

# Um Protocolo de Acesso ao Meio com Network Coding em Ambiente PLC.

Luã M. F. Silveira<sup>1</sup>, Roberto M. Oliveira<sup>2</sup>, Moises V. Ribeiro<sup>2</sup>  
Luiz Filipe M. Vieira<sup>3</sup>, Marcos Augusto M. Vieira<sup>3</sup>, Alex B. Vieira<sup>1</sup>

<sup>1</sup>Departamento de Ciência da Computação, Universidade Federal de Juiz de Fora

<sup>2</sup>Departamento de Engenharia Elétrica, Universidade Federal de Juiz de Fora

<sup>3</sup>Departamento de Ciência da Computação, Universidade Federal de Minas Gerais

lmsilveira@ice.ufjf.br, {rmassi.oliveira,mribeiro,}@engenharia.ufjf.br  
{lfvieira, mmvieira}@dcc.ufmg.br, alex.borges@ufjf.edu.br

**Abstract.** *In this paper, we introduce the use of network coding, in the medium access control layer (MAC), in a power line network environment. We propose a MAC layer protocol and evaluate its performance through simulation of common topology for a PLC system with time division multiple access (TDMA-OFDM). The protocol we propose uses the assistance of a relay, which is responsible for encoding the MPDU's. Our results show that, in a broadcast like transmission, the use of network coding can increase the overall goodput an average of 115% compared to a traditional stop&wait scenario. Moreover, our protocol reduces in 112% the average occupancy of network buffers. Finally, mean latency in a system using the protocol we propose is 4 times lower.*

**Resumo.** *Este artigo apresenta um caso de utilização de network coding, na camada de controle de acesso ao meio (MAC), em ambientes com comunicação via rede elétrica (PLC). Nós propomos e avaliamos o desempenho de um protocolo a partir de simulações de uma topologia comum em um sistema PLC com acesso múltiplo por divisão no tempo (TDMA-OFDM). O protocolo proposto utiliza o auxílio de um relay responsável de codificação dos MPDU's. Nossos resultados apontam que, em uma transmissão tipo broadcast, o uso de network coding pode aumentar o goodput médio em até 115% em relação a um cenário tradicional equivalente, com uso de protocolos estilo stop&wait. Mais ainda, o novo protocolo diminui a ocupação média dos buffers dos dispositivos de rede em até 112%. Finalmente, as latências medianas observadas no sistema com o protocolo proposto são até 4 vezes menores.*

## 1. Introdução

Comunicação via rede elétrica (PLC) atrai cada vez mais a atenção, tanto da comunidade acadêmica, quanto da indústria. A demanda por esse tipo de rede se dá por sua flexibilidade/facilidade na conexão de dispositivos, baixo custo quando comparado a tecnologias sem fio e, até mesmo, por suas altas taxas de transmissão [Ezzine et al. 2014]. Como consequência desse interesse, cada vez mais nos deparamos com redes residenciais e *backbones* de alta velocidade baseadas nesse tipo de rede.

Ainda hoje, técnicas para transmissão de dados utilizadas em PLC podem apresentar altas taxas de falha, dado uma forte atenuação do sinal de comunicação [Gao et al.

2008]. Por esse motivo, há uma grande discussão em torno da utilização de um protocolo de cooperação adequado para melhorar o desempenho da comunicação. A maior parte das contribuições nessa área discute os tipos de melhorias que podem ser obtidas na camada física, utilizando cooperação com técnicas de combinação de sinais provenientes de canais distintos [Biagi 2011, Cheng et al. 2013, Kim et al. 2012, Valencia et al. 2014]. No entanto, há a possibilidade de implementação, tanto de técnicas de cooperação, quanto de *network coding* (NC), em camadas superiores [Noori and Lampe 2013, Bilbao et al. 2014]. Apesar das diversas oportunidades e frentes de trabalho em PLC, ainda há a prevalência de protocolos convencionais, baseados em mestre-escravo ou inundação do canal de comunicação [Gao et al. 2008].

Nesse sentido, dada a natureza de falhas do cenário PLC e os notórios benefícios de *network coding* na melhoria do desempenho de redes, nós propomos um mecanismo de *network coding* na camada de acesso ao meio (MAC) em PLC. Nosso protocolo de comunicação utiliza o auxílio de um nó intermediário (relay), que é responsável pela codificação das unidades de dados (MPDU's). Dado um cenário típico PLC, o protocolo implementado pode melhorar o *goodput* e reduzir a perda de pacotes na rede.

O ganho em vazão e resiliência na rede, a partir do uso de *network coding*, são bem conhecidos para cenários de *broadcast* em redes sem fio [Fragouli et al. 2006]. Entretanto, o uso de *network coding* ainda é pouco explorado em PLC. De fato, há poucos trabalhos na literatura que propõem o uso desse tipo de técnica em PLC [Phulpin et al. 2011, Ezzine et al. 2014]. Como principal consequência, ainda há uma falta de conhecimento sobre o comportamento de tais sistemas na prática. Por exemplo, não são bem delimitadas as circunstâncias nas quais *network coding* na camada MAC poderia melhorar o desempenho de redes PLC. Mesmo que ambos trabalhos citados demonstrem ganhos na vazão ou na resiliência da rede, pouco é tocado no que se refere à sobrecarga imposta à sinalização (ou controle) pelo uso de protocolos com *network coding*.

Para demonstrar tais melhorias, fizemos simulações de um sistema PLC utilizando estimativas mensuradas de taxa de erro de pacote. Nós utilizamos o esquema HS-OFDM (*hermitian symmetric - orthogonally frequency division multiplexing*) [Ribeiro et al. 2014] transmitindo símbolos com a modulação de chaveamento binário de fase (BPSK), por meio de canais domésticos PLC.

Em suma, nossas contribuições são duas: (i) criamos protocolos de comunicação com *network coding* em PLC. (ii) Utilizamos os protocolos desenvolvidos para delimitar os cenários nos quais o uso de *network coding* é viável e oferece melhoras no desempenho de redes PLC. Nossos resultados apontam que, em uma transmissão tipo *broadcast* e transversal, o uso de *network coding* pode aumentar o *goodput* média em até 116% em relação a um cenário tradicional. Mais ainda, a ocupação média dos *buffers* dos dispositivos de rede pode diminuir em até 112%. Finalmente, a latência observada em um cenário que utiliza o protocolo proposto não ultrapassa 20 unidades, enquanto em um ambiente tradicional, menos de 20% dos casos respeitam esse limite.

O restante desse artigo está organizado como segue. Na seção 2, nós avaliamos o estado da arte e motivamos o presente trabalho. Na seção 3, nós descrevemos o cenário e o problema tratado. Na seção 4, nós apresentamos nossa proposta e descrevemos os protocolos desenvolvidos. A seção 5 trás as avaliações realizadas e finalmente, a seção 6

apresenta nossas principais conclusões.

## 2. Trabalhos Relacionados

A Internet cada vez mais experimenta tráfego gerado por aplicações em tempo real. A maioria dessas aplicações, como VoIP e vídeo conferência, são tolerantes ao atraso. É nesse cenário que os enlaces geralmente são super utilizados e, intuitivamente, *network coding* pode trazer seu máximo benefício [Prior et al. 2014]

A principal ideia por traz de *network coding* é deixar que nós intermediários no processo de comunicação misturem pacotes por operações algébricas. Assim, há uma quebra no o esquema tradicional de encaminhamento de mensagens na rede (*store-and-forward*) [Prior et al. 2014]. Por exemplo, em um cenário simples com um *relay* intermediando a comunicação entre dois nós, *network coding* possibilita a transmissão dados entre esses dois nós simultaneamente. Ou seja, o sistema virtualmente transmite as duas mensagens em um único instante de tempo [Katti et al. 2006].

O ganho em vazão e resiliência na rede, a partir do uso de *network coding*, são bem conhecidos para cenários de *broadcast* em redes sem fio [Fragouli et al. 2006]. Nesse contexto, há trabalhos que avaliam as condições de contorno do uso de *network coding* e apresentam os ganhos máximos esperados [Keshavarz-Haddad and Riedi 2014]. Porém, há poucos trabalhos na literatura que propõem o uso desse tipo de técnica em PLC [Phulpin et al. 2011, Ezzine et al. 2014, Parag and Chamberland 2010]. Mesmo que um ambiente sem fio tenha várias características em comum com PLC, os ganhos em um ambiente não podem ser inferidos automaticamente ao outro, e assim, ainda há uma falta de conhecimento sobre o comportamento de NC em PLC na prática [Parag and Chamberland 2010].

Os trabalhos existentes tentam verificar o efeito de *network coding* em PLC, na sua maior parte, em condições ideais. Eles não propõe protocolos na camada de acesso ao meio e tampouco usam cargas realistas em sua simulação. Por exemplo, [Ezzine et al. 2014] foca seu trabalho na proposta de um algoritmo de alocação de subportadoras em um sistema PLC com acesso ao meio baseado em OFDMA (Orthogonal Frequency Division Multiplexing Access). Eles avaliam os ganho de vazão na rede, quando é utilizado um mecanismo de *network coding* simples entre dois nós e um *relay*. Os autores também avaliam o sistema quando os nós estão em diferentes posições em relação ao *relay*. Apesar dos autores terem utilizado exemplos com *network coding*, esse não foi o objetivo do artigo. As avaliações elaboradas são baseadas em ganhos assintóticos, sem a consideração de um protocolo apropriado e erros realistas que podem influenciar no desempenho do sistema.

Phulpin et al. [Phulpin et al. 2011, Prior et al. 2014] também dissertam acerca de prováveis benefícios de *network coding* a uma PLC. Os autores propõem o uso de um esquema de *linear code* em um ambiente de *smart grids*, onde tanto PLC quando redes sem fio podem ser utilizado. O principal objetivo é o aumento da eficiência da coleta de dados na *smartgrid*. Entretanto, os autores também não definem um ambiente e um protocolo para o uso de *coding*. Mais ainda, não há limitações realistas no cenário descrito, como taxas de erro nos canais ou limitações no tamanho dos *buffers*.

Na linha do trabalho que propomos, Bilbao et al. [Bilbao et al. 2014] propõem um protocolo também em camada de acesso ao meio PLC que utiliza *network coding*.

Eles dedicam parte do trabalho à caracterização de erro em PLC de banda estreita e mostram resultados que indicam que *network coding* pode aumentar a vazão da rede em até (aproximadamente) 3x, se comparado a um sistema tradicional com *stop&wait*. Apesar das semelhanças, destacamos que o ambiente experimentado é diferente do proposto nesse trabalho. Acreditamos que nesse trabalho, os autores apresentam um limite superior para os ganhos de *network coding* em PLC. Mais ainda, os autores não avaliam questões importantes, como a ocupação de *buffers* na rede e a latência total para entrega de dados.

### 3. Cenário avaliado

Neste trabalho, consideramos um ambiente típico de PLC, onde há uma sequência de residências (nodos) interconectadas por um canal comandado por um coordenador (nó central). Note que, cada nó pode contemplar diversos dispositivos gerando tráfego na rede. Assim como acontece em meios sem fio, alguns destes nós podem não ter contato com outros na rede, ora por estarem em fases diferente, ora por uma alta atenuação do enlace [Ferreira et al. 2010].

Neste trabalho, consideramos um esquema *uncoded* HS-OFDM juntamente com a modulação de chaveamento de fase binário (BPSK). Consideramos também perfeita sincronização e conhecimento da informação de estado do canal por parte do receptor. A potência total é de  $P = P_0 + P_1$  onde  $P_0$  e  $P_1$  são alocadas para os nós S e R, nessa ordem. As potências são distribuídas igualmente entre as  $N$  subportadoras do símbolo HS-OFDM ( $P_0/N$  e  $P_1/N$  para os nós S e R, respectivamente), durante um ciclo de comunicação de dados (primeiro *time-slot* alocado para o nó S e o segundo, para o nó R).

Nós consideramos um cenário simples, como mostramos na Figura 1, onde a origem de dados se comunica com múltiplos destinos. Cada nó é equipado com um *buffer* de dados onde pacotes de dados são armazenados para posterior transmissão. Mais precisamente, de acordo com a topologia definida na Figura 1, temos cinco nós (A, B, C, D e E). Não há conexão completa entre esses nós, e assim, as linhas nessa figura indicam o compartilhamento do enlace entre os dispositivos PLC, caracterizando um *overhearing* perfeito (A e D podem escutar B e C; e vice-versa). O nó E, geralmente o concentrador na rede, tem capacidade de comunicação com todos os outros.

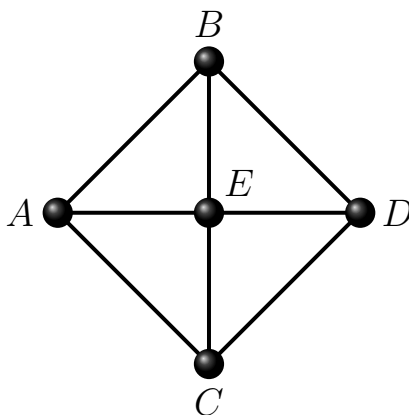


Figura 1. Topologia do cenário estudado.

Cada enlace  $i$ , onde  $i \in \{AB, AC, BD, CD, AE, BE, CE, DE\}$  possui taxas de erro de pacote ( $PER_i$ ) independentes. Note que, apesar de existir apenas um nó *relay*

nesse cenário, a proposta atual pode ser expandida ao uso de múltiplos *relays* de forma transparente ao protocolo proposto na seção 4. Além disso, focamos nesse trabalho em mostrar um cenário simples, onde a aplicação de *network coding* é apropriada.

Os valores das taxas de erro de pacote ( $PER_i$ ) são extraídos a partir da taxa de erro de bits ( $BER_i$ ) obtida na camada física para o  $i$ -ésimo enlace. Seu cálculo é dado por  $PER_i = 1 - (1 - BER_i)^{N_i}$ , onde  $N_i$  corresponde ao tamanho do pacote. Essa fórmula representa a probabilidade de ocorrência de erro em, pelo menos, um bit de um pacote com  $N_i$  bits. Na prática, os valores de  $PER_i$  na camada de enlace são desconhecidos.

Nós estimamos valores para erros a partir de campanhas de medição realizadas em uma área típica urbana brasileira [Roberto de Oliveira 2016], onde foram coletadas informações sobre taxas de erros de bit em mais de 36 mil canais PLC, considerando a frequência de banda entre 1,705 MHz e 100 MHz. Neste trabalho, nós utilizamos dados das medições realizadas em canais com a potência total de 30 dBm, com taxa média de erro de 18% em cada enlace.

#### 4. Protocolo de aplicação do *network coding* XOR em um sistema PLC

Para melhorar o desempenho da rede proposta na figura 1, podemos utilizar uma estratégia de *network coding* XOR [Katti et al. 2006], implementada na camada de controle de acesso ao meio (MAC) de um sistema PLC. Para tal, qualquer um dos nós da topologia poderia ser escolhido para executar o papel de nó codificador, responsável por armazenar em um buffer todo *MAC Protocol Data Unit* (MPDU) recebido, executar uma disjunção exclusiva em todos eles para gerar um MPDU novo que será a combinação de todos outros. O nó codificador gera uma mensagem múltipla para todos os usuários vizinhos, que tem informação suficiente para decodificar o MPDU codificado.

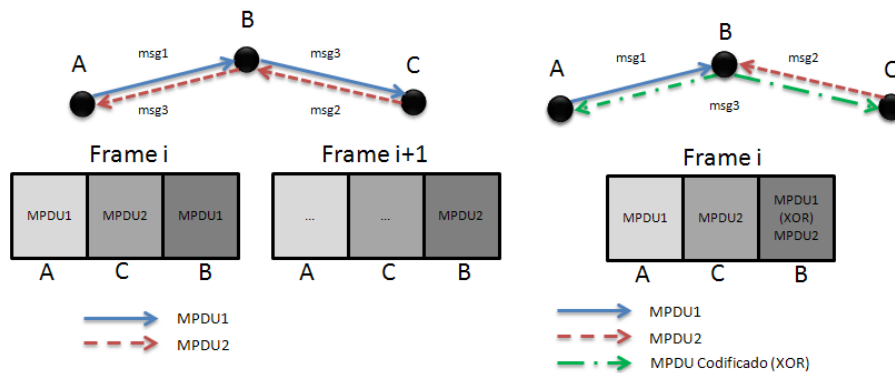
Contudo, espera-se obter melhor desempenho se o nó escolhido for posicionado de maneira central, tal qual o nó E da topologia apresentada na figura 1. O nó E é beneficiado por sua posição relativa aos outros nós da topologia por ter acesso físico a 1-hop de distância de qualquer outro nó. Outra vantagem da escolha de um posicionamento apropriado é que podemos utilizar um escalonamento de *slots* TDMA a fim de favorecer a codificação, estabelecendo o *slot* final de um frame<sup>1</sup> TDMA-OFDM ao nó que executará a combinação (XOR no caso desse trabalho).

Um exemplo de como *network coding* pode propiciar a economia de mensagens em um esquema *multihop* é dado na figura 2.

No exemplo, os MPDU1 e MPDU2 do primeiro quadro são codificados em uma única mensagem (*msg3*), que pode ser enviada ainda no mesmo *frame*, dentro do *time-slot* designado ao nó codificador. Com isso, em um ambiente em que sejam executadas transmissões de fluxo transversal, é possível que uma troca de mensagens sejam executadas em apenas um *frame* TDMA-OFDM, além de proporcionar a economia de uma mensagem para concretizar a comunicação. Dentro da topologia definida no problema, esperamos que a mesma situação ocorra e que os ganhos em termos de economia de mensagem se traduzam em um melhor desempenho geral do sistema.

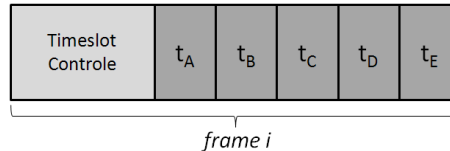
---

<sup>1</sup>Nesse contexto, *frame* é o conjunto de todas as mensagens transmitidas durante todos os *slots* de tempo de uma rodada do TDMA.



**Figura 2. Exemplo de utilização de network coding em TDMA**

Para a implementação do *network coding*, definimos procedimentos de transmissão e recepção, tanto para nós periféricos, quanto para o nó central. O perfil de alocação de *slots* TDMA-OFDM em um *frame*  $i$  pode ser visto na figura 3. Dentro dos *time-slots*  $t_i$ , referente a cada um dos nós  $u$ , ocorre o envio do MPDU do respectivo usuário e o recebimento do mesmo pelos usuários vizinhos ao transmissor. No *slot* de tempo de controle, são enviados ACK e NACK relativos às transmissões no *frame* anterior. Para fins de praticidade na simulação, considera-se que o período de controle é livre de erros.



**Figura 3. Perfil de alocação de um frame TDMA-OFDM**

A estrutura do MPDU pode ser vista na figura 4. Nela, são definidos campos de origem e destino da mensagem, o tipo (dados, controle ou mensagem codificada) e em qual *frame* essa mensagem foi recebida pela camada MAC. O campo "componentes" é um *bitmap* que define quantos e quais usuários estarão presentes em um MPDU codificado. Como utilizamos um esquema TDMA-OFDM de *timeslot* fixo, cada bit do campo representa, em ordem, um *timeslot* do respectivo usuário.



**Figura 4. Estrutura de MPDU**

Os algoritmos que controlam os processos de transmissão e recepção dos nós do sistema simulado são descritos nos fluxogramas das figuras 5 e 6. A figura 5 descreve os procedimentos de transmissão, recepção e decodificação de MPDU's um nó  $u$  em seu respectivo *time-slot*. Quando um erro de MPDU ocorre, o algoritmo compreende também o processo de retransmissão, que está inserido no *time-slot* de controle. Na figura

6, o fluxograma descreve os procedimentos de recepção, transmissão e codificação de MPDU's pelo nó responsável por executar o *network coding* PLC.

Mais precisamente, dentro do algoritmo proposto para o *network coding* PLC, no *time-slot* correspondente, o nó atual executará os seguintes procedimentos:

- Análise de MPDU de controle do último *time-slot*: em um *frame*  $i$ , no *time-slot* de controle, um nó  $u$  recebe um mensagem de ACK/NACK de seus respectivos vizinhos. Se ACK, a mensagem no *frame*  $i - 1$  foi corretamente enviada. Se NACK, a mensagem do *frame*  $i - 1$  é preparada para ser retransmitida no *frame*  $i$ , dentro do *time-slot* relativo ao nó  $u$ .
- Recepção de MPDU de dados: em cada *time-slot*  $t_u$  de um *frame*  $i$ , um nó  $u$  recebe o MSDU dos seus vizinhos. Se o MPDU foi corretamente recebido, ele é armazenado em um buffer de recepção e uma mensagem de ACK é preparada para ser enviada a todos os vizinhos, no *time-slot* de controle do *frame*  $i + 1$ . Se a mensagem tem erro, um NACK é enviado para todos os vizinhos e os MSDU's recebidos no *frame*  $i$  são descartados.
- Recepção de MPDU codificado: em um *frame*  $i$ , no último *time-slot*, todo nó  $u$  recebe uma mensagem codificada do nó responsável pelo *network coding*, contendo as mensagens corretamente recebidas e ele codificadas dentro de um MSDU codificado. Em casos de erro no envio do MPDU codificado, no *time-slot* de controle do *frame*  $i + 1$ , o nó  $u$  envia mensagem de NACK.
- Decodificação de MPDU codificado, caso seja possível: para a decodificação de um MPDU codificado em  $u$ , é necessário que as mensagens dos vizinhos que chegaram sem erro ao nó de codificação, também sejam corretamente recebidas por ele. Se essas mensagens estiverem no *buffer* de recepção de um nó  $u$ , esse nó consegue extrair o MPDU que é destinado a ele. Caso os dados de decodificação não estiverem presentes no *frame* atual, o MPDU codificado é armazenado em *buffer* até que todas as mensagens do *frame* anterior sejam retransmitidas.
- Codificação com disjunção exclusiva: em todo *frame*  $i$ , o último *time-slot*  $t$  é designado ao usuário E, que executa a codificação de todos os MPDU's corretamente transmitidos pelos demais usuários no *frame* e transmite em múltiplas mensagens para todos usuários.
- Transmissão de MPDU gerado pela camada superior: quando um dado é gerada na camada superior de um usuário  $u$ , um MPDU é criado e armazenado na fila de transmissão. Ao ser transmitido, em um *frame*  $i$ , o MPDU fica armazenado nesse *buffer* até a recepção do ACK ou NACK no *time-slot* de controle do *frame*  $i + 1$ .

## 5. Avaliação

### 5.1. Metodologia

Para analisar o desempenho do sistema gerados pelos algoritmos propostos, criamos, no *software* MATLAB, um simulador da topologia definida na figura 1. Nesse sentido, foram testados dois cenários:

- Rede *multihop* com todos os cinco nós iguais. A camada de enlace é responsável por transportar MPDU's utilizando um nó intermediário aleatório entre transmissor e destino, fazendo com que essas MPDU's alcancem usuários que não são acessíveis a *1-hop* pelo nó de origem da transmissão. Todos nós transmitem para

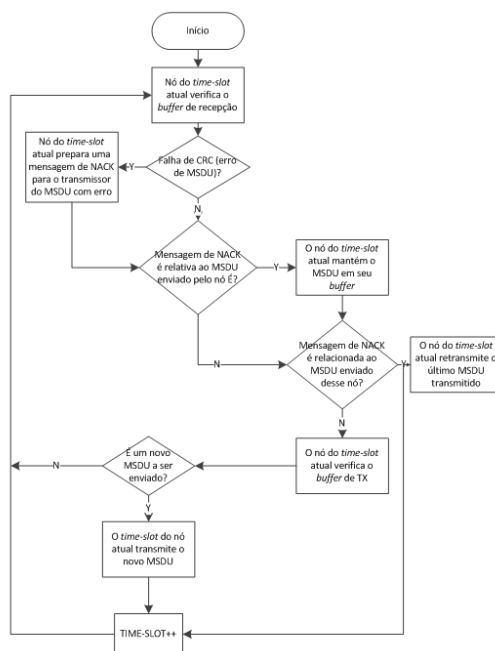


Figura 5. Procedimento de recepção e transmissão com erro no MPDU

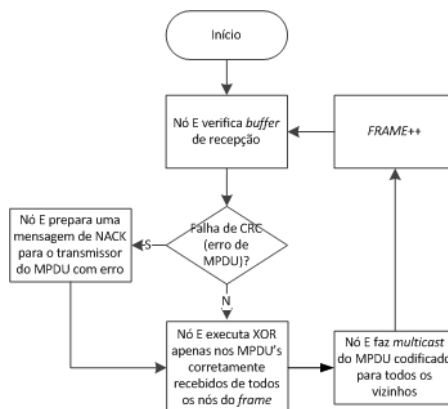


Figura 6. Procedimento de codificação, recepção e transmissão em um nó de Network Coding

todos outros em *multicast*. Assume-se que, para todo nó (exceto o central), há uma probabilidade de geração de uma MPDU de tamanho fixo a ser transmitido em cada *frame* TDMA-OFDM (ou seja, um nó transmite com uma dada probabilidade).

- Rede *multihop* com 4 nós iguais e o nó E (central) responsável pelo *network coding*. O nó central está sempre disponível para receber MPDU's de seus vizinhos, codificá-los e colocá-los em uma fila de transmissão. Assume-se que, para todo nó, há uma probabilidade de geração de uma MPDU de tamanho fixo a ser transmitido em cada *frame* TDMA-OFDM.

Nesses cenários, dois parâmetros foram variados: o tamanho das MPDU's (40, 80, 120, 160, 200 e 248 Bytes) e a probabilidade de um novo MPDU ser gerado pela camada superior e adicionado à fila de transmissão (25, 50 e 75%). Essas variações são interes-



santes pois impactam na taxa de erro de pacotes (i.e. tamanho da MPDU) e o tráfego de dados no sistema simulado (i.e. taxa de transmissão). Os tamanhos utilizados são baseados em estudo empírico que demonstra que mais da metade dos pacotes que trafegam na Internet tem tamanho menor que 300 Bytes [Sinha et al. 2007]. As probabilidades de transmissão apresentam um bom compromisso para demonstrar o comportamento da rede em baixa, média e alta carga de trabalho.

Conforme definimos na seção 3, consideramos também um sistema *uncoded* TDMA-OFDM, que aloca todas as subportadoras para um nó quando o mesmo está usando seu próprio *time-slot* para transmissão. Adotamos a modulação digital BPSK e uma potência  $P = 30$  dBm. A frequência de banda utilizada é de 1,7 até 100 MHz. O tráfego na rede simulada é totalmente transversal (ou seja, A e B enviam pacotes destinados a C e D, e vice-versa).

Em cada cenário avaliado, e em cada uma de suas variações, foram executados os mesmos experimentos, a fim de apurar os seguintes valores: ocupação média dos *buffers*, *goodput* e latência média. A ocupação média de *buffers* é definida pelo total de MPDU's válidas no *buffers* de todos usuários do sistema em um determinado intervalo de tempo. O *goodput* é definido pelo número de recepções corretas por um nó  $u$  em um dado período de tempo. A latência média foi dada pela média do número de *frames* TDMA-OFDM que um MPDU levou para ser entregue corretamente em seu destino, em um determinado intervalo de tempo.

Para medições de ocupação de *buffers*, foram realizadas 1.000 iterações, onde cada uma corresponde a 100 *frames* TDMA-OFDM. Para medições de *goodput*, foram realizadas 1.000 iterações, onde cada uma corresponde à chegada correta de 100 MPDU's a um nó especificado. No caso da medição de latência média, foram realizadas 1.000 iterações, onde cada uma compreende os envios realizados durante a execução de 200 *frames* TDMA-OFDM.

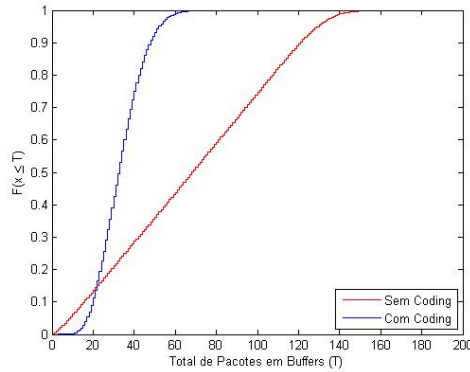
## 5.2. Resultados numéricos

A figura 7 mostra a distribuição acumulada (CDF) da relação entre a ocupação total dos *buffers* em cada *frame* TDMA-OFDM, em um cenário *multihop* comum e o cenário *network coding* PLC. A taxa de geração de MPDU's na camada superior utilizada nesse foi de 50% (rede com carga média de trabalho). Nesse caso, os testes foram executados com MPDU de tamanho com tamanho fixo de 248 Bytes.

Note que, há situações que a rede sem NC (cenário comum) pode apresentar valores de ocupação de *buffer* menores que um cenário com NC. Intuitivamente, quando o sistema tem pouca disputa por recursos, a entrega direta (no caso, por 2 hops) pode gerar menos ocupação dos *buffers* do sistema. Entretanto, para cerca de 90% dos casos, o cenário com NC apresentou melhor desempenho. Observe que em 50% dos casos, uma rede comum tem ocupação de *buffer* média acima da máxima ocupação média encontrada em um cenário com NC. De forma geral, a diminuição média de ocupação de *buffer* pelo uso de NC é alta, chegando a 200% de diferença mediana.

Essa melhoria está relacionada ao fato de, ao adotarmos o protocolo proposto, desobrigamos todos os nós do sistema de exercer um papel de intermediário entre a comunicação de dois nós *multihop*. Com isso, evitamos gargalos que teriam que ser tratados com algoritmos de roteamento mais sofisticados e com maior *overhead* de mensagens

de controle. Como o algoritmo utilizado para o *network coding* é o XOR, o aumento de tamanho do MPDU se dá apenas com alguns campos de controle. Outro fator importante que contribui para o menor uso de *buffers* é o algoritmo utilizado na decodificação dos MPDU's. Quando um nó vizinho não tem os dados necessários para a decodificação de um MPDU codificado, ele descarta os MPDU's do *frame* anterior, conserva apenas mensagem codificada e pede reenvio a todos os vizinhos (menos ao nó de *coding*).



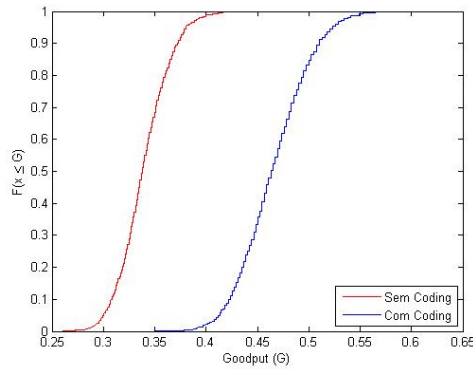
**Figura 7. Ocupação total de *buffers* do sistema ao fim de 100 *frames*.**

Na figura 8, é apresentada a função de distribuição acumulada da relação do *goodput* de todos os usuários do sistema entre os cenários testados. Como esperado, há são necessários menos *frames* TDMA-OFDM para realizar a entrega das MPDU's. Como consequências, a vazão média do sistema se eleva. Mais ainda, há uma redução das filas dos buffers (fig. 7) e assim, menos MPDU's ficam retidos em filas de *buffers* de usuários intermediário. Esses dois fatores contribuem para o ganho observado no gráfico. Na mediana, a diferença é de cerca de 40% entre os dois cenários.

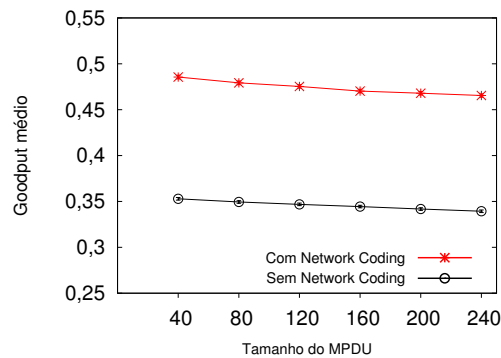
É interessante observar que, utilizando outros algoritmos para regular o processo de ACK e NACK dos pacotes a serem utilizados na decodificação, seria possível ter um ganho ainda maior em *goodput*, em troca de mensagens de controle e uso de *buffers* para armazenar mensagens passadas. Em outras palavras, o nosso algoritmo, de maneira simplista, pede retransmissão de todos os MPDU's em um *frame* que houve erro. Caso fosse implementado mecanismos de retransmissão seletiva, o *goodput* poderia ser ainda melhor em uma rede PLC com *network coding*.

A figura 9 apresenta o *goodput* médio dos nós sistema, em relação a diferentes tamanhos de MPDU. Para os dados apresentados, o intervalo de confiança, para 99% de confiança, é desprezível (menor que 0,01%) Em ambos os cenários, observamos um leve decaimento na métrica de interesse ao se elevar o tamanho das mensagens. Para todas as variações, a diferença observada entre os sistemas é de cerca de 41%. O impacto pelo crescimento do tamanho da MPDU pode estar relacionado a um leve aumento na taxa de erro de pacotes que nesse caso, influencia negativamente ambos os cenários, gerando mais retransmissões.

A variação da taxa de transmissão (probabilidade de se transmitir dados) destaca ainda mais a diferença de desempenho entre os sistemas avaliados. A figura 10 apresenta os valores médios de *goodput* ao se variar essa taxa. Novamente, os valores apresentados são médias e o intervalo de confiança é desprezível (com nível de confiança de 99% é des-

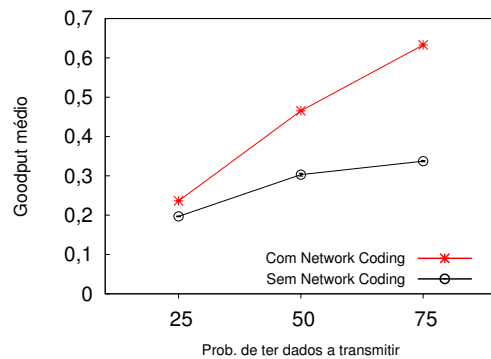


**Figura 8. Goodput total dos nós do sistema ao fim de 100 MPDU's corretamente recebidas.**



**Figura 9. Média do *goodput* do sistema em relação à variação de tamanho de MPDU.**

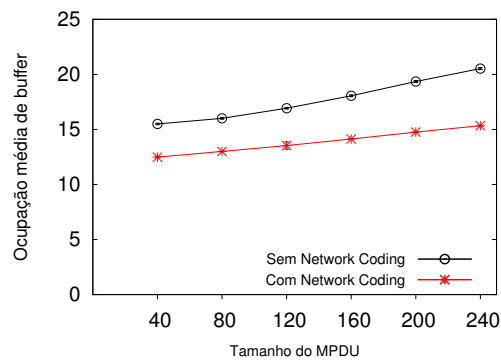
prezível). Por essa figura, observamos que um sistema sem NC satura mais rapidamente. Enquanto o *goodput* ainda é crescente quando há 75% de chances de transmissão com um sistema em NC, a diferença entre os dois últimos pontos para um sistema sem NC é pouco significativa. Com a rede em baixa carga, a diferença entre os 2 sistemas também é baixo. Porém, com 75% de prob. de transmissão em cada nó, a diferença média alcança mais de 115%.



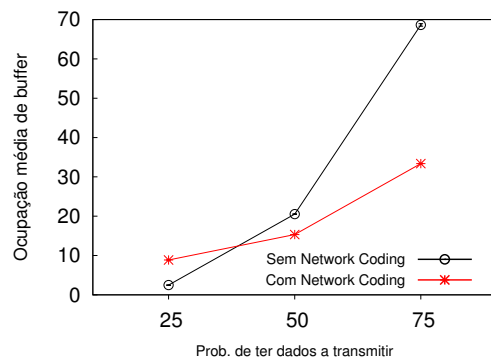
**Figura 10. Média do *goodput* do sistema em relação à probabilidade de geração de um MPDU na camada superior.**

As figuras 11 e 12 apresentam a ocupação de buffer no sistema, ao se variar o ta-

manho das MPDU's e a taxa de transmissão, respectivamente. Assim como para goodput, o aumento do tamanho da MPDU também piora o desempenho de ambos os sistemas. Como a taxa de erros de pacotes é proporcional ao tamanho deles, um aumento na MPDU aumenta a taxa de erro do sistema, o que gera mais retransmissões (e filas nos *buffers* dos dispositivos). Em redes com baixa carga, a transmissão direta, sem necessidade de codificação/decodificação apresenta vantagens em relação ao uso de NC. Esse comportamento afirma o resultado apresentado pela figura 7. Entretanto, com o aumento da taxa de transmissão (i.e. maior probabilidade de transmissões), a rede se torna mais disputada pelos dispositivos. Nesse cenário, o sistema com *network coding* sofre menos impacto que o sistema tradicional. A diferença de desempenho entre os dois sistemas em alta carga chega a ser de aproximadamente 112%.

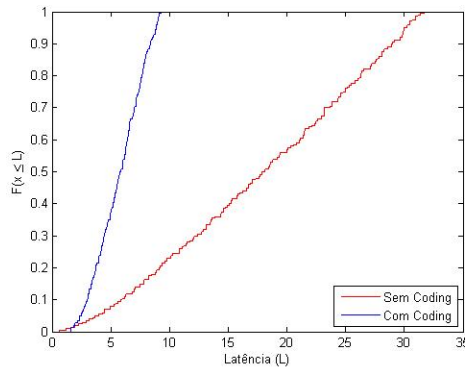


**Figura 11. Média de ocupação de *buffer* no sistema em relação à variação de tamanho de MPDU.**



**Figura 12. Média de ocupação de *buffer* no sistema em relação à probabilidade de geração de um MPDU na camada superior.**

Finalmente, o uso do protocolo proposto com NC proporciona transmissões com menos latência. De fato, virtualmente há um paralelismo nas transmissões, conforme exemplificamos na sessão anterior. De acordo com a figura 13, enquanto a latência máxima experimentada em um sistema com coding foi de apenas 10 unidades (execuções de frame), em sistemas tradicionais, apenas 20% das transmissões conseguiram atingir esse limite. Em mais de 50% dos casos, a latência foi superior a 20 unidades.



**Figura 13. Latência média dos MPDU's em um determinado intervalo de tempo.**

## 6. Conclusões

Nesse artigo, nós propomos um protocolo de comunicação com uso de *network coding* camada MAC para redes PLC que faz uso de um modelo com um único *relay*. Nós propomos o desempenho do protocolo a partir de simulações de uma topologia comum em um sistema PLC com acesso múltiplo por divisão no tempo (TDMA-OFDM).

Nossas simulações mostram ganhos expressivos com a utilização do novo protocolo. Nesse caso, novo protocolo é capaz de reduzir a perda de pacotes e aumentar o *goodput*. De fato, comparado a um sistema tradicional baseado em *stop&wait*, um sistema utilizando o protocolo proposto pode atingir um *goodput* até 116% maior. Mais ainda, a taxa de ocupação de *buffers* no sistema cai pela metade. Finalmente, a latência observada com o uso do novo protocolo, na mediana é 4 vezes menor.

Trabalhos futuros incluem extensões para novos protocolos, especialmente em ambientes híbridos sem-fio/PLC. Mais ainda, pretendemo agregar o uso de *network coding* com cooperação oportunistas nesses sistemas, reduzindo assim a necessidade de um *relay* específico.

## 7. Agradecimentos

Os autores agradecem o apoio do CNPq, CAPES e da FAPEMIG.

## Referências

- Biagi, M. (2011). MIMO self-interference mitigation effects on power line relay networks. *IEEE Communications Letters*, 15(8):866–868.
- Bilbao, J., Calvo, A., Armendariz, I., Crespo, P. M., and Medard, M. (2014). Reliable communications with network coding in narrowband powerline channel. In *Proc. IEEE International Symposium on Power Line Communications and its Applications*, pages 316–321.
- Cheng, X., Cao, R., and Yang, L. (2013). Relay-aided amplify-and-forward powerline communications. *IEEE Trans. on Smart Grid*, 4(1):265–272.
- Ezzine, S., Abdelkefi, F., Cances, J. P., Meghdadi, V., and Bouallegue, A. (2014). Joint network coding and ofdma based mac-layer in plc networks. In *Power Line Commu-*

- nications and its Applications (ISPLC), 2014 18th IEEE International Symposium on, pages 311–315. IEEE.
- Ferreira, H. C., Lampe, L., and Newbury, J. (2010). *Power Line Communications Theory and Applications for Narrowband and Broadband Communications over Power Lines*. John Wiley Sons, United Kingdom, 1st edition.
- Fragouli, C., Le Boudec, J.-Y., and Widmer, J. (2006). Network coding: an instant primer. *ACM SIGCOMM Computer Communication Review*, 36(1):63–68.
- Gao, Q., Yu, J., Chong, P., So, P., and Gunawan, E. (2008). Solutions for the “silent node” problem in an automatic meter reading system using power-line communications. *Power Delivery, IEEE Transactions on*, 23(1):150–156.
- Katti, S., Rahul, H., Hu, W., Katabi, D., Médard, M., and Crowcroft, J. (2006). Xors in the air: practical wireless network coding. In *ACM SIGCOMM Computer Communication Review*, volume 36, pages 243–254. ACM.
- Keshavarz-Haddad, A. and Riedi, R. H. (2014). Bounds on the benefit of network coding for wireless multicast and unicast. *Mobile Computing, IEEE Transactions on*, 13(1):102–115.
- Kim, Y., Choi, S., Kim, S., and Lee, J. (2012). Capacity of ofdm two-hop relaying systems for medium-voltage power-line access networks. *IEEE Trans. on Power Delivery*, 27(2):886–894.
- Noori, M. and Lampe, L. (2013). Improving data rate in relay-aided power line communications using network coding. In *Proc. IEEE Global Communications Conference*, pages 2975–2980.
- Parag, P. and Chamberland, J.-F. (2010). Queueing analysis of a butterfly network for comparing network coding to classical routing. *Information Theory, IEEE Transactions on*, 56(4):1890–1908.
- Phulpin, Y., Barros, J., and Lucani, D. (2011). Network coding in smart grids. In *In. proc. IEEE International Conference on Smart Grid Communications (SmartGridComm)*, pages 49–54.
- Prior, R., Lucani, D. E., Phulpin, Y., Nistor, M., and Barros, J. (2014). Network coding protocols for smart grid communications. *IEEE Transactions on Smart Grid*, 5(3):1523–1531.
- Ribeiro, M. V., Colen, G. R., Campos, F. V. P., Quan, Z., and Poor, H. V. (2014). Clustered-orthogonal frequency-division multiplexing for power line communication: When can it be beneficial? *IET Communications*, 8(13):2336–2347.
- Roberto de Oliveira, Moisés Ribeiro, A. B. V. (2016). Um protocolo mac de cooperação para redes plc. *Elsevier Computer Networks*.
- Sinha, R., Papadopoulos, C., and Heidemann, J. (2007). Internet packet size distributions: Some observations. <https://www.isi.edu/~johnh/PAPERS/Sinha07a.pdf>. Acessado em: 26-12-2015.
- Valencia, J., Oliveira, T. R., and Ribeiro, M. V. (2014). Cooperative power line communication: Analysis of brazilian in-home channels. In *Proc. IEEE International Symposium on Power Line Communications and its Applications*, pages 301–305.