

Uma Abordagem Bidinâmica para a Identificação de Etiquetas RFID

Shalton Viana dos Santos, Paulo André da S. Gonçalves

Centro de Informática (CIn) – Universidade Federal de Pernambuco (UFPE)
50.740-560 – Recife – PE – Brasil

{svs2, pasg}@cin.ufpe.br

Abstract. *DFSA (Dynamic Framed Slotted ALOHA) is an access medium control protocol commonly used in RFID systems. Typically, the size of the frames under approaches without frame size resetting is dynamically adjusted in accordance with the specific value returned by the tag estimation method used. We propose a different and dual dynamic approach for the tag identification process. Our approach is doubly dynamic because it takes advantage of the dynamism of DFSA while using dynamic multiplicative factors for readjusting the specific value returned by the tag estimation method used. We evaluate our approach through simulation by using three different tag estimation methods. Time requirements of the communication channel are modelled in accordance with ISO 18000-6 Type C standard. The results show that our dual dynamic approach allows to optimize the system by minimizing the tag identification time under all the tag estimation methods evaluated.*

Resumo. *O DFSA (Dynamic Framed-Slotted Aloha) é um protocolo de acesso ao meio comumente usado em sistemas RFID. Tradicionalmente, o tamanho dos quadros é ajustado dinamicamente em função apenas do valor retornado pelo estimador de população de etiquetas empregado em abordagens sem reset de quadro. Este artigo propõe uma abordagem bidinâmica para o processo de identificação de etiquetas RFID. A proposta é duplamente dinâmica por contar com a dinamicidade no reajuste de tamanho de quadros do DFSA e por usar fatores multiplicativos dinâmicos no reajuste de estimativas retornadas pelo estimador utilizado no sistema RFID. As avaliações de desempenho da abordagem proposta foram realizadas através de simulação utilizando-se três estimadores distintos e requisitos temporais do canal de comunicação no padrão ISO 18000-6 Type C. Os resultados mostram que a abordagem bidinâmica permite otimizar o sistema para todos os estimadores estudados, reduzindo o tempo total de identificação de etiquetas.*

1. Introdução

A Internet das Coisas ou *Internet of Things* (IoT) é uma nova e promissora abordagem de rede composta por objetos físicos inteligentes (e.g. sensores, atuadores, eletrodomésticos, dispositivos eletrônicos, veículos, prédios, produtos em supermercados, produtos industrializados em geral) que podem, dependendo da aplicação, realizar comunicações autônomas, interagir entre si e trocar dados com a Internet [Perera et al. 2015] [Al-Fuqaha et al. 2015]. Existem aplicações de IoT no cotidiano onde é interessante que

exista alguma maneira de contar e identificar de forma rápida e automática uma determinada quantidade de objetos. Como exemplo, quanto mais rápido um caixa de supermercado contar e identificar os produtos de uma determinada compra, menores seriam as filas e mais satisfeitos ficariam os clientes com a eficiência dos caixas. Outro exemplo está após o processo de fabricação de grandes quantidades de produtos (*e.g.* canetas, parafusos, tampas), onde uma contagem automática deve ser rapidamente feita no momento de embalagem, distribuição e recepção pelo revendedor.

Os sistemas RFID (*Radio Frequency IDentification*) aparecem como os mais promissores no contexto de IoT não só para a identificação automática de objetos como também para outras aplicações. As principais razões para isso decorrem dos seguintes fatores: 1) a comunicação é através de sinais de radiofrequência (RF), não requerendo linha de visada direta; 2) o alcance de comunicação varia de centímetros a vários metros; 3) etiquetas RFID permitem suporte a aplicações de sensoriamento, localização, contabilização e indexação de objetos; 4) as etiquetas RFID são econômicas em termos de recursos de *hardware*, o que reduz custos financeiros; e 5) as etiquetas RFID podem funcionar sem bateria e sem necessidade de recarga constante caso usem bateria.

Os sistemas RFID mais simples são compostos por um leitor, possuindo ou não um servidor associado, e uma ou mais etiquetas. Os leitores podem ser ou não portáteis e, geralmente, não são tão restritos em termos de processamento e memória quanto as etiquetas. A depender do uso de bateria para a tarefa de comunicação, as etiquetas podem ser classificadas em passivas, semi-ativas ou ativas. As passivas não possuem bateria. Elas obtêm energia do sinal RF recebido do leitor e transmitem usando uma técnica conhecida por *backscatter*. As etiquetas ativas possuem bateria para comunicação e processamento. As etiquetas semi-ativas utilizam *backscatter* para comunicação e a bateria para processamento. Cada etiqueta possui um identificador (ID) único, podendo ser colada ou embutida em um objeto. E a depender da aplicação, o leitor requisita o ID das etiquetas em seu alcance de comunicação para identificação dos objetos. Nesse processo, como existe a possibilidade de haver colisões de transmissões, é necessário utilizar um protocolo anti-colisão de etiquetas a fim de se resolver os conflitos de transmissão e permitir uma rápida identificação de todos os objetos.

Dentre os protocolos anti-colisão para sistemas RFID, o DFSA (*Dynamic Framed Slotted ALOHA*) é um dos mais populares [EPC Global 2015, Klair et al. 2010]. A execução do DFSA é orientada pelo leitor que organiza o tempo em um ou mais quadros. Cada quadro é ainda subdividido em *slots* de tempo. As etiquetas são requisitadas a transmitir em um *slot*, aleatoriamente escolhido a cada quadro, até serem identificadas pelo leitor. Contudo, o DFSA requer o uso de alguma técnica para reajuste dinâmico do tamanho dos quadros. Existem vários estudos que fazem isso com base na estimativa da população de etiquetas competindo por *slots* [Andrade and Gonçalves 2013, Klair et al. 2010] ou com base em algum algoritmo [EPC Global 2015].

Os estimadores ou algoritmos de ajuste de tamanho de quadro para o DFSA em sistemas RFID podem ser divididos em duas classes básicas: com e sem *reset* de quadro. A primeira classe recalcula o valor do próximo quadro a ser aberto somente ao término da execução de um quadro completo [Andrade and Gonçalves 2013, Eom and Lee 2010, Li and Wang 2011, Wu and Zeng 2010, Tong et al. 2009, Chen 2009, Vogt 2002, Schoute 1983]. A segunda classe decide ao término de cada *slot* ou de um subgrupo de *slots*

consecutivos se cancela ou não a execução dos *slots* restantes e recalcula o tamanho do próximo quadro [Šolić et al. 2016, Chen 2014, EPC Global 2015].

Tradicionalmente, o tamanho dos quadros no DFSA na classe sem *reset* é ajustado dinamicamente em função apenas do valor retornado pelo estimador de população de etiquetas utilizado. Assim sendo, há apenas um fator dinâmico sendo utilizado no processo de identificação de etiquetas RFID. De forma diferente, este artigo propõe uma abordagem bidinâmica para esse processo. A proposta é duplamente dinâmica por contar com a dinamicidade no reajuste de tamanho de quadros do DFSA e por usar fatores multiplicativos dinâmicos no reajuste de estimativas retornadas pelo estimador utilizado no sistema RFID. Para demonstrar a potencialidade da proposta, foram realizadas avaliações de desempenho através de simulações com os estimadores clássicos *Lower Bound* e Schoute bem como com o estimador Eom-Lee, sendo este último um dos mais acurados já propostos na literatura. A modelagem do canal de comunicação no estudo de simulação é feita com base na estrutura de temporização do canal de comunicação do padrão ISO 18000-6 Type C, o qual é também conhecido como EPC Class-1 Generation 2 [EPC Global 2015].

O foco deste trabalho está em abordagens que analisam todo o quadro em execução para cálculo do tamanho do quadro subsequente. Os estudos apresentados neste artigo utilizam o *Lower Bound* e o Schoute devido ao fato de serem os estimadores mais simples já propostos e com custo computacional muito baixo. Em contrapartida, eles possuem baixa acurácia, o que reflete, geralmente, em uma maior quantidade de *slots* utilizados no processo de identificação de etiquetas em relação a estimadores mais acurados e pior tempo de identificação de etiquetas [Andrade and Gonçalves 2011]. Por outro lado, o estimador Eom-Lee possui um custo computacional elevado, sendo porém um dos estimadores mais acurados já propostos na literatura [Andrade and Gonçalves 2013, Eom and Lee 2010]. Desta forma, é possível mostrar o impacto da proposta deste artigo com tipos de estimadores distintos em relação ao custo computacional, à qualidade de estimação e ao tempo de identificação.

Os resultados alcançados mostram que a abordagem bidinâmica proposta permite otimizar o sistema com qualquer um dos estimadores sem *reset* estudados, reduzindo o tempo total de identificação de etiquetas. As principais contribuições deste artigo são a proposição de um otimizador de estimadores sem *reset* de quadros e a demonstração de que é possível se alcançar um desempenho similar ao do Eom-Lee utilizando o simples *Lower Bound* com o otimizador resultante da proposta bidinâmica e com custo computacional significativamente menor. O trabalho proposto é uma extensão e generalização de abordagem inicialmente proposta em [Andrade and Gonçalves 2011] que utiliza fatores multiplicativos fixos, independentes da saída do estimador, em vez de fatores multiplicativos dinâmicos, dependentes da saída do estimador do sistema RFID para reajuste de quadros. A otimização de abordagens derivadas do estimador Vogt [Andrade and Gonçalves 2013, Li and Wang 2011] ou que resetam o tamanho dos quadros em momento de execução a partir da análise de cada *slot* ou um subgrupo de *slots* consecutivos será abordada em trabalhos futuros.

O restante deste artigo está organizado da seguinte forma: a Seção 2 apresenta os trabalhos relacionados. A Seção 3 detalha e discute especificamente os estimadores *Lower Bound*, Schoute e Eom-Lee. A Seção 4 apresenta e discute a abordagem bidinâmica proposta para a identificação de etiquetas RFID. A Seção 5 apresenta as avaliações de

desempenho realizadas com canal de comunicação sob requisitos temporais do padrão ISO 18600-6 Type C. Por fim, a Seção 6 conclui este trabalho.

2. Trabalhos Relacionados

Esta seção apresenta os principais estimadores e algoritmos para o DFSA com e sem *reset* de quadros. Para facilitar a leitura, a Tabela 1 apresenta os principais parâmetros utilizados e o significado de cada um.

Tabela 1. Lista de Parâmetros.

Parâmetro	Significado
f	tamanho do quadro executado ou sendo executado
\hat{f}	tamanho do quadro subsequente calculado pelo estimador
\hat{n}	estimativa de etiquetas não identificadas (<i>backlog</i>) calculada pelo estimador
s_s	quantidade de <i>slots</i> bem sucedidos
s_v	quantidade de <i>slots</i> vazios
s_c	quantidade de <i>slots</i> em colisão

2.1. Sem Reset

A Figura 1 ilustra o funcionamento da classe sem *reset*. Basicamente, um quadro de tamanho f é aberto e cada etiqueta escolhe aleatoriamente um *slot* para transmitir informações. Ao término do quadro, caso haja ao menos um *slot* em colisão, executa-se um estimador para se determinar o tamanho \hat{f} do quadro subsequente a ser utilizado. Os parâmetros utilizados como entrada para um estimador variam de acordo com a proposta. Contudo, essas entradas costumam ser os parâmetros s_v , s_s , s_c e f ou um subconjunto deles, onde s_v , s_s e s_c representam, respectivamente, o total de *slots* vazios, bem sucedidos e em colisão no quadro executado com tamanho f . O estimador calcula o tamanho \hat{f} do quadro subsequente com base na estimativa do quantitativo de etiquetas que não foram identificadas no quadro de tamanho f executado. Essa estimativa é representada neste trabalho por \hat{n} .

Existem diversos estimadores sem *reset* de quadro propostos na literatura como o *Lower Bound* [Vogt 2002], o Schoute [Schoute 1983], o Vogt [Vogt 2002], o Eom-Lee [Eom and Lee 2010], o Chen [Chen 2009], o IV-II [Andrade and Gonçalves 2013], o CMEBE [Li and Wang 2011] e estimadores Bayesianos como os apresentados em [Wu and Zeng 2010] e [Tong et al. 2009]. Os estimadores *Lower Bound*, Schoute e Eom-Lee são utilizados nas avaliações de desempenho neste artigo e, por isso, são apresentados e discutidos separadamente na Seção 3. Os demais estimadores são apresentados a seguir.

O estimador Vogt [Vogt 2002] modela o processo de identificação de etiquetas com base em uma distribuição binomial. Basicamente, o estimador procura um quantitativo de etiquetas que minimiza a norma Euclidiana da diferença entre dois vetores: um com os valores de s_v , s_s e s_c e outro com os valores esperados para esses parâmetros. Esse quantitativo de etiquetas é a quantidade estimada de etiquetas que competiram por *slots* no quadro de tamanho f analisado. O tamanho \hat{f} do quadro subsequente é igual a esse quantitativo menos s_s . Em [Eom and Lee 2010] é sugerido que o Vogt possui maior custo computacional do que o Eom-Lee. Contudo, a literatura ainda carece de estudos

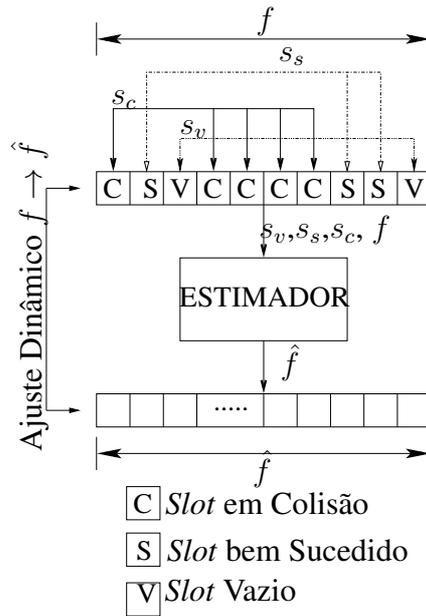


Figura 1. Abordagem sem reset de quadro.

mais detalhados com comparativos de custos entre o Vogt e o Eom-Lee em função do quantitativo de etiquetas a serem identificadas.

O IV-II [Andrade and Gonçalves 2013] identifica e trata a deficiência do estimador Vogt na produção de estimativa de etiquetas quando o quadro analisado possui todos os *slots* em colisão. Os resultados sugerem que o IV-II e o Eom-Lee são equivalentes em termos de quantidade de *slots* totais gerados no processo de identificação com um quadro inicial de 64 *slots*. Quando o quadro inicial é de 128 *slots*, os dois estimadores apresentam desempenho equivalente até 700 etiquetas. Acima disso, o IV-II apresenta melhor desempenho, chegando a usar até 69 *slots* a menos para identificar 1.000 etiquetas. Adicionalmente, o IV-II possui desempenho idêntico ou melhor do que o do Vogt, chegando a usar até menos 232 *slots* para identificar 1.000 etiquetas. O custo do IV-II é ligeiramente maior do que o do Vogt.

O estimador Chen [Chen 2009] também modela inicialmente o processo de identificação de etiquetas com base em uma distribuição binomial. Contudo, ele procura determinar a probabilidade de se obter exatamente s_v vazios, s_s sucessos e s_c colisões dado um quadro de tamanho f . A solução para esse problema deriva uma equação de probabilidades $P(n|f, s_v, s_c, s_s)$ que requer o cálculo de vários fatoriais e exponenciações. O valor de n que maximiza a probabilidade $P(n|f, s_v, s_c, s_s)$ é a estimativa de etiquetas que competiram por *slots* no quadro de tamanho f . Em geral, o estimador Chen usa mais *slots*, é menos acurado e mais custoso do que os estimadores Eom-Lee e IV-II [Eom and Lee 2010, Andrade and Gonçalves 2013]

Os estimadores em [Wu and Zeng 2010] e [Tong et al. 2009] se diferenciam dos demais por utilizarem um processo Bayesiano para a determinação da estimativa da quantidade de etiquetas que competiram por *slots* num quadro e para o cálculo do tamanho ótimo do quadro subsequente. A contrapartida é o custo computacional associado ao processo e, por isso, técnicas de redução de espaço de busca precisam ser utilizadas para

tornar viável o uso de estimadores desse tipo. Em [Wu and Zeng 2010] é mostrado como se reduzir o espaço de buscas e como se reduzir a complexidade computacional.

Em [Andrade and Gonçalves 2011] é apresentada a proposta de uma função de cálculo de tamanho de quadros para o DFSA. A função proposta reajusta o tamanho de quadro obtido pelo estimador empregado. Os estudos foram realizados com o *Lower Bound*, Schoute e Eom-Lee. A função ajusta o valor do tamanho de quadro da saída do estimador, multiplicando-o por um fator fixo cujo valor depende do estimador, mas independe do valor de saída do estimador. Os resultados mostram que é possível reduzir significativamente o tempo total de identificação de etiquetas dependendo da população de etiquetas a ser identificada e da relação temporal entre os diferentes tipos de *slots* de tempo.

Outros estimadores sem *reset* focam em melhorar a qualidade das estimativas em cenários com canais de comunicação sob efeito de captura como o CMEBE [Li and Wang 2011]. Esse estimador pode ser visto como uma variante do Vogt que considera um canal de comunicação com efeito de captura e calcula o tamanho \hat{f} do quadro subsequente com base em uma estimativa \hat{n} de etiquetas e uma estimativa da probabilidade média de captura em *slots* em colisão.

2.2. Com Reset

Outros estudos na literatura focam no desenvolvimento de estimadores com *reset*, onde o quadro de tamanho f , em execução, é analisado ao término de cada *slot* ou de um conjunto consecutivo de *slots*. Com base na análise realizada, o estimador pode decidir cancelar a abertura dos *slots* restantes e abrir imediatamente um novo quadro com um tamanho \hat{f} recalculado. Comparada com a abordagem sem *reset*, a com *reset* incorre em processamento extra a cada *slot* ou conjunto de *slots* devido ao algoritmo de decisão de anulação de *slots*. Além disso, o cancelamento de *slots* aumenta o custo das mensagens do leitor para as etiquetas visto a necessidade de informá-las sobre o cancelamento de um quadro para a geração de novo número pseudoaleatório.

O estimador Chen II [Chen 2014] pode ser visto como uma versão do Schoute que analisa grupos consecutivos de *slots*. Ao analisar um grupo de *slots*, o estimador verifica se os parâmetros de desempenho obtidos a partir do grupo analisado correspondem ao esperado para se maximizar a eficiência do sistema. Caso não correspondam, os *slots* restantes são cancelados e um novo quadro é aberto. A estimativa \hat{n} é igual a $(s_s + 2,39s_c)f/i$, onde $4 \leq i \leq f$ é o índice do último *slot* do subgrupo sendo analisado. O tamanho do quadro subsequente \hat{f} é dado por \hat{n} menos o quantitativo de *slots* bem sucedidos até o *slot* i . Caso não haja *reset* do quadro, o tamanho do quadro subsequente e a estimativa de etiquetas são obtidas conforme o estimador Schoute.

O padrão EPC Classe 1 Gen2 [EPC Global 2015] especifica dois procedimentos de identificação de etiquetas: um com quadros de tamanho fixo igual a 2^Q , ou seja, um FSA (*Framed Slotted Aloha*) simples e outro com quadros ajustados dinamicamente com tamanho 2^Q , porém, com Q recalculado ao término de cada *slot* com base no algoritmo-Q. Este segundo procedimento resolve ineficiências típicas de um sistema FSA e nada mais é do que um DFSA com *reset* de quadro ao término de cada *slot* vazio ou em colisão. Quando um *slot* termina, o leitor analisa o tipo de *slot*. De acordo com o resultado da análise, o leitor atualiza o valor do parâmetro Q_{fp} que nada mais é do que a versão em ponto flutuante

do parâmetro inteiro Q e é calculado como segue: $Q_{fp} = \min(15, Q_{fp} + C)$ para *slot* em colisão, $Q_{fp} = \max(0, Q_{fp} - C)$ para *slot* vazio e $Q_{fp} = Q_{fp} + 0$ para *slot* bem sucedido. O novo valor do parâmetro Q passa a ser $\text{round}(Q_{fp})$. Valores para C são dinâmicos e estão no intervalo de 0,1 a 0,5. A dificuldade está em se determinar valores adequados para C visto que valores ótimos são dependentes do número de etiquetas competindo por *slots*. O padrão não define como calcular valores adequados para C . Em [Khanna and Uysal 2015] é proposta uma modificação no algoritmo-Q. A proposta utiliza informações da camada física para estimar a quantidade de etiquetas que transmitiram em um mesmo *slot* e, assim, calcular um valor otimizado para o parâmetro Q .

O ILCM Sbs [Šolić et al. 2016] é uma extensão de trabalhos anteriores dos autores e foca em produzir um valor Q otimizado para sistemas baseados no algoritmo-Q. Embora tenha como base uma equação de probabilidades $P(n|f, s_v, s_c, s_s)$ que também requer o cálculo de vários fatoriais e exponenciações, os autores conseguem encontrar funções menos custosas. Os resultados para um quantitativo de até 250 etiquetas mostram que o ILCM Sbs permite um tempo de identificação inferior ao de uma versão do Vogt com *reset* de quadros.

3. Os Estimadores Estudados

Esta seção detalha e discute os estimadores *Lower Bound*, Schoute e Eom-Lee dado que serão os utilizados nas avaliações de desempenho neste artigo. Como explicitado na Seção 1, a escolha dos estimadores *Lower Bound* e Schoute é em virtude da simplicidade e do baixo custo computacional deles. Em contrapartida, são estimadores de baixa acurácia, o que reflete, geralmente, em um maior tempo de identificação de etiquetas em relação a estimadores mais acurados. Por outro lado, o estimador Eom-Lee possui um custo computacional significativamente maior, porém é um dos mais acurados já propostos na literatura. Assim, é possível avaliar o impacto da proposta neste artigo com estimadores distintos em relação ao custo computacional, à qualidade de estimação e ao tempo total de identificação de etiquetas. A Tabela 2 apresenta o custo de operações em ponto flutuante (FLOP) [Vales-Afonso et al. 2015] para análise dos estimadores.

Tabela 2. Custos de operações em ponto flutuante (FLOP).

Operação	Custo
adição, subtração, multiplicação	1
comparação	2
divisão, raiz quadrada	10
exponenciação, logaritmo	50
fatorial	100

3.1. Lower Bound

Formalmente, o estimador *Lower Bound* define [Vogt 2002]:

$$\hat{n} = s_s + 2 \cdot s_c, \quad (1)$$

$$\hat{f} = 2 \cdot s_c. \quad (2)$$

A lógica desse estimador é simples: há ao menos duas etiquetas envolvidas em uma colisão. Logo, a menor quantidade possível de etiquetas que competiram por *slots* em um quadro sendo analisado pode ser facilmente calculada, sendo igual a quantidade total de *slots* com transmissões bem sucedidas somada com o dobro da quantidade de *slots* em colisão. O *backlog* é a quantidade de etiquetas que ainda não foram identificadas e possui valor igual ao dobro da quantidade de *slots* em colisão. Para se maximizar a eficiência do sistema quando todos os *slots* possuem o mesmo tamanho, o quadro subsequente ao quadro sendo analisado deve ter tamanho igual ao *backlog* [Schoute 1983].

O *Lower Bound* é um estimador grosseiro, pois sempre estima a quantidade mínima possível de etiquetas restantes dentro de uma gama de possibilidades. Assim, ele pode ser visto como uma fronteira para se avaliar a qualidade de outros estimadores. Contudo, o custo FLOP dele é baixo. Com o auxílio da Tabela 2 é possível notar que o custo FLOP é igual 2 para o cálculo conjunto de cada tamanho de quadro e \hat{n} . Note que \hat{n} é obtido através de simples soma tendo \hat{f} já calculado. A Figura 2(a) apresenta o custo FLOP total de uso do estimador em função da quantidade de etiquetas a serem identificadas. Os resultados na Figura 2 são médias obtidas considerando-se a Tabela 2 e simulações de desempenho do estimador com os mesmos parâmetros apresentados na Seção 5, exceto pelo passo para o quantitativo de etiquetas. O intervalo de confiança é de 99% e é representado por barras de erro quase sempre imperceptíveis.

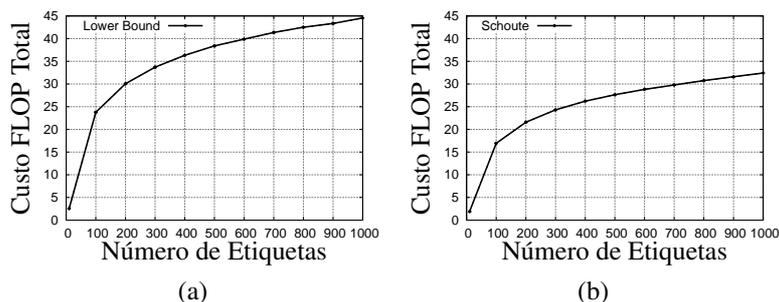


Figura 2. FLOP para os estimadores *Lower Bound* e *Schoute*.

3.2. Schoute

Formalmente, o estimador Schoute [Schoute 1983] define:

$$\hat{n} = s_s + 2,39 \cdot s_c, \quad (3)$$

$$\hat{f} = 2,39 \cdot s_c. \quad (4)$$

A mudança em relação ao *Lower Bound* está no uso do fator multiplicativo 2,39 ao invés de 2. Esse novo fator é o número esperado de etiquetas que transmitirão em cada *slot* em colisão no próximo quadro ao que está sendo analisado. A formulação de Schoute é obtida considerando-se um processo de chegadas do tipo Poisson. Para outras distribuições, as Equações (3) e (4) trazem resultados subótimos.

Note com base nas Figuras 2(a) e 2(b) que o Schoute possui, em geral, um custo FLOP total menor do que o do *Lower Bound* apesar de ambos terem o mesmo custo FLOP

no cálculo conjunto do tamanho do quadro e da estimativa de etiquetas. Isso ocorre porque o Schoute utiliza, em geral, menos quadros no processo de identificação de etiquetas. Assim, ele é executado menos vezes, impactando positivamente no custo FLOP total de uso do estimador. Para sistemas que requerem apenas o valor de \hat{f} , o custo FLOP total mostrado nas Figuras 2(a) e 2(b) cai pela metade.

3.3. O Estimador Eom-Lee

O estimador Eom-Lee [Eom and Lee 2010] utiliza um algoritmo iterativo tanto para estimar a quantidade de etiquetas que competiram por *slots* em um quadro analisado de tamanho L quanto para calcular o tamanho do quadro subsequente. O estimador Eom-Lee define que

$$\hat{f} = \gamma \cdot s_c, \quad (5)$$

onde

$$\gamma = \frac{1 - e^{-\frac{1}{\beta}}}{\beta(1 - (1 + \frac{1}{\beta})e^{-\frac{1}{\beta}})}. \quad (6)$$

Contudo, não é trivial encontrar uma solução fechada para se determinar os valores de γ e β a partir da Eq. (6). O problema é contornado calculando-se γ e β de forma iterativa. Considere γ_k e β_k , respectivamente, uma aproximação para o valor de γ e de β na k -ésima iteração do algoritmo. Essas aproximações são obtidas de acordo com as seguintes equações:

$$\beta_k = \frac{L}{\gamma_{k-1} \cdot s_c + s_s}, \quad (7)$$

$$\gamma_k = \frac{1 - e^{-\frac{1}{\beta_k}}}{\beta_k(1 - (1 + \frac{1}{\beta_k})e^{-\frac{1}{\beta_k}})}. \quad (8)$$

No primeiro passo do algoritmo iterativo, considera-se $\beta_1 = \infty$ e $\gamma_1 = 2$ e em cada passo k seguinte se determina uma nova aproximação para β e γ com o auxílio das Eqs. (7) e (8), respectivamente. Quando $|\gamma_{k^*-1} - \gamma_{k^*}|$ for menor que um limiar pré-definido $\epsilon_{threshold}$, o processo iterativo é interrompido. Os valores γ_{k^*-1} e γ_{k^*} representam, respectivamente, a aproximação anterior e atual para o valor de γ . A partir de então, o tamanho \hat{f} do próximo quadro e a quantidade estimada \hat{n} de etiquetas são obtidos, respectivamente, pelas Eqs. (9) e (10), onde β_{k^*} é a aproximação mais recente para o valor de β .

$$\hat{f} = \gamma_{k^*} \cdot s_c. \quad (9)$$

$$\hat{n} = \frac{\hat{f}}{\beta_{k^*}}. \quad (10)$$

O custo do Eom-Lee por cálculo de tamanho de quadro depende do número de iterações calculando-se β_k e γ_k até se atingir o critério de parada. Após atingir o critério de parada, há ainda o cálculo de \hat{f} que adiciona 1 ao custo total das iterações e o cálculo de \hat{n} que adiciona mais 10 ao custo. A Figura 3(a) apresenta o número total de iterações realizadas pelo estimador ao longo do processo de estimação de etiquetas. Note que o estimador possui um aumento abrupto de iterações a partir de 300 etiquetas. A Figura 3(b) apresenta o custo FLOP total de uso do estimador em função da quantidade de etiquetas a serem identificadas. Note que o custo FLOP total é significativamente maior do que o dos estimadores *Lower Bound* e Schoute. Para sistemas que requerem apenas o cálculo de \hat{f} , não há mudança significativa no custo visto que o número de iterações e cálculos nessas iterações possuem um peso muito mais importante. Os resultados na Figura 3 são médias obtidas considerando-se a Tabela 2 e simulações de desempenho do estimador com os mesmos parâmetros apresentados na Seção 5. Os intervalos de confiança são de 99%.

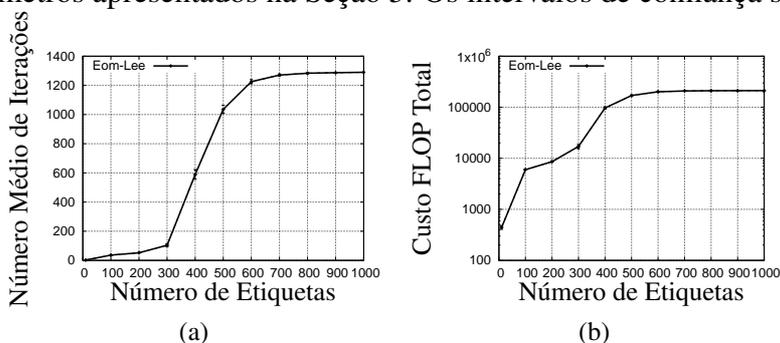


Figura 3. Número médio de iterações e custo FLOP total para o Eom-Lee.

4. A Abordagem Bidinâmica Proposta

Conforme mencionado na Seção 1, o tamanho dos quadros no DFSA é ajustado dinamicamente em função apenas de valor retornado pelo estimador de população de etiquetas utilizado. Desta forma, nota-se que há apenas um fator dinâmico sendo utilizado no processo de identificação de etiquetas. A proposta deste trabalho consiste em uma abordagem que adicionalmente analisa o valor retornado pelo estimador e o reajusta também dinamicamente tendo como base uma função de otimização aqui proposta. Assim, o bidinamismo da proposta é consequência do reajuste dinâmico de tamanho de quadros e do reajuste dinâmico da saída do estimador empregado. A Figura 4 apresenta uma esquematização da abordagem bidinâmica proposta.

Note que o diferencial em relação à Figura 1 está no módulo *Otimizador* que altera dinamicamente o valor de \hat{f} que é retornado pelo estimador utilizado. O módulo *Otimizador* proposto neste artigo é uma generalização inspirada a partir dos achados apresentados em [Andrade and Gonçalves 2011]. Nesse trabalho, os autores demonstram ser possível reduzir o tempo total de identificação de etiquetas ao se multiplicar todo \hat{f} retornado pelo estimador por um mesmo fator fixo calculado de forma adequada. Em vez de utilizar um mesmo fator fixo, o módulo *Otimizador* proposto usa um fator otimizado que varia de acordo com o resultado de saída do estimador.

Seja p a população de etiquetas a serem identificadas. Seja $\bar{\Omega}(i, p, \delta)$ uma função que representa o tempo médio total de identificação das p etiquetas utilizando-se um quadro inicial com i slots e um fator multiplicativo de correção de tamanho de quadro \hat{f} igual

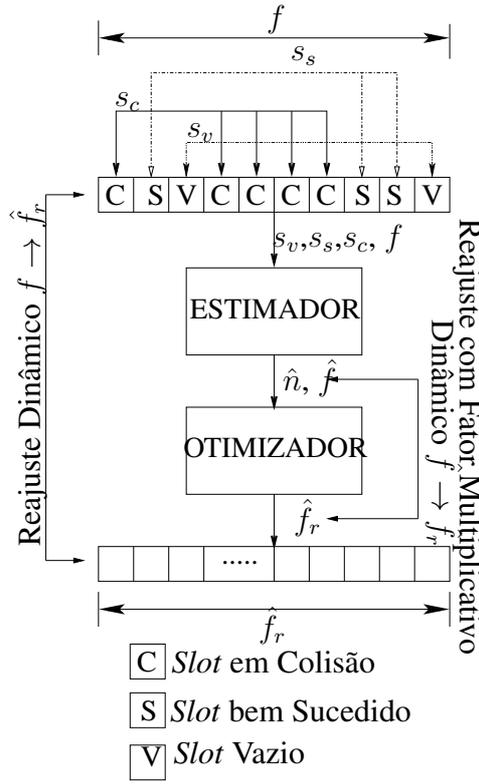


Figura 4. Abordagem bidinâmica para ajuste de tamanho de quadros.

a δ . A função $\bar{\Omega}$ é facilmente obtida através de simulação de Monte-Carlo para qualquer estimador desejado e seguindo o tamanho dos *slots* conforme norma utilizada pelo sistema RFID. A partir da função $\bar{\Omega}$, o tamanho do quadro corrigido dinamicamente para qualquer estimativa \hat{n} e, respectivo tamanho de quadro \hat{f} , calculados pelo estimador empregado é dado por:

$$\hat{f}_r = \hat{f} \times \underset{\delta > 0}{\operatorname{argmin}} \bar{\Omega}(i, p, \delta) \quad \text{para } p = \hat{n}, \quad (11)$$

significando que \hat{f}_r é igual a estimativa \hat{f} do estimador empregado multiplicada pelo valor de δ que minimiza o valor da função $\bar{\Omega}$ para $p = \hat{n}$, ou seja, para p igual ao valor da estimativa de etiquetas calculado pelo estimador.

A função $\bar{\Omega}$ não deve ser calculada em tempo de execução por causa de seu custo de processamento. Ela deve ser pré-computada e os dados produzidos podem ser pré-instalados em forma de uma árvore binária em memória com p sendo chave de busca para resultado pré-computado de $\underset{\delta > 0}{\operatorname{argmin}} \bar{\Omega}(i, p, \delta)$ e dado i . Isso permite que apenas simples consultas sejam feitas em tempo de execução, reduzindo-se de forma importante o tempo de processamento do otimizador. Portanto, o custo do otimizador proposto em relação ao uso exclusivo do estimador escolhido está no uso adicional de memória no leitor ou servidor associado, na realização adicional de uma multiplicação e na realização de uma busca em uma árvore binária na memória. Não há alterações nas etiquetas visto que o otimizador e o estimador são sempre executados no leitor RFID ou servidor associado. O custo de memória não é significativo. Cada nó da árvore precisa armazenar um único valor de δ para o valor de i definido no sistema RFID considerado. Valores típicos de i são 32,

64 e 128 *slots*. Além disso, o uso de árvore binária balanceada reduz significativamente o custo das consultas visto a complexidade de pior caso ser igual a $O(\log p_{max})$, onde p_{max} é o número total de nós da árvore. Estudos na literatura consideram tipicamente a identificação de até 1.000 etiquetas, o que limita o valor de p_{max} .

5. Avaliação de Desempenho

Esta seção apresenta um estudo de simulação para avaliação do desempenho da abordagem bidinâmica proposta face a abordagem tradicional, considerando-se os estimadores *Lower Bound*, *Schoute* e *Eom-Lee*. Para isso, o simulador do protocolo DFSA para RFID desenvolvido por [Andrade and Gonçalves 2013] foi estendido com o módulo *Otimizador*. As simulações apresentadas consideram um sistema básico RFID com um leitor e um determinado quantitativo de etiquetas a serem identificadas. Esse quantitativo é variado de 10 a 1.000 em passos de 10. As métricas de avaliação são o tempo total de identificação e a redução média percentual no tempo total de identificação de etiquetas. Ambas as métricas consideram apenas o tempo gasto com *slots* vazios, em colisão e bem sucedidos. Esses tempos são os seguintes conforme norma ISO 18000-6 Type C: 1) tempo de *slot* bem sucedido = $T_S = T4 + TQuery + 2T1 + 2T2 + TRN16 + TACK + TPC + EPC + CRC16 + TQREP = 2312 \mu s$; 2) tempo de *slot* em colisão = $T_C = T1 + T2 + TRN16 = 337,5 \mu s$; 3) tempo de *slot* vazio = $T_E = T1 + T3 = 67,5 \mu s$.

Nos gráficos apresentados, cada ponto em cada curva apresentada é uma média dos resultados de 2.000 simulações. O canal de comunicação é livre de erros visto o intuito de se estudar o impacto isolado dos estimadores e deles com o otimizador. Contudo, a proposta é aplicável com estimadores que levam em conta efeitos de captura, o que será melhor investigado em trabalhos futuros. Em particular às simulações com o estimador *Eom-Lee*, o parâmetro $\epsilon_{threshold}$ é igual a 0,001. Todas as simulações consideram um quadro inicial de 64 *slots*. Todos os resultados foram calculados com intervalo de confiança de 99%.

A Figura 5(a) mostra a redução média percentual no tempo total de identificação de etiquetas ao se utilizar a abordagem bidinâmica proposta. Cada curva revela o ganho obtido com a inclusão do otimizador em relação ao uso de cada estimador de forma isolada. Note que o otimizador consegue melhorar o tempo de identificação com os três estimadores e possui maior impacto com o *Lower Bound* visto que o ganho alcança perto de 9% para 1.000 etiquetas. Embora o *Eom-Lee* seja um estimador acurado, o gráfico mostra que alguma otimização ainda é possível. A Figura 5(b) apresenta o tempo total de identificação de etiquetas utilizando-se apenas as versões originais do *Lower Bound* e do *Eom-Lee*. O desempenho do *Eom-Lee* começa a se diferenciar de forma discreta do desempenho do *Lower Bound* a partir de 100 etiquetas. É importante observar que para diversos quantitativos de etiquetas, o maior custo FLOP total do *Eom-Lee* não implica em uma melhoria significativa no tempo total de identificação de etiquetas. Já para 300 e 1.000 etiquetas, por exemplo, o *Eom-Lee* permite um tempo total de identificação em torno de 6,3% e 8,5% menor, respectivamente. São ganhos consideráveis, mas a contrapartida é um custo de processamento significativo. As Figuras 5(c) e 5(d) apresentam um resultado interessante. Elas comparam o tempo total de identificação de etiquetas utilizando-se somente o *Eom-Lee* original com a versão do *Lower Bound* otimizada pela abordagem bidinâmica proposta. Note que o otimizador permite levar o *Lower Bound*, um estimador inacurado e de baixo custo computacional, a um desempenho similar ao do

Eom-Lee, um estimador acurado mas com custo computacional significativo. E isso trocando processamento por memória no leitor ou servidor associado e adicionando apenas uma multiplicação e uma busca em árvore binária balanceada que possui custo baixo.

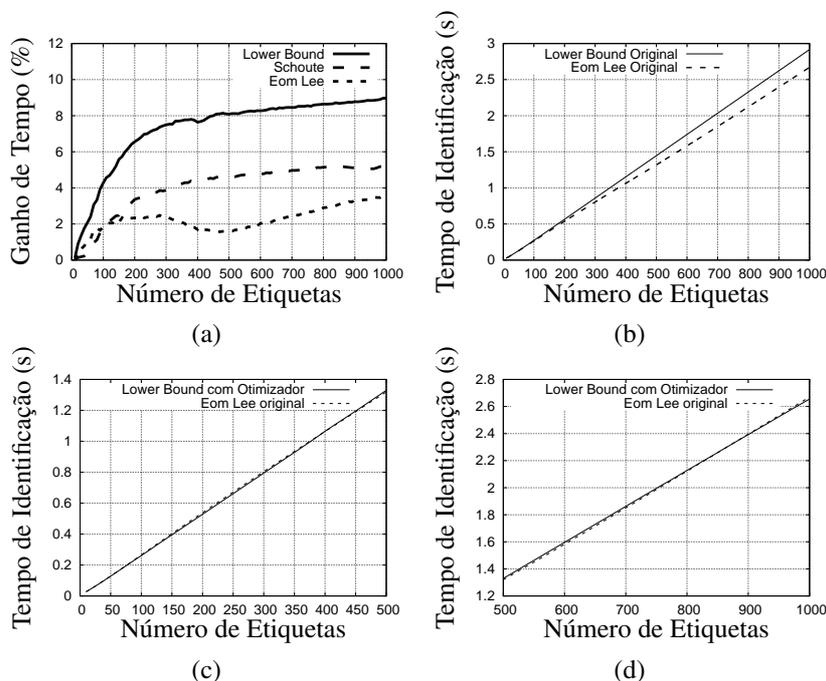


Figura 5. Resultados das Avaliações de Desempenho.

6. Conclusões

Este artigo propôs uma abordagem bidinâmica para a rápida identificação de etiquetas em sistemas RFID baseados no DFSA sem *reset* de quadros. A proposta otimiza dinamicamente a saída do estimador empregado pelo sistema RFID. As principais contribuições deste artigo foram a proposição de um otimizador de estimadores sem *reset* de quadros e a demonstração de que é possível se alcançar um desempenho similar ao do Eom-Lee utilizando apenas o simples *Lower Bound* associado ao otimizador resultante da proposta bidinâmica, mas com custo computacional significativamente menor do que o do Eom-Lee. Para isso, trocou-se processamento por memória e adicionou-se uma multiplicação e uma busca em árvore binária por tamanho de quadro calculado. Os trabalhos futuros incluem 1) o mapeamento do impacto da abordagem proposta com outros estimadores sem *reset*; 2) a extensão da proposta para uso com estimadores com *reset* e os impactos de desempenho e; 3) a extensão da proposta para sistemas baseados no algoritmo-Q e impactos de desempenho.

Referências

- Al-Fuqaha, A., Guizani, M., Aledhari, M., and Ayysah, M. (2015). Internet of things: A Survey on Enabling Technologies, Protocols and Applications. *IEEE Communications Surveys Tutorials*, 17(4):2347 – 2376.
- Andrade, J. D. and Gonçalves, P. A. S. (2011). Uma Função de Cálculo de Tamanho de Frames para o Protocolo DFSA em Sistemas RFID. In *Proc. of XVI Workshop de Gerência e Operação de Redes (WGRS)*, pages 61–74, Campo Grande, MS.

- Andrade, J. D. and Gonçalves, P. A. S. (2013). Um Estimador Acurado para o Protocolo DFSA em Sistemas RFID. In *Proc. of XXXI Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos (SBRC)*, pages 325–338, Brasília, DF.
- Chen, W.-T. (2009). An Accurate Tag Estimate Method for Improving the Performance of an RFID Anticollision Algorithm Based on Dynamic Frame Length ALOHA. *IEEE Transactions on Automation Science and Engineering*, 6(1):9–15.
- Chen, W.-T. (2014). A Feasible and Easy-to-Implement Anticollision Algorithm for the EPCglobal UHF Class-1 Generation-2 RFID protocol. *IEEE Transactions on Automation Science and Engineering*, 11:485–491.
- Eom, J.-B. and Lee, T.-J. (2010). Accurate Tag Estimation for Dynamic Framed-slotted ALOHA in RFID Systems. *IEEE Communications Letters*, 14:60–62.
- EPC Global, I. (2015). *EPC radio-frequency identify protocols Generation-2 UHF RFID Specification for RFID air interface protocol for communications at 860MHz-960 MHz - ratified*, 2.0.1 edition.
- Khanna, N. and Uysal, I. (2015). Q-frame-collision-counter: a novel and dynamic approach to RFID Gen 2's Q algorithm. In *Proc. of the IEEE International Conf. on RFID Technology and Application (RFID-TA)*, pages 120–125.
- Klair, D., Chin, K.-W., and Raad, R. (2010). A Survey and Tutorial of RFID Anti-Collision Protocols. *IEEE Communications Surveys Tutorials*, 12(3):400–421.
- Li, B. and Wang, J. (2011). Efficient Anti-collision Algorithm Utilizing the Capture Effect for ISO 18000-6C Protocol. *IEEE Communications Letters*, 15:352–354.
- Perera, C., Liu, C. H., and Jayawardena, S. (2015). The Emerging Internet of Things Marketplace From an Industrial Perspective: A Survey. *IEEE Transactions on Emerging Topics in Computing*, 3(4):585–598.
- Schoute, F. C. (1983). Dynamic Frame Length ALOHA. *IEEE Transactions on Communications*, 31:565–568.
- Tong, Q., Zou, X., and Tong, H. (2009). Dynamic Framed Slotted ALOHA Algorithm Based on Bayesian Estimation in RFID System. In *Proceedings of the WRI World Congress on Computer Science and Information Engineering*, pages 384–388, Washington, DC, USA. IEEE Computer Society.
- Vales-Afonso, J., Bueno-Delgado, V., Egea-Lopez, E., Gonzales-Castano, F., and Alcaraz, J. (2015). Multiframe maximum-likelihood tag Estimation for RFID Anticollision Protocols. *IEEE Transactions on Industrial Informatics*, 7(3):487–496.
- Vogt, H. (2002). Efficient Object Identification with Passive RFID Tags. In *Proc. of the First International Conf. on Pervasive Computing*, pages 98–113, London, UK.
- Šolić, P., Radić, J., and Rožić, N. (2016). Early Frame Break Policy for ALOHA-Based RFID Systems. *IEEE Trans. on Automation Science and Engineering*, 13(2):876–881.
- Wu, H. and Zeng, Y. (2010). Bayesian tag estimate and optimal frame length for anti-collision aloha rfid systems. *IEEE Transactions on Automation Science and Engineering*, 7(4):963–969.