

MICE: Algoritmo de escalonamento para redes sem fio baseado no modelo antiferromagnético de Ising

Nilton Guedes Duarte¹, José Ferreira de Rezende¹

¹Laboratório de Modelagem, Análise e Desenvolvimento de Redes e Sistemas de Computação (LAND)
Programa de Engenharia de Sistemas e Computação
Universidade Federal do Rio de Janeiro (UFRJ)
Rio de Janeiro – RJ – Brazil

{nilton.gduarte, rezende}@land.ufrj.br

Abstract. *Shared medium access algorithms for wireless networks that take into account the link queue size have recently received a lot of attention. In this paper, we propose a distributed algorithm for link scheduling based on Ising's physical model for antiferromagnetism. This model was adapted, by another work, to consider the size of the queues and to use Glauber dynamics to minimize the energy captured by the model. In this work, we propose the inclusion of the external field of the Ising model to avoid that the algorithm get stuck in local minima of the energy function. In addition, an additional algorithm is proposed to transform the result obtained by the model into a viable scheduling. The results demonstrate a good performance in controlling the queues sizes in comparison to the existing algorithms.*

Resumo. *Algoritmos de acesso ao meio compartilhado para redes sem fio que levam em conta o tamanho da fila dos enlaces ganharam destaque recentemente. Neste artigo, propomos um algoritmo distribuído de escalonamento de enlaces baseado no modelo físico de Ising para o antiferromagnetismo. Esse modelo foi adaptado, por outro trabalho, para considerar o tamanho das filas e utilizar a dinâmica de Glauber para minimizar a energia capturada pelo modelo. Neste trabalho, propomos a inclusão do campo externo do modelo de Ising para evitar que o algoritmo fique preso em pontos de mínimos locais da função de energia. Além disso, é proposto um algoritmo adicional para transformar o resultado obtido pelo modelo em um escalonamento viável. Os resultados demonstram um bom desempenho no controle do tamanho das filas em comparação aos algoritmos existentes.*

1. Introdução

O algoritmo para escalonar enlaces em redes sem fio é de muita importância, e um bom algoritmo deve não somente alcançar alta vazão com baixa latência, como deve também ser simples. Pela lei de Little, a vazão e a latência estão relacionadas com o tamanho da fila do enlace e recentemente algoritmos onde a probabilidade de um enlace ser escalonado depende do tamanho de sua fila vem ganhando destaque [Wang and Xia 2017a, Ni et al. 2010, Jiang et al. 2012, Kwak et al. 2016, Xue et al. 2017]. A alta vazão é alcançada quando o algoritmo consegue estabilizar as filas para todas as taxas de chegada dentro da região de capacidade da rede. Foi demonstrado por

[Tassiulas and Ephremides 1992] que o algoritmo de escalonamento de peso máximo (*max-weight scheduling*-MWS) é *throughput-optimal* quando o peso é uma função da fila. No entanto, além de requerer informação global da rede, esse algoritmo precisa resolver um problema combinatório NP-difícil para cada escalonamento, enquanto um algoritmo distribuído e guloso como o IEEE 802.11 não garante vazão máxima.

Uma característica do problema de escalonamento de enlaces é a necessidade do desalinhamento do momento de transmissão dos enlaces vizinhos, ou seja, dois enlaces vizinhos não podem ser escalonados ao mesmo tempo, pois causariam interferência mútua. Esse efeito de desalinhamento é encontrado em partículas com propriedades magnéticas. Um modelo conhecido e estudado sobre o ferromagnetismo é o modelo de Ising [Bissacot and Cioletti 2010], esse modelo probabilístico baseado na dinâmica de Glauber se aproxima da solução do problema combinatório.

Em [Wang and Xia 2017a], os autores adaptam esse modelo para o problema do escalonamento de enlaces. No entanto, esse trabalho apresenta dois problemas. Primeiro, ele apresenta um problema de convergência para a solução de escalonamento adequada. Em certas configurações, o algoritmo não consegue sair de pontos de mínimo da configuração de energia do sistema. O segundo problema é que o conjunto de enlaces E encontrado pelo algoritmo pode ser um subconjunto próprio de um outro escalonamento de enlaces S , em que S também é um escalonamento viável ($E \subsetneq S$). No entanto, uma vantagem desse modelo é o fato dele não se restringir a um subconjunto das configurações de escalonamentos viáveis da cadeia de Markov subjacente, permitindo assim que novos algoritmos sejam explorados.

Neste trabalho, propomos expandir o modelo para adaptar a ação do campo externo do modelo de Ising para o problema de escalonamento. Essa modificação evita que o algoritmo fique preso em pontos de mínimos locais da função de energia. Além disso, propomos um algoritmo para transformar o resultado E obtido pelo modelo em um escalonamento viável maximal, tal que para todo escalonamento viável S , temos que $S \neq E \Rightarrow E \not\subset S$.

O artigo está estruturado da seguinte forma. O modelo de Ising é apresentado na Seção 2.1 para que depois seja adaptado na Seção 2.2 para levar em consideração o tamanho da fila do enlace e de seus vizinhos. Na Seção 3.1, propomos a alteração para adaptar o campo externo atuando sobre o modelo. Na Seção 3.2, apresentamos dois algoritmos baseado em TDMA para encontrar um escalonamento viável a partir de uma configuração qualquer do modelo de Ising e na Seção 3.3 apresentamos e propomos algumas heurísticas para o algoritmo. Na Seção 4, apresentamos os resultados das simulações em vários cenários e concluímos na Seção 5.

2. Trabalhos Relacionados

Nesta seção, conectaremos o modelo físico de Ising com o problema de escalonamento de enlaces em redes sem fio, mas primeiro precisamos conectar a vazão e a latência com o tamanho das filas.

Pela lei de Little, em um sistema em equilíbrio, a taxa de chegada é igual à taxa de saída. A lei de Little se aplica para qualquer sistema, inclusive para sistemas dentro do sistema. Portanto, podemos utilizar a lei de Little para um único enlace, assim como

para todo o conjunto de enlaces do sistema. Seja f o número de pacotes na fila, v a taxa de chegada de pacotes e a o tempo de serviço, por Little $f = va$.

O tempo de serviço de um pacote é o tempo que ele demora para ser enviado pelo enlace, portanto a é o atraso (*delay*). Um bom algoritmo irá escalonar mais vezes os enlaces com maior taxa de chegada de pacotes, diminuindo o tempo de serviço para os pacotes da rede.

2.1. Modelo de Ising

O modelo de Ising [Bissacot and Cioletti 2010] foi desenvolvido pelo físico alemão Ernst Ising para o ferromagnetismo em 1925. O modelo representa os momentos de dipolo magnético de *spins* atômicas, que podem estar em um de dois possíveis estados $\{-1, +1\}$.

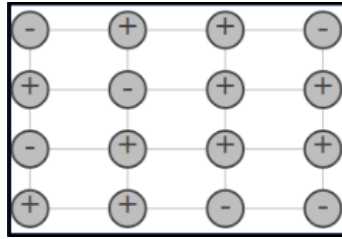


Figura 1. Representação dos *spins* no modelo de Ising

Para estudar o modelo, utilizaremos um grafo $K = (V, E)$ para representar as interações entre os sítios vizinhos. Os nós em V são os *spins* e as arestas em E representam as interações entre os *spins* vizinhos. Para um nó $v \in V$, a variável $\sigma_v \in \{-1, +1\}$ representa o estado do *spin* de v . Esta representação está ilustrada na figura 1.

O espaço de todas as possíveis configurações do sistema é $\Omega = \{-1, +1\}^{|V|}$. Uma configuração $\sigma \in \Omega$ desse modelo é representada por um vetor que associa um *spin* para cada nó $(\sigma_v)_{v \in V} = \sigma$. Para cada par $(i, j) \in E$ de nós vizinhos no grafo, existe uma interação $J_{i,j}$, e para cada nó $v \in V$ existe um campo magnético externo h_j e o momento magnético γ . A energia da configuração σ é dada pela função Hamiltoniana $\mathcal{H}(\sigma)$ (1):

$$\mathcal{H}(\sigma) = - \sum_{(i,j) \in E} J_{i,j} \sigma_i \sigma_j + \gamma \sum_{i \in V} h_i \sigma_i \quad (1)$$

O primeiro somatório de (1) é sobre todos os spins dos pares de nós vizinhos e o segundo sobre todos os spins do modelo. A probabilidade de uma configuração σ qualquer ocorrer no sistema é dada pela medida de Gibbs com o inverso da temperatura $\beta = (k_B T)^{-1} \geq 0$ (2). A constante de normalização $Z(\beta)$ é chamada função de partição.

$$\mu(\sigma) = \frac{e^{-\beta \mathcal{H}(\sigma)}}{Z(\beta)} \quad Z(\beta) = \sum_{\sigma \in \Omega} e^{-\beta \mathcal{H}(\sigma)} \quad (2)$$

Simplificaremos o modelo fazendo com que todas as interações entre os *spins* sejam iguais, $J_{i,j} = J$. Podemos classificar o modelo de acordo com o sinal de J . Para $J > 0$, o modelo é ferromagnético, para $J < 0$ o modelo é antiferromagnético. No modelo ferromagnético, a interação entre os spins vizinhos fazem com que eles desejem se alinhar (ter

o mesmo sinal) e as configurações onde os vizinhos têm o mesmo sinal têm maior probabilidade de ocorrer. No modelo antiferromagnético, os vizinhos desejam se desalinhar (terem sinais opostos) e, conseqüentemente, configurações onde os vizinhos se encontram com sinais opostos têm maior probabilidade de ocorrência.

A temperatura tem papel importante para o modelo ferromagnético ($J > 0$). Quando $\beta = 0$ (temperatura infinita), o modelo se comporta de forma totalmente aleatória, de forma que todas as configurações têm a mesma probabilidade de ocorrer no sistema. Conforme a temperatura do sistema diminui e se aproxima de zero ($\beta \rightarrow \infty$), as probabilidades tendem a levar o sistema para o estado onde todas os *spins* estão alinhados.

A dinâmica de Glauber para o modelo de Ising é uma cadeia de Markov com o espaço de estados Ω e com μ como sua distribuição de estado estacionária. Para determinar a próxima configuração (estado), um nó v é escolhido aleatoriamente e será $\sigma_v = +1$ com probabilidade $q(+1; \sigma, v)$ (4) ou $\sigma_v = -1$ com probabilidade $q(-1; \sigma, v) \triangleq 1 - q(+1; \sigma, v)$, essas configurações serão representadas respectivamente por $\Psi_{\sigma,v}^+$, descrito em (3), e $\Psi_{\sigma,v}^-$. Então temos que $q(+1; \sigma, v)$ é dado por (4).

$$\Psi_{\sigma,v}^+ = \{x_i | x_i = +1 : i = v, x_i = \sigma_i : i \in V \setminus \{v\}\} \quad (3)$$

$$q(+1; \sigma, v) = \frac{\mu(\Psi_{\sigma,v}^+)}{\mu(\Psi_{\sigma,v}^+) + \mu(\Psi_{\sigma,v}^-)} \quad (4)$$

Na dinâmica de Glauber, apenas um nó é atualizado por vez.

O modelo antiferromagnético é interessante para o problema do escalonamento de enlaces, pois é necessário que dois enlaces que se interferem não transmitam simultaneamente, isto é, estejam *desalinhados*. A medida de Gibbs do modelo de Ising para o antiferromagnetismo tem maior probabilidade em configurações de menor energia, em que os vizinhos tendem a estar desalinhados.

2.2. I-CSMA e a adaptação do modelo para filas

O modelo de Ising foi adaptado por [Wang and Xia 2017a] para considerar o tamanho das filas. Em seu artigo, os autores propõem um algoritmo de escalonamento para redes sem fio baseado na versão modificada do modelo de Ising, chamado de I-CSMA.

Segundo os autores, o I-CSMA é uma generalização de algoritmos anteriores *throughput-optimal* baseados na dinâmica de Glauber, como o Q-CSMA [Ni et al. 2010], em que o algoritmo se restringia a uma cadeia de Markov truncada apenas nos estados nos quais não havia interferência entre os enlaces. O artigo demonstra que a restrição não é necessária, a cadeia de Markov pode se mover livremente sobre todo o espaço de configurações e outros métodos de restringir a configuração para o subconjunto de configurações viáveis funcionam igualmente bem para alcançar *throughput-optimality*.¹

Para o algoritmo do I-CSMA, a influência do campo externo do modelo de Ising é eliminada, $\gamma = 0$. A atualização dos *spins* do modelo é realizada em paralelo (dinâmica de Glauber paralela), um subconjunto ξ dos nós do grafo é escolhido aleatoriamente com a restrição de que dois nós vizinhos não podem fazer parte do mesmo conjunto. O conjunto

¹Parágrafo adaptado do artigo original do algoritmo I-CSMA [Wang and Xia 2017a].

é chamado de conjunto de atualização, então cada vértice de ξ é atualizado de acordo com as probabilidades $q(+1; \sigma, v)$ ou $q(-1; \sigma, v)$, independentemente de outros vértices em ξ .

Neste artigo, utilizaremos um grafo de interferência $G(V, E)$, em que V representa o conjunto dos enlaces e E indica a relação de interferência entre os enlaces de V . Os valores $\sigma_v = \{-1, +1\}$ que representavam os valores de *spin* no modelo de Ising, agora representarão o **desejo** de um enlace transmitir. Quando $\sigma_v = +1$, o enlace v deseja transmitir, mas não deseja quando $\sigma_v = -1$.

Dado $\sigma \in \Omega$, é calculado o vetor $F = (F_v)_{v \in V}$ em função da fila de cada enlace v , onde $F_v > 0$ é uma função crescente da fila de v . Por fim, calculamos a função $df_{\sigma, F}(v)$ (5) que depende do desejo e da fila de cada enlace v

$$df_{\sigma, F}(v) = \begin{cases} F_v, & \text{se } \sigma_v = +1 \\ -1, & \text{se } \sigma_v = -1 \end{cases} \quad (5)$$

Definimos então a energia antiferromagnética H da configuração σ sob o vetor F por (6)

$$H(\sigma, F) = \sum_{(v, w) \in E} df_{\sigma, F}(v) df_{\sigma, F}(w) \quad J = -1 \quad (6)$$

As configurações de baixa energia tendem a ter poucos pares onde dois enlaces vizinhos desejam transmitir $(+1, +1)$ e muitos pares onde apenas um vizinho deseja transmitir $(+1, -1)$. Tem a mesma distribuição de probabilidade do modelo de Ising (2), agora β passa a ser interpretado como um parâmetro do sistema e $Z(\beta)$ apenas uma constante normalizadora. Para a probabilidade de atualização de um enlace $q(+1; \sigma, v)$ temos

$$q(+1; \sigma, v) = \frac{\mu(\Psi_{\sigma, v}^+)}{\mu(\Psi_{\sigma, v}^+) + \mu(\Psi_{\sigma, v}^-)} = \frac{e^{-F_v \beta S(\sigma, v)}}{e^{-F_v \beta S(\sigma, v)} + e^{\beta S(\sigma, v)}} = \frac{1}{2} (1 - \tanh(\frac{F_v + 1}{2} \beta S(\sigma, v))) \quad (7)$$

$$S(\sigma, v) \triangleq \sum_{w : (v, w) \in E} df_{\sigma, F}(w) \quad (8)$$

É importante perceber que a probabilidade depende apenas dos vizinhos de v . Essa propriedade faz com que apenas a informação local seja necessária.

O I-CSMA utiliza a função de fila $F_v = 2(d^* - 1) + \log(Q_v(t) + 1)$, em que d^* é o maior grau do grafo e $Q_v(t)$ é o tamanho fila de v no tempo t . A visão geral do algoritmo I-CSMA é a seguinte:

1. O algoritmo funciona com a dinâmica de Glauber paralela, em que os *spins* foram substituídos pelo vetor de fila F , que varia no tempo.
2. Quando um estado da dinâmica de Glauber no tempo t não for um escalonamento viável, o algoritmo converterá para um escalonamento válido.

Para a segunda etapa, seja $\sigma(t) \in \Omega$ um estado da dinâmica de Glauber. O algoritmo converte $\sigma(t)$ para um escalonamento viável $\sigma'(t)$. A conversão é definida por um mapeamento $\psi : \sigma(t) \mapsto \sigma'(t)$. Cada mapeamento ψ é uma nova versão do algoritmo.

A versão a seguir é apresentada pelos autores do I-CSMA. O intervalo de tempo é dividido em três intervalos: dois intervalos de controle e um intervalo de transmissão. Os

intervalos de controle têm duração constante W e W' mini-intervalos. Na primeira fase de controle, será escolhido coletivamente o conjunto $\xi(t)$, o conjunto dos nós independentes que podem ser atualizados segundo a dinâmica de Glauber em paralelo, para definir o vetor $\sigma(t)$. Na segunda fase de controle, será encontrado o resultado $\sigma'(t)$ do mapeamento de ψ . Por fim, os nós em $\sigma'(t)$ transmitem.

I-CSMA - Algoritmo de escalonamento no enlace v

Inicialização:

1. No início do intervalo de tempo, o enlace v calcula $S(\sigma(t-1), v)$ baseado no estado dos vizinhos e do tamanho da fila que aprendeu durante o intervalo anterior. O enlace v calcula a probabilidade $q(+1; \sigma(t-1), v)$ de acordo com (7).

Fase de Controle 1 - W Mini-Intervalos: Estabelecer $\sigma_v(t)$

2. Enlace v seleciona um tempo aleatório T_1 distribuído uniformemente em $\{0, 1, \dots, W-1\}$ e inicia um temporizador de T_1 mini-intervalos de controle.
3. Se o enlace v receber uma mensagem de *INTENT* de qualquer um de seus enlaces vizinhos antes do temporizador T_1 expirar, ele define $\sigma_v(t) = \sigma_v(t-1)$ e não vai transmitir uma mensagem de *INTENT* (v não será incluído em $\xi(t)$).
4. Caso contrário, quando T_1 expirar, o enlace v envia em *broadcast* uma mensagem de *INTENT* no início do mini-intervalo ($T_1 + 1$).
 - Se a mensagem de *INTENT* de v colidiu, o enlace v define $\sigma_v(t) = \sigma_v(t-1)$ (v não será incluído em $\xi(t)$).
 - Caso contrário, o enlace v define $\sigma_v(t) = +1$ com probabilidade $q(+1; \sigma, v)$, ou $\sigma_v(t) = -1$ com probabilidade $q(-1; \sigma, v)$

Fase de Controle 2 - W' Mini-Intervalos: Estabelecer $\sigma'_v(t)$

5. Enlace v define $\sigma'_v(t) = 0$. Se $\sigma_v(t) = 1$, v executa o seguinte
 - Enlace v seleciona um tempo aleatório T_2 distribuído uniformemente em $\{0, \dots, W'-1\}$ e inicia um temporizador de T_2 mini-intervalos de controle.
 - Quando o temporizador T_2 expira, v envia uma mensagem de *RESERVE* em *broadcast* com seu tamanho de fila atual
 - Se o enlace v não recebeu nenhuma mensagem de *RESERVE* de seus enlaces vizinhos antes do fim do temporizador T_2 e sua mensagem de *RESERVE* não colidiu, o enlace v define $\sigma'_v(t) = 1$

Intervalo de Dado:

6. Se $\sigma'_v(t) = 1$, o enlace v transmite um pacote

As informações para calcular $S(\sigma(t-1))$ são obtidas através da mensagem de *RESERVE* do intervalo anterior.

3. Proposta

O trabalho de [Wang and Xia 2017a] de adaptação do modelo de Ising para o I-CSMA sem restrição dos estados pelos quais a cadeia de Markov pode se mover traz maior liberdade. Neste trabalho, propomos expandir o modelo para considerar a ação do campo externo do modelo de Ising para o problema de escalonamento.

A proposta é motivada pelo problema encontrado com o modelo utilizado pelo I-CSMA em que apenas a interação dos vizinhos é utilizada. Quando apenas essa interação é considerada, a função de probabilidade tende a prender a cadeia em algum estado no qual a energia do sistema é um ponto de mínimo local e a cadeia de Markov tem dificuldade de mover. Para o caso físico é esperado, pois o sistema encontrou um ponto onde sair significaria aumentar a energia do sistema, mas para o problema de escalonamento de enlaces isso significa que apenas alguns enlaces serão escalonados.

Por exemplo, para dois enlaces v_1 e v_2 se interferindo, com $F_{v_1} = df_{\sigma,F}(v_1) = 7$, $df_{\sigma,F}(v_2) = -1$, $F_{v_2} = 9$ e $\beta = 1$, temos que $q(+1; \sigma, v_1) = 0.999$ e $q(+1; \sigma, v_2) \approx 0$, o sistema dificilmente sairá desse estado, e quanto maior o tamanho da fila de v_2 , menor a probabilidade de v_2 mudar de estado.

Uma forma de resolver esse problema seria diminuir o valor de β para que o sistema se comporte de forma mais aleatória e, com isso, não se prenda aos mínimos locais da função de energia. Porém, o que nos interessa é justamente os pontos de mínimo local da função de energia.

3.1. MICE: Modelo de Ising com Campo Externo

Dado o problema mencionado acima, propomos a utilização do campo externo do modelo de Ising ($h_j, \gamma > 0$) de modo que o crescimento da fila de um enlace interagindo com o campo externo leve o enlace para o estado $\sigma_v = +1$, perturbando o sistema e fazendo com que se mova para outro estado de mínimo local.

Para provocar esse comportamento, a interação do campo externo será proporcional ao tamanho da fila, $h_j = \alpha df_{\sigma,F}(v)$, $\alpha > 0$ e $\gamma > 0$. Assim, o somatório referente ao campo externo será $\gamma \alpha \sum_{i \in V} df_{\sigma,F}(v)^2$. Para simplificar podemos incorporar α ao γ , $\gamma = \gamma \alpha$. Teremos, então,

$$H(\sigma, F) = \sum_{(v,w) \in E} df_{\sigma,F}(v) df_{\sigma,F}(w) - \gamma \sum_{v \in V} df_{\sigma,F}(v)^2 \quad (9)$$

A distribuição de probabilidade é a mesma do modelo de Ising, agora considerando o campo externo γ .

$$\mu(\sigma) = \frac{e^{-\beta H_\gamma(\sigma, F)}}{Z(\beta, \gamma)} \quad Z(\beta, \gamma) = \sum_{\sigma \in \Omega} e^{-\beta H_\gamma(\sigma, F)} \quad (10)$$

Precisamos agora encontrar as novas equações para $\mu(\Psi_{\sigma,v}^+)$ (11) e $q(1; \sigma, v)$ (12).

$$\begin{aligned} Z(\beta, \gamma) \mu(\Psi_{\sigma,v}^+) &= \exp(-\beta(F_v \sum_{w:(w,v) \in E} df(w) - \gamma F_v^2 \\ &\quad + \sum_{(k,w) \in E; k,w \neq v} df(k) df(w) - \gamma \sum_{w \in V \setminus \{v\}} df(w)^2)) \\ \mu(\Psi_{\sigma,v}^+) &= \frac{e^{-F_v \beta S(v)} e^{+\beta \gamma F_v^2} e^{-\beta H(\sigma \setminus \{v\})}}{Z(\beta, \gamma)} \end{aligned} \quad (11)$$

$$\begin{aligned} q(1; \sigma, v) &= \frac{\mu(\Psi_{\sigma,v}^+)}{\mu(\Psi_{\sigma,v}^+) + \mu(\Psi_{\sigma,v}^-)} = \frac{e^{F_v \beta (F_v \gamma - S(\sigma, v))}}{e^{F_v \beta (F_v \gamma - S(\sigma, v))} + e^{\beta (S(\sigma, v) + \gamma)}} \\ &= \frac{1}{1 + e^{\beta((\gamma + S(\sigma, v)) - F_v (F_v \gamma - S(\sigma, v)))}} \end{aligned} \quad (12)$$

Esta formulação foi chamada de Modelo de Ising com Campo Externo (MICE).

Voltando ao exemplo de dois enlaces v_1 e v_2 se interferindo, utilizando (12) com $F_{v_1} = df_{\sigma,F}(v_1) = 5$, $df_{\sigma,F}(v_2) = -1$, $F_{v_2} = 7$, $\beta = 1$ e $\gamma = 1$, temos que $q(+1; \sigma, v_1) \approx 1$ e $q(+1; \sigma, v_2) \approx 1$. Na próxima fase de controle do algoritmo teremos $q(+1; \sigma, v_1) \approx 0$ e $q(+1; \sigma, v_2) \approx 1$, possibilitando assim a transição do sistema para outro ponto de mínimo.

3.2. Algoritmos de escalonamento

Um problema do algoritmo I-CSMA é o número de colisões durante as fases de controle, que pode ser muito grande dependendo da topologia da rede e de β , deteriorando assim o desempenho do algoritmo. Para minimizar esse problema, desenvolvemos dois algoritmos baseados em TDMA (sem colisões) para a segunda parte do problema, ou seja, o mapeamento de σ para um escalonamento viável. Os algoritmos foram baseados nas seguintes ideias: Cada nó deve transmitir apenas uma mensagem durante a fase de controle, apenas informação local pode ser utilizada, pode demorar várias fases de controle para terminar, se interrompido deve entregar um resultado parcial viável e se o algoritmo terminar, o resultado deve ser maximal. O resultado E é dito maximal quando não existe outro escalonamento viável S que contenha todos os enlaces de E e seja diferente de E .

Podemos desenvolver algoritmos simples para mapear σ em um escalonamento viável. O primeiro algoritmo é chamado de Escalonador Maximal (EsMa):

Escalonador Maximal - EsMa

Inicialização (Entrada inteiro k):

1. Cada nó v escolhe um número aleatório em $[0, M]$, anuncia para seus vizinhos e se inicia no estado **DISPUTA**

Atualização de cada nó v em estado **DISPUTA**

2. Se o número escolhido por v é o maior entre todos os nós em estado **DISPUTA** da rede
 - Estado de v passa para **GANHADOR**
 - Estado dos vizinhos de v passa para **PERDEDOR**

Finalização

3. Executar k -vezes o passo de atualização (passo 2)
4. Todos os nós em estado **GANHADOR** transmitem

O funcionamento é explicado utilizando o conhecimento global da rede para facilitar o entendimento, mas na prática é necessário apenas conhecimento local. Quando um nó é o maior entre os seus vizinhos, ele é, ou eventualmente será, o maior da rede, pois nenhum dos seus vizinhos poderá passá-lo para o estado **PERDEDOR**. Assim, pode então tomar a decisão com a informação local dos vizinhos e passar para o estado **GANHADOR** e seus vizinhos para **PERDEDOR**.

Para fazer a conexão com o MICE, cada nó v irá escolher um valor em $[0, N]$, caso $df_{\sigma,F}(v) = -1$, e um valor em $[N, M]$, $N < M$ caso contrário.

O segundo algoritmo busca resolver o problema de atribuição de intervalo, definido em [Rhee et al. 2009] como o problema de encontrar um intervalo de tempo para cada nó ser escalonado sem que nenhum de seus vizinhos sejam escalonados no mesmo intervalo. O algoritmo é chamado de Escalonador Maximal para Atribuição de Intervalo (EsMAI), baseado no algoritmo anterior e no algoritmo proposto por [Rhee et al. 2009].

Escalonador Maximal para Atribuição de Intervalo - EsMAI

Inicialização (Entrada inteiro k , inteiro c):

1. Cada nó v escolhe um número aleatório em $[0, M]$, anuncia para seus vizinhos e se inicia no estado **DISPUTA**

Atualização de cada nó v em estado **DISPUTA**

2. Se o número escolhido por v é o maior entre todos os nós em estado **DISPUTA**
 - Estado de v passa para **GANHADOR**
 - Estado dos vizinhos de v passa para **PERDEDOR**

Escalonamento parcial

3. Executar k -vezes o passo de atualização (passo 2)
4. Todos os nós em estado **GANHADOR** transmitem

Finalização - Atribuição de Intervalo

5. Todos os nós voltam ao estado **DISPUTA**
 - Nós que já foram escalonados escolhem um número em $[0, N]$, $N < M$
 - Nós que ainda não foram escalonados escolhem um número em $[N, M]$
 - Executar o passo de atualização (passo 2)
6. Executar c -vezes o passo de atribuição de intervalo (passo 5)

Assim como no algoritmo EsMa, apesar da explicação utilizar conhecimento global, a implementação precisa de conhecimento apenas local. A conexão com o MICE é feita como no algoritmo anterior, com um adicional de que todo nó v com $df_{\sigma,F}(v) = -1$ começa o algoritmo como se já tivesse sido escalonado. Para acelerar o término do algoritmo, a condição de ser o maior entre os vizinhos é relaxada e é necessário que seja pelo menos o segundo maior entre os vizinhos, com a condição de que um nó no intervalo $[0, N]$ não pode nunca ganhar de um nó no intervalo $[N, M]$.

Para que em cada intervalo de tempo tenhamos um escalonamento, é necessário executar k instâncias do algoritmo em paralelo, assim em cada intervalo de tempo uma das instâncias terminou o algoritmo. Desta forma o tempo gasto no controle do algoritmo é menor em troca de adicionar um atraso entre o momento em que a configuração do modelo é obtida e em que é realizado o escalonamento relacionado à configuração.

3.3. Heurísticas para o modelo

O modelo de Ising utiliza a dinâmica de Glauber sobre a cadeia dos estados, o que traz mais complexidade ao algoritmo. Duas heurísticas são propostas por [Wang and Xia 2017a]: uma permite que todos os nós atualizem seus estados ao

mesmo tempo, sem a necessidade de se formar um conjunto ξ independente, a outra utiliza a função de fila $F_v^H = \log(Q_v(t) + 1)$.

Outra heurística proposta e avaliada neste trabalho é relacionada à função $S(\sigma, v)$, a soma sobre todos os vizinhos. A função pode ter uma variação muito grande de valor dependendo da quantidade de vizinhos do nó, o que faz com que a interação de S com γ seja dependente do número de vizinhos. Para minimizar esse problema relacionado à topologia do grafo, a função $S(\sigma, v)$ passa a ser a *média* entre todos os vizinhos: $S^H(\sigma, v) = \frac{S(\sigma, v)}{d_v}$, onde d_v é o número de vizinhos de v .

Com isso, podemos nomear os algoritmos de acordo com suas fases e heurísticas: com a formulação do modelo de fila (MICE, I-CSMA ou nenhum), com a utilização do conjunto ξ da dinâmica de Glauber em paralelo (GD) ou não e com o mapeamento para o escalonamento viável (ICSMA, EsMa e EsMAI). Temos, então, os seguintes algoritmos:

Tabela 1. Algoritmos

| Algoritmo | Modelo de fila | GD | Mapeamento |
|---------------|----------------|-----|------------|
| I-CSMA | I-CSMA | Sim | ICSMA |
| MICE-ICSMA | MICE | Sim | ICSMA |
| MICE-EsMa | MICE | Não | EsMa |
| MICE-EsMAI | MICE | Não | EsMAI |
| MICE-GD-EsMa | MICE | Sim | EsMa |
| MICE-GD-EsMAI | MICE | Sim | EsMAI |
| EsMa-S/F | Nenhum | Não | EsMa |

Os algoritmos EsMa e EsMAI utilizam as funções S^H e F_v^H . Para os algoritmos MICE-GD-EsMa e MICE-GD-EsMAI, o conjunto ξ do nós independentes é encontrado utilizando o algoritmo EsMa sem levar em consideração o tamanho das filas.

4. Simulações e Resultados

Nesta seção, apresentamos os resultados de desempenho, obtidos via simulação, do algoritmo I-CSMA e dos algoritmos propostos. Simulamos 90 grafos de conflito com 16 nós posicionados aleatoriamente, com cada nó do grafo tendo obrigatoriamente pelo menos um vizinho. A rede é de salto-único.

Cada simulação foi executada 5 vezes por 10^5 intervalos de tempo em cada grafo, totalizando 450 simulações. Os resultados coletados foram as médias das filas dos 16 nós (enlaces) do grafo, todos os resultados estão com intervalo de confiança de 90%. Para os algoritmos EsMa e EsMAI utilizamos $k = 4$ e $c = 4$.

4.1. Geração de tráfego

Os tráfegos da rede foram gerados através de uma distribuição de Pareto limitada. A função de distribuição cumulativa é dada por $F(x) = \frac{1-(L/x)^\alpha}{1-(L/H)^\alpha}$. Como sugerido em [Wang and Xia 2017b], utilizamos o fator $\alpha = 1.5$, o limite superior $H = 1000$, já o limite inferior L depende da média da taxa de chegada de cada grafo.

O vetor de chegada λ_G de cada grafo é calculado através dos seus respectivos conjuntos de escalonamentos maximais. Seja $M_G = \{E_1, E_2, \dots, E_n\}$ o conjunto de

todos os E_i escalonamentos maximais do grafo G , definimos o vetor 16-dimensional s_i para representar o escalonamento E_i . Dada uma ordenação dos nós de G , se o nó $v \in E_i$ então $s_{i,v} = 1$, caso contrário, $s_{i,v} = 0$. Por exemplo, para o escalonamento $E_k = \{1, 4, 7, 11, 16\}$, temos $s_k = (1, 0, 0, 1, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 1)$. A região de capacidade do sistema é definida pela combinação convexa do conjunto M_G [Tassiulas and Ephremides 1992]:

$$\lambda_G = \rho \sum_i w_i s_i \quad \text{restrito a} \quad \sum_i w_i = 1 \quad \text{e} \quad w_i \geq 0, \forall i \quad (13)$$

Sendo ρ a carga do sistema e w_i o peso dado ao escalonamento E_i , quando $w_1 = w_2 = \dots = w_n = 1/n$, todos os escalonamentos E_i têm a mesma importância. Para este caso, consideramos o tráfego como sendo uniforme. Quando $0 < \rho < 1$, temos que λ_G está dentro da região de capacidade do sistema.

Encontramos o valor de L para fazer com que a média da distribuição de Pareto seja igual à taxa de chegada do sistema. Para tráfegos não uniformes, foram utilizadas distribuições de pesos seguindo a progressão geométrica (14).

$$w_i = \frac{r^i}{\sum_{j=0}^n r^j} \quad (14)$$

4.2. Resultados

Comparamos os algoritmos propostos e o algoritmo I-CSMA. Os valores utilizados foram $\beta \in \{0.01, 0.1, 1, 1.5\}$ e $\gamma \in \{0.5, 1.0, 1.5, 2.0, 2.5\}$, os tráfegos simulados foram o tráfego uniforme e tráfegos não uniformes (14) com $r \in \{0.2, 0.6\}$. Os valores da carga do sistema são $\rho \in \{0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0\}$, mas para algumas simulações, apenas os resultados com maior carga são apresentados.

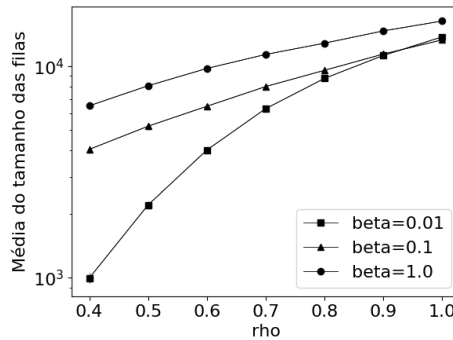


Figura 2. Algoritmo I-CSMA para diferentes valores de β

Para o caso do tráfego uniforme com o I-CSMA (Figura 2), obtivemos resultados melhores quando diminuimos o valor de β . Já para o MICE-ICSMA (Figura 3), os diferentes valores de β têm resultados similares, enquanto aumentar o valor de γ melhora o desempenho do algoritmo. Tornar os algoritmos mais aleatórios melhorou suas performances, isso se deve à característica uniforme do tráfego gerado na rede, fazendo com que seja mais importante que o escalonamento seja maximal do que levar em consideração o tamanho das filas. Por exemplo, quando aumentamos o valor de γ , estamos aumentando

a probabilidade de um enlace ir para o estado $\sigma(t) = 1$ e quando todos os enlaces estão com $\sigma(t) = 1$, o mapeamento do I-CSMA será maximal caso não ocorra colisões, porém todos os enlaces em $\sigma(t) = 1$, aumenta a chance de colisões.

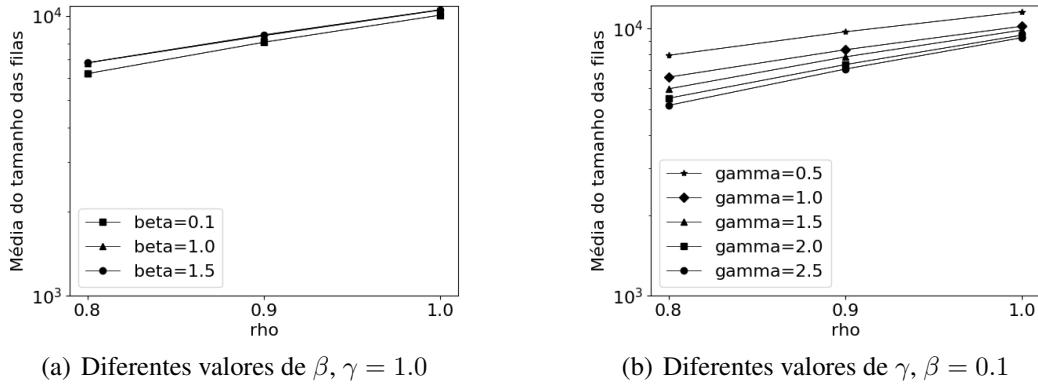


Figura 3. Algoritmo MICE-ICSMA para diferentes valores de β e γ .

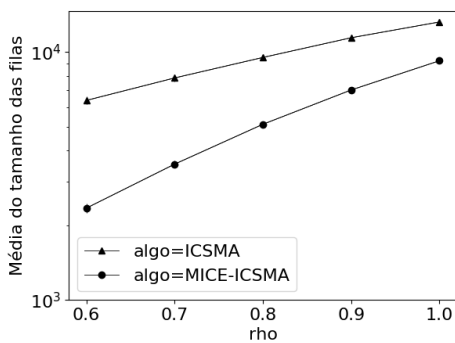


Figura 4. Comparação entre MICE-ICSMA e I-CSMA

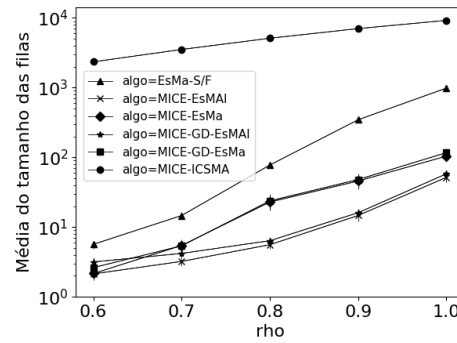
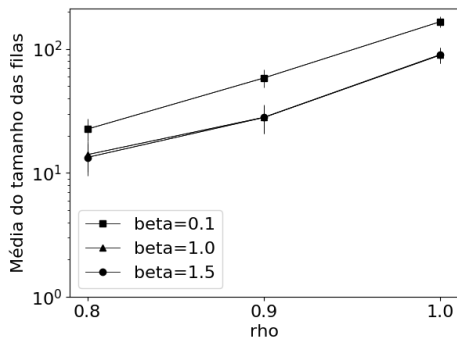


Figura 5. Comparação entre os algoritmos propostos

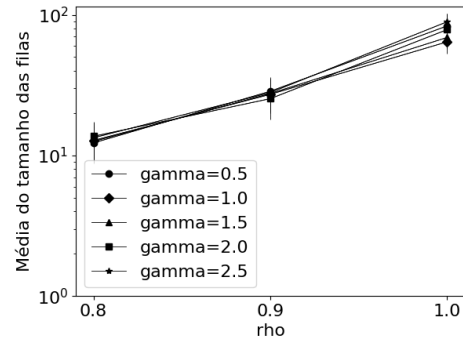
Na Figura 4, os algoritmos foram simulados utilizando tráfego uniforme e os melhores valores encontrados para β e γ de cada algoritmo. A formulação proposta tem ganho de 63% para $\rho = 0.6$ e 30% para $\rho = 1.0$.

Para comparação, também foi simulado o algoritmo EsMa sem levar em consideração o tamanho da fila, EsMa-S/F. A Figura 5 apresenta o resultado de todos os algoritmos propostos neste artigo com os melhores valores de β e γ encontrados mais o algoritmo EsMa-S/F para referência, todos com o tráfego uniforme. A diferença entre os algoritmos que utilizam o conjunto ξ e os que não utilizam é desprezível. O algoritmo MICE-ICSMA tem a média das filas mais de 400 vezes maior que o algoritmo EsMa-S/F para $\rho = 0.6$, demonstrando a importância do resultado do mapeamento ser maximal e do algoritmo sem colisões. Os algoritmos MICE-EsMa e MICE-EsMAI, que consideram a fila para a tomada de decisão, têm performance 86% e 95% melhores, respectivamente, em relação ao algoritmo EsMa-S/F quando $\rho = 0.9$. Fica evidente a importância de considerar o tamanho das filas na tomada de decisão do algoritmo quando o resultado é garantidamente maximal e não há colisões.

Na Figura 6 com o tráfego uniforme, quando aumentamos $\beta = 0.1$ para $\beta = 1.0$ a performance do algoritmo é melhor (51% para $\rho = 0.9$), isto é, diminuir a aleatoriedade



(a) Diferentes valores de β , $\gamma = 2.5$



(b) Diferentes valores de γ , $\beta = 1.5$

Figura 6. Algoritmo MICE-EsMa para diferentes valores de β e γ .

do sistema trouxe resultados melhores. Já os diferentes valores de γ não apresentam mudança significativa para os valores simulados.

Também realizamos simulações com tráfegos não uniformes, na Figura 7 é apresentado o resultado de todos os algoritmos para $\beta = 1.0$ e $\gamma = 2.5$ quando $r = 0.2$. A diferença entre o MICE-ICSMA e EsMa-S/F é de apenas 7%, para tráfegos não uniformes considerar a fila é tão importante quanto ter um algoritmo sem colisões com resultado maximal. O algoritmo MICE-EsMa é 64% melhor que o EsMa-S/F ($\rho = 0.9$), reforçando a importância de considerar as filas.

O algoritmo MICE-EsMAI está com baixa performance para esse tipo de tráfego, ficando pior que o MICE-EsMa, provavelmente pela escolha de c para este cenário.

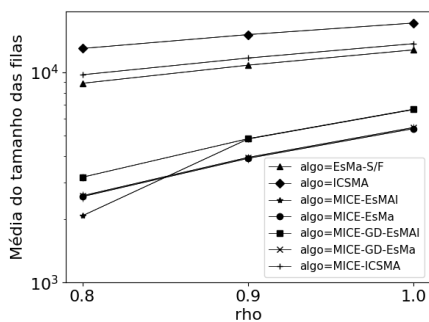


Figura 7. Comparação entre os algoritmos propostos, $r = 0.2$

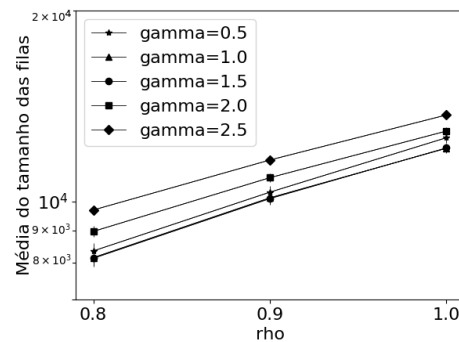


Figura 8. Algoritmo MICE-ICSMA para diferentes valores de γ , $\beta = 0.1$

O algoritmo MICE-ICSMA apresenta comportamento diferente para $r = 0.2$ em relação ao parâmetro γ quando comparado com o tráfego uniforme, ou seja, a média das filas com $\gamma \in \{1.5, 1.0\}$ são menores quando comparadas com as filas quando $\gamma \in \{0.5, 2.0, 2.5\}$, como visto na Figura 8. O resultado difere do resultado obtido para o tráfego uniforme, no cenário não uniforme é preciso que a probabilidade $q(1; \sigma, v)$ se aproxime de 1 de forma mais moderada, e quanto maior o γ mais rapidamente a probabilidade se aproxima de 1.

5. Conclusões e Trabalhos Futuros

A partir da contribuição do trabalho de [Wang and Xia 2017a], com maior liberdade de mover a cadeia de Markov sobre todo o espaço de configurações, propomos a inclusão do campo externo do modelo de Ising na adaptação para considerar o tamanho da fila dos enlaces. Como dito anteriormente, haviam dois problemas na proposta original, mínimos locais e escalonamentos não-maximais. A inclusão do campo externo soluciona o problema de mínimos locais e os algoritmos de mapeamento propostos demonstraram, através das simulações, a importância do resultado ser um conjunto maximal, solucionando o segundo problema. Os algoritmos EsMa e EsMAI, apesar de terem diferença entre o tempo em que é calculado e o tempo em que é realizado, apresentaram bons resultados.

Como trabalho futuro está provar se, com a inclusão do campo externo, o algoritmo é *throughput-optimal*. Avaliar mais valores de β e γ , simulações para valores de $\beta > 1.5$ e $\gamma > 2.5$ podem produzir resultados ainda melhores e ajudar a compreender melhor sua interação. O baixo desempenho do algoritmo EsMAI para o tráfego não uniforme se deve provavelmente a escolha de $c = 4$, um número menor como $c = 2$ talvez tenha um resultado melhor. Também seria interessante avaliar a formulação do I-CSMA para os algoritmos EsMa e EsMAI.

Referências

- Bissacot, R. and Cioletti, L. (2010). Phase transition in ferromagnetic ising models with non-uniform external magnetic fields. *Journal of Statistical Physics*, 139(5):769–778.
- Jiang, L., Leconte, M., Ni, J., Srikant, R., and Walrand, J. (2012). Fast mixing of parallel glauber dynamics and low-delay csma scheduling. *IEEE Transactions on Information Theory*, 58(10):6541–6555.
- Kwak, J., Lee, C. H., and Eun, D. Y. (2016). A high-order markov-chain-based scheduling algorithm for low delay in csma networks. *IEEE/ACM Transactions on Networking*, 24(4):2278–2290.
- Ni, J., Tan, B., and Srikant, R. (2010). Q-csma: Queue-length based csma/ca algorithms for achieving maximum throughput and low delay in wireless networks. In *2010 Proceedings IEEE INFOCOM*, pages 1–5.
- Rhee, I., Warrier, A., Min, J., and Xu, L. (2009). Drand: Distributed randomized tdma scheduling for wireless ad hoc networks. *IEEE Transactions on Mobile Computing*, 8(10):1384–1396.
- Tassiulas, L. and Ephremides, A. (1992). Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks. *IEEE Transactions on Automatic Control*, 37(12):1936–1948.
- Wang, Y. and Xia, Y. (2017a). I-csma: A link scheduling algorithm for wireless networks based on ising model. *IEEE Transactions on Control of Network Systems*, PP(99):1–1.
- Wang, Y. and Xia, Y. (2017b). Improving the queue size and delay performance with the i-csma link scheduling algorithm. *Computer Networks*, 122(Supplement C):105 – 119.
- Xue, D., Ekici, E., Ibrahim, R., and Youssef, M. (2017). A novel queue-length-based csma algorithm with improved delay characteristics. *Computer Networks*, 122(Supplement C):56 – 69.