yang et al. 2015]. O controlador associa um conjunto de *slots* de tempo e um nível de prioridade a cada fluxo. De acordo com o valor de prioridade, cada ponto de acesso determina ordens de transmissão em nível de pacote de seus fluxos locais. A abordagem reduz os efeitos de terminais expostos e escondidos e aumenta a justiça na rede. Apesar disso, a abordagem ainda apresenta um alto *overhead* quando há poucos dispositivos tentando transmitir.

Otimizar os protocolos existentes aumentam o seu desempenho. Entretanto, todo protocolo possui limitações em função das suas premissas básicas do modo de acesso ao meio. Assim, outra vertente de investigação prioriza os protocolos híbridos, em que diversos modos de acesso são incorporados em um único protocolo MAC.

Sharp et al. propuseram um protocolo híbrido, que alterna entre CSMA e TDMA [Sharp et al. 1995]. Apesar da contribuição, a solução foi implementada sobre um Sistema para demonstração de Rádio Pacotes (PRDS), um equipamento de difícil acesso fora de grandes laboratórios. Diversos protocolos híbridos foram propostos desde então. O padrão IEEE 802.11 suporta operação híbrida. O *Distributed Foundation Wireless Medium Access Control* é um protocolo híbrido implementado no IEEE 802.11 [Diepstraten and WCND-Utrecht 1993], onde a rede trabalha em um período com contenção (PCF) e outro livre de contenção (DCF). Entretanto, a proporção alocada para cada período é fixa.

O Z-MAC é um protocolo híbrido baseado em *slots* [Rhee et al. 2008]. O Z-MAC utiliza um algoritmo de escalonamento para a alocação dos *slots*. Assim, cada *slot* possui um *dono*, que tem prioridade sobre o seu uso. Quando o *slot* não é utilizado pelo dono, ele pode ser usado por outros nós. O uso de prioridades gera um efeito implícito de alternância entre CSMA e TDMA. O LA-MAC alterna entre os modos CSMA e HYBRID (baseado em TDMA) em função do nível de contenção da rede [Hu et al. 2011]. A métrica utilizada para essa decisão é a quantidade de perdas de pacotes em uma vizinhança de dois saltos. Quando em baixa contenção, todos os nós competem igualmente pelos *slots* (CSMA). Já no estado de alta contenção, os nós competem pelos *slots* com base em suas prioridades, que são baseadas na sua janela de contenção (modo HYBRID).

AGENT utiliza um protocolo baseado em contenção dentro de um protocolo livre de contenção: há um CSMA em operação dentro dos *slots* de tempo do TDMA [Myers et al. 2002]. Dessa forma, nos cenários onde a carga na rede é baixa, o protocolo tem eficiência similar ao CSMA, enquanto em cenários de maior congestionamento, se comporta similarmente ao TDMA. Apesar da flexibilidade, o protocolo se limita ao uso de uma implementação específica de TDMA e outra de CSMA, não deixando espaço para a inclusão de outros protocolos.

Apesar de serem uma evolução dos protocolos MAC clássicos, os protocolos híbridos também são imutáveis. Isso impede que o MAC acompanhe as mudanças nos requisitos das aplicações e na carga da rede.

Uma forma de flexibilizar o MAC é permitir a sua programabilidade a partir de um conjunto de instruções ou de APIs. O SoftMAC [Neufeld et al. 2005] e o FLAVIA [Tinnirello et al. 2012] permitem a reprogramação do MAC em placas de redes comerciais de baixo custo. O WiSHFUL é uma arquitetura que permite a programação de placas de diversos modelos, empregando uma linguagem comum [Ruckebusch et al. 2016]. Já em [Nychis et al. 2009] os autores identificam um conjunto de funções MAC importantes que devem ser implementadas mais próximas do rádio por razões de eficiência, e definem também uma API que permite ao *host* controlar essas funções, provendo a flexibilidade necessária para a implementação de novos protocolos MAC. Ambos os trabalhos focam na criação de novos protocolos MAC, e não na adaptação dinâmica. O *Overlay MAC Layer* (OML) adiciona uma nova camada sobre a subcamada MAC, que permite adaptar o controle de acesso ao meio a requisitos de roteamento e aplicação, por exemplo [Rao and Stoica 2005]. No entanto, ela está limitada à interface de comunicação entre a camada MAC e a camada de redes. Não é possível, por exemplo, realizar a verificação do meio, pois informações da camada física não estão disponíveis para os níveis superiores.

Uma segunda alternativa de flexibilização é o emprego de plataformas que trocam dinamicamente entre diversos protocolos. A programabilidade do MAC fornece somente algumas primitivas de alto nível por razões de desempenho, limitando as características que podem ser programadas. Enquanto isso, a troca dinâmica permite um maior grau de liberdade para os protocolos MAC.

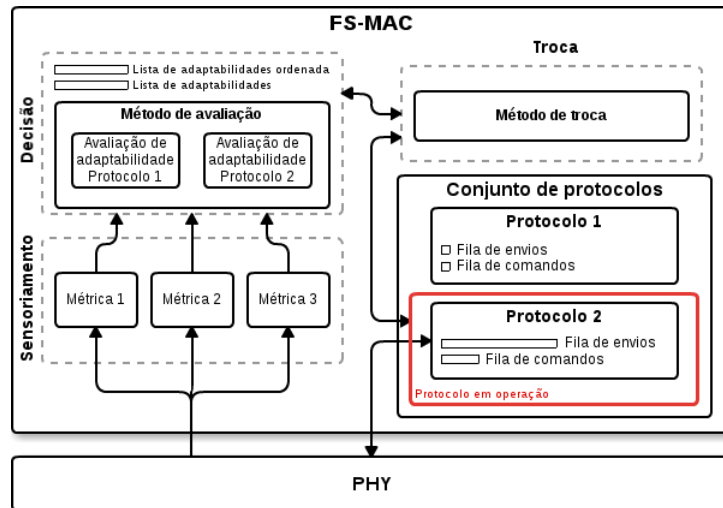
A plataforma MultiMAC pode abrigar diversos protocolos, onde cada um é utilizado na situação mais adequada [Doerr et al. 2005]. O MultiMAC permite não só o uso de CSMA em situações de baixa contenção e TDMA em situações de alta contenção, mas prevê a inclusão de novos protocolos nesse conjunto. A troca entre protocolos é baseada em regras, que avaliam quando um dado protocolo deve entrar em operação. A avaliação emprega dois protocolos, sendo um com detecção de erros (CRC) e outro com correção de erros. Os pontos fracos do trabalho são a falta de um mecanismo de sincronização entre as estações, o que é importante para redes de larga escala, pois alguns protocolos podem ser conflitantes entre si, gerando problemas de desempenho se as estações não entram em consenso sobre o melhor protocolo a ser utilizado. Assim, neste trabalho propomos um mecanismo centralizado de decisão.

### 3. A Plataforma FS-MAC

A plataforma *Flexible System for Medium Access Control* (FS-MAC) é uma arquitetura que promove a flexibilização da subcamada MAC de redes sem fio. Esta arquitetura foi desenvolvida para permitir o uso de mais de um protocolo no controle de acesso ao meio, utilizando cada um na situação em que é mais eficiente. Para tanto, ela emprega regras baseadas em lógica *fuzzy*.

O FS-MAC possui módulos e um conjunto de protocolos conforme representado na Figura 1. O módulo de *Sensoriamento* reúne dados sobre o contexto da rede e os passa ao módulo de Decisão. A decisão no FS-MAC é centralizada, ocorrendo em um nó chamado *coordenador*. O objetivo do módulo de *Decisão* é analisar os dados recebidos,

decidir qual protocolo se adapta melhor às atuais condições da rede e repassar essa análise ao módulo de Troca. Finalmente, o módulo de *Troca* determina se a troca de protocolo é necessária, caso afirmativo, ele notifica os nós da decisão tomada e executa todos os procedimentos para realizar a substituição do protocolo em operação. Um exemplo em que pode ser empregada essa solução são as redes Wi-Fi com o ponto de acesso sendo o coordenador FS-MAC.



**Figura 1. Arquitetura do FS-MAC.**

A grande vantagem do FS-MAC sobre as soluções existentes é o alto grau de extensibilidade e de configuração. O módulo de Sensoriamento permite a adição de novas métricas a serem monitoradas, que poderão ser usadas no módulo de Decisão. Já o módulo de Decisão utiliza um conjunto de regras, que podem ser trocadas pelos administradores ou pelos usuários da rede, de forma que a escolha de protocolos seja adaptada às suas necessidades. Finalmente, o FS-MAC permite a adição de novos protocolos MAC, desde que esses implementem os métodos necessários para a operação do FS-MAC.

### 3.1. Módulo de Sensoriamento

O módulo de Sensoriamento colhe informações da rede que serão usadas para decidir qual protocolo MAC será utilizado. Este módulo é composto por unidades básicas de coleta e processamento, cada unidade é responsável por coletar dados da rede, processá-los e gerar o valor de uma métrica específica. Algumas métricas possíveis são: quantidade de estações se comunicando, volume de pacotes enviados, tipo de fluxo (contínuo ou rajadas) e quantidade de pacotes confirmados.

O FS-MAC suporta tanto sensoriamento centralizado quanto sensoriamento distribuído. No caso de sensoriamento centralizado, apenas o nó coordenador realiza as coletas. Um exemplo de métrica centralizada é o número de estações transmitindo. Quando o sensoriamento é distribuído, todas as estações recolhem dados individualmente e enviam o valor calculado da métrica ao módulo de Decisão do nó coordenador. Alguns exemplos de métricas distribuídas são taxa de entrega e latência. Neste caso, cada nó da rede envia os dados coletados para o coordenador, que os repassa para o módulo de Decisão.

Embora o sensoriamento centralizado seja mais simples, a decisão de manter a possibilidade de um sensoriamento distribuído se apoia no fato de que algumas métricas

não podem ser geradas centralizadamente, pois perdem precisão nesse caso. O valor de uma métrica como taxa de entrega, por exemplo, não pode ser calculado centralizadamente, pois a um nó não é possível saber isoladamente se outro recebeu a confirmação de seus pacotes. Outra métrica como nível de contenção da rede, quando calculada centralizadamente levando em consideração dados locais, pode não corresponder à realidade da rede, principalmente durante a execução de protocolos como CSMA/CA onde não há garantia de justiça.

Cada métrica é sensoreada a partir de um submódulo específico. Isto permite que novas métricas sejam adicionadas ao FS-MAC de acordo com os protocolos MAC empregados ou a partir das necessidades da rede.

### 3.2. Módulo de Decisão

A decisão de qual protocolo MAC mais se adapta às condições da rede ocorre em um nó central, o coordenador. O módulo de Decisão recebe as informações do Sensoriamento e calcula a adaptabilidade de cada protocolo ao contexto atual da rede. Para tanto, o módulo precisa conhecer as características dos protocolos, de forma a identificar onde eles apresentam maior eficiência e as situações onde não possuem bom desempenho.

A representação das características dos protocolos é feita por regras, que são expressadas em lógica *fuzzy*. O administrador da rede pode adaptar o funcionamento do FS-MAC ao mudar as regras de cada protocolo, por exemplo, priorizando desempenho, consumo de energia, confiabilidade ou outras características dos mesmos.

O motor de decisão avalia a adaptabilidade de cada protocolo, ou seja, o quão adequado ele é para o estado atual da rede. Assim, o módulo de Decisão gera uma lista de adaptabilidade, que é repassada para o módulo de Troca. Este desenho garante o baixo acoplamento entre os dois módulos, pois o módulo de Decisão não precisa saber que protocolo está em operação no momento.

### 3.3. Módulo de Troca

O módulo de Troca verifica se é necessário modificar o protocolo MAC empregado no momento. Esta operação acontece em três passos:

1. A lista de adaptabilidades é analisada, se o protocolo em operação é diferente daquele com maior grau de adaptabilidade na lista, os dois protocolos são comparados. Para que a troca seja viável, é preciso que a diferença entre a adaptabilidade dos dois protocolos seja maior que um certo limiar. A definição desse limiar deve garantir que o custo operacional da troca possa ser compensado pela melhora de desempenho na rede. Caso a diferença seja maior, o protocolo deve ser efetivamente trocado e os passos seguintes são executados, caso contrário, mantém-se o protocolo atual.
2. Depois de concluído que deve haver a troca, o nó coordenador comunica imediatamente a todos os nós qual deve ser o novo protocolo a entrar em operação, enviando uma mensagem à rede.
3. Internamente, cada nó da rede substitui o protocolo em operação pelo novo protocolo definido.

A troca de protocolos é notificada ao protocolo atual, para que este possa terminar as suas operações em um estado consistente. Feito isso, o novo protocolo é ativado e

recebe o controle da fila de pacotes a serem enviados. Esta fila contém todos os pacotes que ainda não foram enviados e confirmados, na ordem em que aparecem na estrutura.

A efetivação da troca pode ser implementada de duas formas diferentes, dependendo de como a plataforma irá tratar a coexistência de protocolos distintos:

- Conduzir uma troca distribuída atomicamente, ou seja, enquanto houver pelo menos um nó operando no protocolo antigo, o novo não entra em operação.
- O coordenador avisa a todos os nós periodicamente qual protocolo deverá estar em atividade, minimizando assim o tempo de operação com protocolos diferentes.

Na primeira opção, nunca haverá protocolos diferentes operando ao mesmo tempo. No entanto, esta abordagem é mais complexa e lenta, e dificulta a entrada e saída de nós na rede. Assim, escolhemos a segunda abordagem. A sua desvantagem é que o uso de protocolos diferentes pode degradar o desempenho da rede. Entretanto, o aviso periódico do coordenador faz com que a quantidade de nós empregando o protocolo antigo diminua com o tempo.

### 3.4. Conjunto de Protocolos MAC

É o conjunto de protocolos MAC implementados. Para acrescentar novos protocolos a esse conjunto, devemos implementar duas interfaces: *Interface de troca* e *Interface de comunicação externa*. A interface de troca possibilita que o FS-MAC interrompa um protocolo em funcionamento e ative outro protocolo. Já a interface de comunicação externa possibilita ao FS-MAC solicitar o envio de mensagens de controle, tais como as mensagens de envio de dados de sensoriamento e as mensagens de troca de protocolo. Além disso, o gerente da rede deve escrever as regras de troca para o protocolo e instalá-las no módulo de decisão. Essas regras devem descrever o grau de adaptabilidade do protocolo, para que o protocolo mais adequado seja escolhido.

## 4. Implementação do FS-MAC em SDR

Esta seção descreve uma implementação do FS-MAC que permite a troca entre um protocolo CSMA/CA e um protocolo TDMA. Na implementação utilizamos a tecnologia SDR [Silva et al. 2016] através do *framework* Ettus USRP com o apoio da plataforma de desenvolvimento GNU Radio. Como base para a implementação, utilizando a pilha ZigBee desenvolvida por [Schmid et al. 2016]. Como a implementação de ZigBee selecionada se limitava à camada física e ao encapsulamento dos quadros, desenvolvemos novos módulos que implementam o protocolo CSMA/CA do ZigBee. A partir dessa pilha, substituímos o bloco da camada MAC pelo conjunto de blocos que compõem a plataforma FS-MAC, construídos por nós. O código foi desenvolvido parte em C++ e parte em Python.

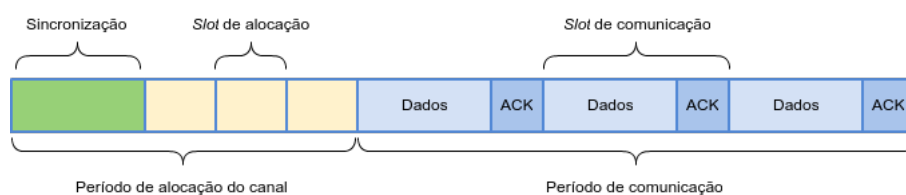
O FS-MAC é uma arquitetura genérica, que permite o sensoriamento de várias métricas bem como protocolos e modelos de decisão diferentes, assim iremos detalhar a seguir as características que implementamos para a avaliação experimental que será apresentada na próxima seção.

### 4.1. Protocolos MAC Implementados

**Protocolo CSMA/CA:** foi implementado um CSMA/CA com back-off exponencial binário e confirmação de envio de mensagens, como ocorre no ZigBee.

**Protocolo TDMA:** O TDMA implementado tem como principais características: a alocação de *slots* de tempo distintos para a comunicação de cada nó e a confirmação de mensagens. Existem dois tipos diferentes de nós, o nó normal, que apenas se comunica com outros nós e o nó coordenador, que além de se comunicar normalmente, também gerencia a distribuição de *slots* de tempo para a troca de mensagens. O nó coordenador é fixo durante todo o tempo de execução, e é determinado por uma variável no GNU Radio.

O tempo é dividido em intervalos chamados *superquadros*, neste intervalo, todos os nós têm a chance de se comunicar com outros nós transmitindo dados e recebendo confirmações. O *superquadro* e suas subdivisões estão representados na Figura 2. O superquadro é formado por dois períodos: o **período de alocação do canal**, em que o nó coordenador envia uma mensagem *broadcast* informando a ordem em que os nós devem requisitar a alocação de um *slot* de comunicação. Esta mensagem também serve como marco de início do superquadro, sincronizando o processo entre todos os nós. O **período de comunicação** define uma quantidade de *slots* igual à quantidade de nós que requisitaram o canal. Usando o mesmo mecanismo da fase de alocação, cada nó espera o momento correto para se comunicar. Quando chega esse momento, ele envia a mensagem que deseja e aguarda a confirmação.



**Figura 2. Formato do superquadro do TDMA implementado.**

## 4.2. Métricas de sensoriamento

Para a fase de Sensoriamento, foram implementadas duas métricas, *Quantidade de transmissores* e *Latência*.

A métrica de quantidade de transmissores é calculada de forma centralizada pelo nó coordenador. Todos os pacotes vindos da camada física passam pelo bloco dessa métrica, que armazena os endereços únicos encontrados na rede. Essa métrica é amostrada a cada 5 segundos, ou seja, ao fim de cada intervalo o valor da métrica é enviado ao módulo de Decisão, o conjunto de endereços únicos é zerado e o processo de armazenamento desses endereços é reiniciado.

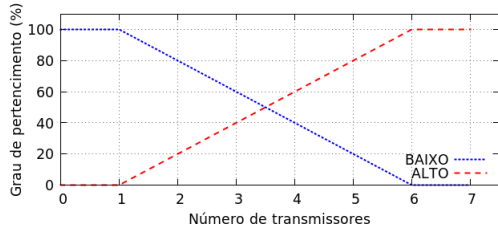
Já os dados da métrica de latência são colhidos de forma distribuída, ou seja, cada nó reúne as informações locais referentes a latência de entrega dos pacotes e as envia para o nó coordenador para que seja gerado o valor da métrica que será enviado ao módulo de Decisão. A latência é calculada como o tempo entre o envio do quadro de dados e a recepção de uma confirmação de recepção, de forma a contabilizar a quantidade de retransmissões necessárias para que o quadro seja recebido com sucesso. Os nós enviam os dados coletados para o coordenador a cada 10 segundos.

## 4.3. Modelo de decisão

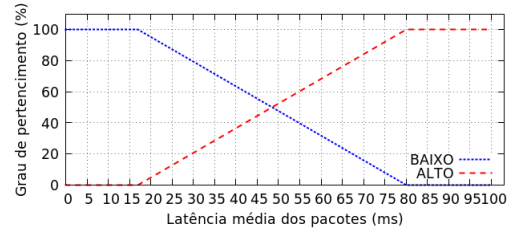
O modelo de decisão construído emprega lógica *fuzzy*, e é modelado como descrevemos a seguir.



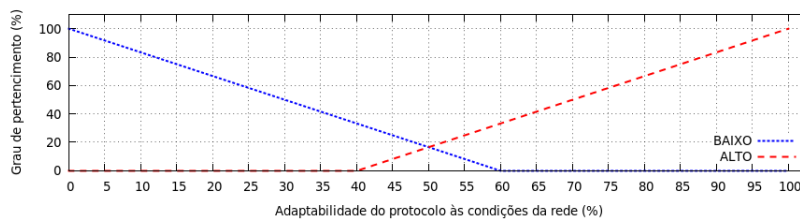
**Variáveis linguísticas:** O modelo considera três variáveis linguísticas, que são (i) Latência média dos pacotes (*LM*), (ii) número de transmissores (*NT*) e (iii) Adaptabilidade do protocolo (*ADP*). Todas elas aceitam os termos nebulosos *BAIXO* e *ALTO*. Foram empregadas as funções de mapeamento apresentadas nas figuras 3, 4 e 5.



**Figura 3. Funções de pertinência para número de transmissores.**



**Figura 4. Funções de pertinência para latência média de entrega.**



**Figura 5. Funções de pertinência para adaptabilidade dos protocolos.**

O motor de inferência, por sua vez, calcula o grau de adaptatividade de cada protocolo. As regras de inferência empregadas foram derivadas a partir da constatação de que técnicas de TDMA são adequadas quando temos muitos transmissores ou quando há um grande grau de contenção. A contenção é medida pela latência média, que captura a quantidade de retransmissões no meio. O motor de inferência utiliza as seguintes regras para calcular a adaptabilidade de cada protocolo:

CSMA

Se NT é BAIXO e LM é ALTO então ADP é ALTO

Se NT é ALTO e LM ALTO então ADP é BAIXO

TDMA

Se NT é BAIXO e LM é ALTO então ADP é BAIXO

Se NT é ALTO e LM é ALTO então ADP é ALTO

Esse conjunto de regras tem foco em redes onde a carga por estação é alta e possui pouca variação. Por consequência, essa implementação é efetiva em redes onde a carga total varia predominantemente em função da quantidade de transmissores.

A partir do cálculo da adaptabilidade de cada protocolo, um dicionário é construído e enviado ao módulo de Troca. Nessa estrutura, as chaves correspondem aos identificadores dos protocolos e os valores correspondem aos graus de adaptabilidade desses protocolos. A defuzzificação das regras é feita empregando o método da centroide.

#### 4.4. Troca

O módulo de troca possui atribuições diferentes dependendo do tipo de nó (normal ou coordenador). O nó normal efetua a troca do protocolo localmente, enquanto o

nó coordenador, além dessa atribuição, coordena a troca do protocolo na rede e avisa periodicamente a todos os nós qual o protocolo está em atividade no momento.

Quando recebe o dicionário com as adaptabilidades vindo do módulo de Decisão, o módulo de Troca executa os passos descritos na seção 3.3. A troca só ocorre caso o grau do protocolo marcado como mais adaptado seja pelo menos 5% superior ao grau de adaptabilidade do protocolo atual. Isto é feito para evitar o efeito ping-pong, que ocorre quando o sistema oscila entre dois protocolos. Inicialmente o módulo de troca do nó coordenador envia uma mensagem do tipo *Envio de informações* ao protocolo ativo para que ele avise aos outros nós a necessidade de troca. Quando o módulo de troca recebe a mensagem do tipo *Informações enviadas*, ele inicia a troca local. A troca local envia uma mensagem do tipo *Parar atuação* ao protocolo ativo. O protocolo interrompe suas atividades e retorna a fila de envios juntamente com uma mensagem do tipo *Atuação parada*. O módulo de troca envia então uma mensagem do tipo *Iniciar atuação* juntamente com a fila de envios ao protocolo que deve entrar em operação. Esse protocolo inicia suas atividades utilizando a fila de envios recebida e responde com uma mensagem do tipo *Atuação iniciada* e a troca é concluída.

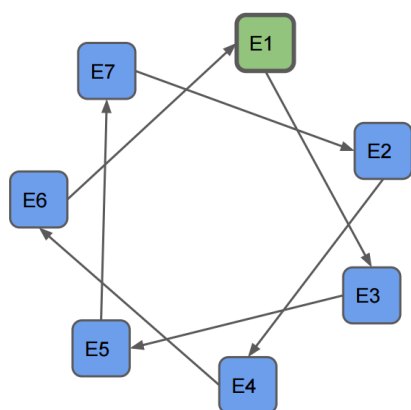
Após a troca, o coordenador informa em *broadcast* a cada 5 segundos qual protocolo está em atividade no momento. Essa mensagem periódica evita que algum nó permaneça muito tempo operando com um protocolo diferente.

## 5. Avaliação

Para avaliar a arquitetura FS-MAC, realizamos um experimento que mede a capacidade da plataforma de se adaptar a diferentes cenários de contenção na rede. Nesse experimento, a carga de dados de cada estação foi mantida constante e variou-se o número de estações transmitindo. Dessa forma, a carga total da rede, e conseqüentemente o seu grau de contenção, variaram principalmente em função do número de transmissores. Utilizamos sete estações, cada uma composta por um computador ligado a uma USRP Ettus B210. Os resultados estão expressos nos gráficos das figuras 8, 9, 10 e 11. Durante os testes, com até quatro estações transmitindo, a plataforma FS-MAC opera com o protocolo CSMA ativo, a partir de cinco transmissores ocorre a troca e o protocolo ativo passa a ser o TDMA.

A Figura 6 mostra o esquema de comunicação organizado durante o experimento. A seta indica a comunicação entre duas estações, a direção da seta indica a direção em que os dados estão sendo enviados e a estação E1 corresponde ao nó coordenador FS-MAC. A Figura 7 mostra como as URPS foram organizadas de acordo com o diagrama apresentado.

O experimento se inicia com um nó transmitindo dados e recebendo confirmação, na segunda etapa mais um nó é acrescentado à rede como transmissor. Nas etapas seguintes, outros nós são acrescentados um a um até um total de sete. Cada etapa dura 180 segundos, dos quais 60 são reservados para a estabilização do nó, instanciação de objetos, ativação dos blocos e formação da fila e durante os outros 120 segundos as métricas de avaliação são coletadas. Com o acréscimo dos nós aumenta-se a quantidade de tentativas de transmissão e conseqüentemente a contenção da rede. Esse processo foi realizado para os protocolos CSMA e TDMA separadamente e para a plataforma FS-MAC contendo os dois simultaneamente. Cada etapa foi repetida cinco vezes e foram coletados os valores



**Figura 6. Diagrama de topologia e comunicação do experimento.**



**Figura 7. Organização das USRPs durante o experimento.**

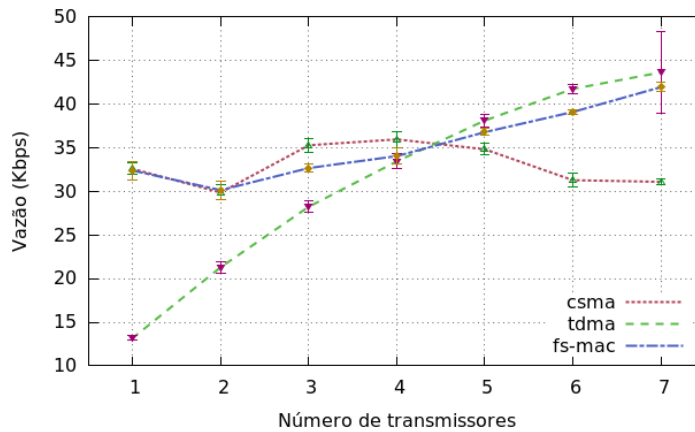
da quantidade de pacotes enviados, quantidade de pacotes confirmados, quantidade de retransmissões e a latência da entrega de cada pacote.

A Figura 8 apresenta a vazão dos protocolos avaliados. Os resultados observados mostram que os protocolos CSMA e TDMA separadamente possuem comportamento compatível com o esperado. Conforme observado em trabalhos como [Takagi and Kleinrock 1985], [Ziouva and Antonakopoulos 2002] e [Hu et al. 2011], o CSMA possui melhor desempenho quando a contenção é baixa, apresenta um ganho durante um intervalo de aumento de contenção, mas tem seu desempenho degradado à medida em que a contenção aumenta. Por outro lado, o TDMA possui baixo desempenho em cenários de baixa contenção, mas melhora à medida em que a contenção aumenta.

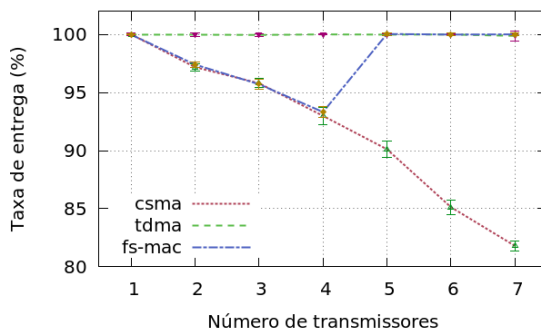
Na mesma figura podemos notar que o FS-MAC tende a seguir o protocolo que possui melhor desempenho em cada cenário. No início do experimento, onde a contenção é baixa, os desempenhos do CSMA e do FS-MAC são praticamente idênticos. Isso pode ser observado também nos gráficos das outras métricas avaliadas, como veremos a seguir. Analisando ainda o gráfico da Figura 8, a partir de três transmissores, o desempenho do FS-MAC é ligeiramente inferior ao desempenho do melhor protocolo. Notamos que com 3 e 4 transmissores a curva da plataforma FS-MAC passa abaixo do CSMA isolado, porém acima do TDMA. Nesses cenários, o FS-MAC está operando com CSMA. A partir do cenário com 5 nós transmitindo, a contenção da rede aumenta e assim o FS-MAC troca o protocolo para o TDMA.

Essa perda de desempenho em relação aos protocolos isolados deve-se principalmente às mensagens de controle do FS-MAC. As mensagens de sensoriamento distribuído possuem maior impacto, pois embora não tenham grande relevância quando há poucos nós transmitindo, aumentam linearmente à medida em que os nós são inseridos na rede. Essa mensagem é enviada por cada nó normal a cada 10 segundos, gerando um grande *overhead* na rede. A mensagem de anúncio de protocolo ativo tem menor impacto, pois embora seja enviada a cada 5 segundos, parte apenas do nó coordenador, independente do tamanho da rede. Apesar do *overhead*, o desempenho do FS-MAC mantém uma distância relativamente pequena do melhor protocolo para cada cenário.

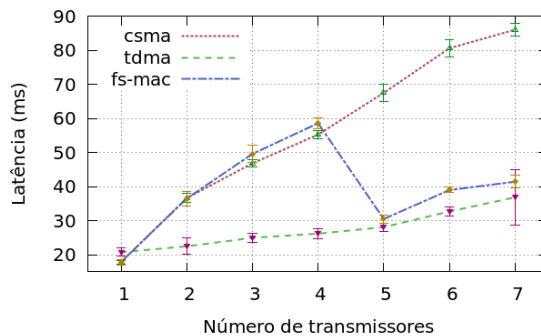
Como esperado, em relação às métricas de taxa de entrega e latência, o desempe-



**Figura 8. Vazão total da rede por número de transmissores.**



**Figura 9. Taxa de entrega média dos pacotes por número de transmissores.**



**Figura 10. Latência média de entrega dos pacotes por número de transmissores.**

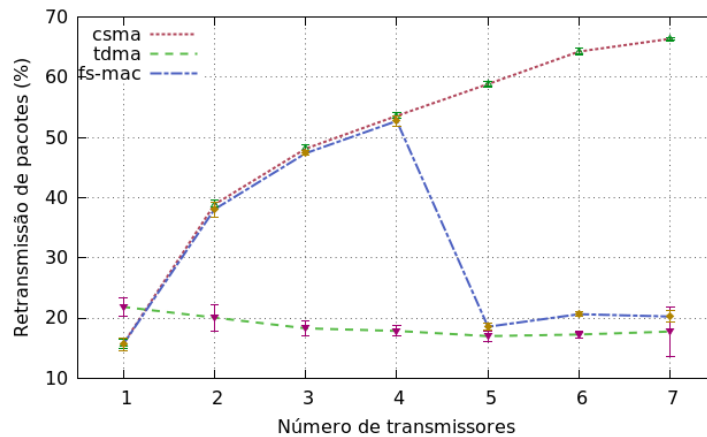
no do FS-MAC segue o do protocolo mais adaptado para o cenário. No caso da latência, mostrada na Figura 10, a perda está associada também ao envio de mensagens de controle. Essas mensagens podem interromper o reenvio de alguma mensagem que esteja na fila e não tenha sido entregue no primeiro envio.

Quanto à retransmissão de pacotes, ao contrário do esperado, o FS-MAC possui menor taxa que o CSMA no início do experimento. Nesse cenário, o número de envios de mensagem de controle do FS-MAC também é menor, assim é compreensível a menor taxa de retransmissão de pacotes mesmo tendo um desempenho ligeiramente inferior.

Em relação ao procedimento de troca dos protocolos, em todas as situações a plataforma detectou corretamente o protocolo a ser utilizado e a troca ocorreu em todos os nós, ou seja, nenhum nó permaneceu operando no protocolo antigo. Acreditamos que esse problema não ocorreu devido à topologia montada, onde todos os nós estavam a curtas distâncias entre si. Nos casos em que esse problema ocorre, o FS-MAC lida com ele fazendo o anúncio periódico de qual protocolo está em operação.

## 6. Conclusões e Trabalhos Futuros

Nesse trabalho apresentamos FS-MAC, uma arquitetura híbrida e com alta flexibilidade para a subcamada MAC de redes sem fio. Essa arquitetura é capaz de detectar,



**Figura 11. Taxa de retransmissão de pacotes por número de transmissores**

através de um conjunto de regras, qual o protocolo mais eficiente para cada cenário apresentado na rede e colocar esse protocolo em atividade. A arquitetura é extensível, permitindo facilmente a adição de novos protocolos e também a alteração das regras de escolha, de acordo com as necessidades da aplicação e dos administradores da rede.

Para a avaliação da arquitetura implementamos um protótipo sobre o padrão IEEE 802.15.4, onde o FS-MAC escolhe entre um protocolo MAC baseado em contenção (CSMA) e outro livre de contenção (TDMA). Os resultados da avaliação mostram o correto funcionamento da arquitetura. Dadas as condições da rede, o protótipo é capaz de perceber qual é o protocolo mais eficiente e colocá-lo em operação corretamente. O comportamento dos protocolos dentro da arquitetura se mostrou muito similar ao seu funcionamento isolado. Devido ao *overhead* causado pelas mensagens de controle que garantem o seu funcionamento, FS-MAC apresenta um desempenho ligeiramente inferior em relação aos protocolos isolados nos cenários em que eles são mais eficientes.

Como trabalhos futuros, pretendemos analisar como incorporar o conceito de SDN no FS-MAC, de forma que um controlador defina regras para a troca de protocolos. Além disso, iremos analisar outros mecanismos para a tomada de decisão, como aprendizado de máquina, que permite que a arquitetura refine as suas escolhas com o tempo. Pretendemos também analisar cenários de redes com mais de um nó desempenhando o papel de coordenador FS-MAC e onde haja entradas e saídas dinâmicas de nós.

## Referências

- Diepstraten, W. and WCND-Utrecht, N. (1993). IEEE 802.11 wireless access method and physical specification. *Power*, 5:10.
- Doerr, C., Neufeld, M., Fifield, J., Weingart, T., Sicker, D. C., and Grunwald, D. (2005). MultiMAC-an adaptive MAC framework for dynamic radio networking. In *First IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN)*, pages 548–555.
- Hu, W., Yousefi'zadeh, H., and Li, X. (2011). Load Adaptive MAC: A Hybrid MAC Protocol for MIMO SDR MANETs. *Wireless Communications, IEEE Transactions on*, 10(11):3924–3933.

- Myers, A. D., Záruba, G. V., and Syrotiuk, V. R. (2002). An adaptive generalized transmission protocol for ad hoc networks. *Mob. Netw. Appl.*, 7(6):493–502.
- Neufeld, M., Fifield, J., Doerr, C., Sheth, A., and Grunwald, D. (2005). SoftMac-flexible wireless research platform. In *Proc. HotNets-IV*, pages 1–5.
- Nychis, G., Hottelier, T., Yang, Z., Seshan, S., and Steenkiste, P. (2009). Enabling MAC Protocol Implementations on Software-defined Radios. In *6th USENIX Symposium on Networked Systems Design and Implementation (NSDI)*, pages 91–105.
- Puschmann, A., Di Francesco, P., Kalil, M. A., DaSilva, L. A., and Mitschele-Thiel, A. (2013). Enhancing the Performance of Random Access MAC Protocols for Low-cost SDRs. In *8th ACM WiNTECH*, pages 9–16.
- Rao, A. and Stoica, I. (2005). An overlay MAC layer for 802.11 networks. In *Proceedings of the 3rd international conference on Mobile systems, applications, and services*, pages 135–148.
- Rhee, I., Warrier, A., Aia, M., Min, J., and Sichitiu, M. L. (2008). Z-MAC: A Hybrid MAC for Wireless Sensor Networks. *IEEE/ACM TON*, 16(3):511–524.
- Ruckebusch, P., Giannoulis, S., Poorter, E. D., Moerman, I., Tinnirello, I., Garlisi, D., Gallo, P., Kaminski, N., DaSilva, L., Gawlowicz, P., Chwalisz, M., and Zubow, A. (2016). A unified radio control architecture for prototyping adaptive wireless protocols. In *European Conference on Networks and Communications (EuCNC)*, pages 58–63.
- Schmid, T., Bloessl, B., and Wunsch, F. (2016). An IEEE 802.15.4 transceiver for GNU Radio v3.7. Disponível em: <https://github.com/bastibl/gr-ieee802-15-4>. Acesso em: 08/05/2016.
- Sharp, B., Grindrod, E., and Camm, D. (1995). Hybrid TDMA/CSMA protocol for self managing packet radio networks. In *Fourth IEEE International Conference on Universal Personal Communications*, pages 929–933.
- Silva, W. S., Cordeiro, J. R. S., Macedo, D. F., Vieira, M. A., Vieira, L. F., and Nogueira, J. M. S. (2016). Introdução a rádios definidos por software com aplicações em GNU radio. *Minicurso SBRC 2016*.
- Takagi, H. and Kleinrock, L. (1985). Throughput analysis for persistent CSMA systems. *IEEE Transactions on Communications*, 33(7):627–638.
- Tinnirello, I., Bianchi, G., Gallo, P., Garlisi, D., Giuliano, F., and Gringoli, F. (2012). Wireless MAC processors: Programming MAC protocols on commodity hardware. In *IEEE INFOCOM*, pages 1269–1277.
- Yang, Z., Zhang, J., Tan, K., Zhang, Q., and Zhang, Y. (2015). Enabling TDMA for today’s wireless LANs. In *2015 IEEE Conference on Computer Communications (INFOCOM)*, pages 1436–1444.
- Ziouva, E. and Antonakopoulos, T. (2002). CSMA/CA performance under high traffic conditions: throughput and delay analysis. *Computer Communications*, 25(3):313 – 321.