

Uma Função de Cálculo de Tamanho de *Frames* para o Protocolo DFSA em Sistemas RFID

Júlio D. de Andrade, Paulo André da S. Gonçalves

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

{jda, pasg}@cin.ufpe.br

Abstract. *DFSA (Dynamic Framed Slotted ALOHA) is a popular anti-collision protocol for RFID systems. In such protocol, the size of each frame succeeding the first frame is dynamically calculated based on the estimate of the number of competing tags in the previous frame. The main existing estimators for DFSA seek to increase their accuracy in order to minimize the total identification time of tags. However, such estimators do not consider the impact of the Early-End extension when calculating frame sizes. In this paper, we show that there are missing optimization opportunities in the total tag identification time when DFSA uses such an extension but its impact is ignored in the frame size calculus. Based on this fact, we propose a function to calculate frame sizes but taking into account the estimator's estimate and the impact of the Early-End. Performance evaluations of the proposed function with the Eom-Lee, Vogt, and Schoute estimators show that the total tag identification time is reduced, respectively, up to 24%, 20%, and 29% when compared to that observed by using the estimator itself.*

Resumo. *O DFSA (Dynamic Framed Slotted ALOHA) é um protocolo anticollisão popular para sistemas RFID. Nesse protocolo, o tamanho de cada frame subsequente ao frame inicial é calculado dinamicamente com base na estimativa da população de etiquetas que competiram no frame anterior. As principais propostas de estimadores para o DFSA procuram maximizar a acurácia de suas estimativas na busca da minimização do tempo total de identificação de etiquetas. Contudo, elas desconsideram o impacto do uso da extensão Early-End no cálculo do tamanho dos frames. Este artigo mostra que oportunidades de otimização do tempo total de identificação de etiquetas são perdidas ao se desconsiderar o impacto dessa extensão no cálculo de tamanho de frames quando a mesma é utilizada no DFSA. A partir disso, este artigo propõe uma função de cálculo de tamanho de frames que considera a estimativa do estimador usado e o impacto do uso da extensão Early-End. As avaliações de desempenho da função proposta com os estimadores Eom-Lee, Vogt e Schoute mostram que o tempo total de identificação de etiquetas é reduzido, respectivamente, em até 24%, 20% e 29% quando comparado ao tempo obtido com o uso do estimador isoladamente.*

1. Introdução

Os sistemas RFID (*Radio Frequency IDentification*) são os mais promissores para a identificação automática de objetos através de sinais de radiofrequência. Em geral, os

sistemas RFID mais básicos são compostos por um *leitor* e várias *etiquetas*. Cada etiqueta armazena um identificador (ID) único e é colada ou embutida em um objeto. No processo de identificação, o leitor requisita o ID das etiquetas que se encontram em seu alcance de comunicação. Contudo, é possível que duas ou mais etiquetas transmitam informações ao mesmo tempo ao longo desse processo. Nesse caso, ocorre uma colisão de sinais, impedindo que o leitor reconheça as informações enviadas. Assim sendo, se faz necessária a utilização de um protocolo anticisão de etiquetas a fim de se resolver os conflitos de transmissão e permitir uma rápida identificação de todos os objetos.

O problema em questão é um caso especial do problema de controle de acesso ao meio em redes sem fio, pois introduz um novo desafio conforme explicitado a seguir: as etiquetas RFID geralmente possuem limitações importantes de poder computacional, de memória, de custo e de consumo de energia. Dessa forma, torna-se irreal assumir que elas poderiam trocar mensagens para a reserva do meio de comunicação, ou ainda, assumir que elas poderiam avaliar o uso do canal de comunicação antes de qualquer transmissão a fim de se minimizar as chances de colisão. Portanto, existe a necessidade de se desenvolver e adotar protocolos anticisão de etiquetas específicos para sistemas RFID [Klair et al. 2010].

Dentre as diversas propostas de protocolos anticisão de etiquetas, o protocolo DFSA (*Dynamic Framed Slotted ALOHA*) vem recebendo recentemente grande atenção na literatura [Eom and Lee 2010] [Tong et al. 2009] [Chen 2009]. Nesse protocolo, o leitor organiza o tempo em um ou mais *frames*, onde cada *frame* está subdividido em *slots*. As etiquetas são requisitadas a transmitir em um *slot* a cada *frame* até que sejam identificadas pelo leitor. O número total de *slots* de cada *frame* subsequente ao *frame* inicial é calculado dinamicamente com base na estimativa da população de etiquetas que competiram por *slots* no *frame* precedente. Um dos problemas existentes é como estimar tal população de etiquetas da forma mais precisa possível e relacionar a estimativa obtida em uma função de cálculo de tamanho de *frames*.

Em geral, as propostas de estimadores para o DFSA procuram maximizar a acurácia de suas estimativas na busca da minimização do tempo total de identificação de etiquetas. Contudo, elas desconsideram o impacto do uso da extensão *Early-End* no cálculo do tamanho dos *frames*. Tal extensão é comumente usada em sistemas RFID e permite que a duração de *slots* vazios seja menor do que a duração de *slots* em colisão e bem sucedidos. Este artigo mostra que oportunidades de otimização do tempo total de identificação de etiquetas são perdidas ao se desconsiderar o impacto dessa extensão no cálculo de tamanho de *frames* quando a mesma é utilizada no DFSA. A partir disso, este artigo propõe uma função de cálculo de tamanho de *frames* que considera a estimativa do estimador usado e o impacto do uso da extensão *Early-End*. As avaliações de desempenho da função proposta com os estimadores Eom-Lee [Eom and Lee 2010], Vogt [Vogt 2002] e Schoute [Schoute 1983] mostram que o tempo total de identificação de etiquetas é reduzido, respectivamente, em até 24%, 20% e 29% quando comparado ao tempo obtido com o uso do estimador de forma isolada.

O restante deste artigo está organizado como segue: a Seção 2 apresenta os trabalhos relacionados. O desempenho dos principais estimadores para o DFSA é avaliado e analisado na Seção 3. A Seção 4 apresenta a função de cálculo de tamanho de *frames* proposta neste artigo. As avaliações de desempenho do DFSA com a função proposta são

apresentadas na Seção 5. Finalmente, a Seção 6 apresenta as conclusões deste trabalho.

2. Trabalhos Relacionados

Atualmente, existem diversos estimadores para o DFSA 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] e o Tong [Tong et al. 2009]. Todos, exceto o *Lower Bound*, focam apenas na melhoria da acurácia das estimativas na busca da minimização do tempo total de identificação de etiquetas. Como será apresentado adiante, o objetivo do *Lower Bound* é obter apenas uma estimativa grosseira para análises de pior caso. Este artigo foca no estimador mais recente proposto na literatura – o Eom-Lee – e nos estimadores clássicos *Lower Bound*, Schoute e Vogt. As principais conclusões deste artigo também são aplicáveis aos estimadores Chen e Tong.

Os estimadores estudados neste artigo serão detalhados nas próximas seções. Antes disso, considere a Figura 1, na qual os *frames* de interesse no DFSA estão representados: o *frame* que acabou de ser finalizado (*frame finalizado*) e o seu *frame* imediatamente posterior (*frame posterior*), cujo tamanho precisa ser calculado. No *frame finalizado*, s_v , s_s e s_c representam, respectivamente, a quantidade de *slots* vazios, a quantidade de *slots* bem sucedidos e a quantidade de *slots* em colisão. O tamanho do *frame posterior* é representado por \hat{f} . Para todos os estimadores descritos neste artigo, \hat{n} representa a estimativa do número de etiquetas que competiram no *frame finalizado*.

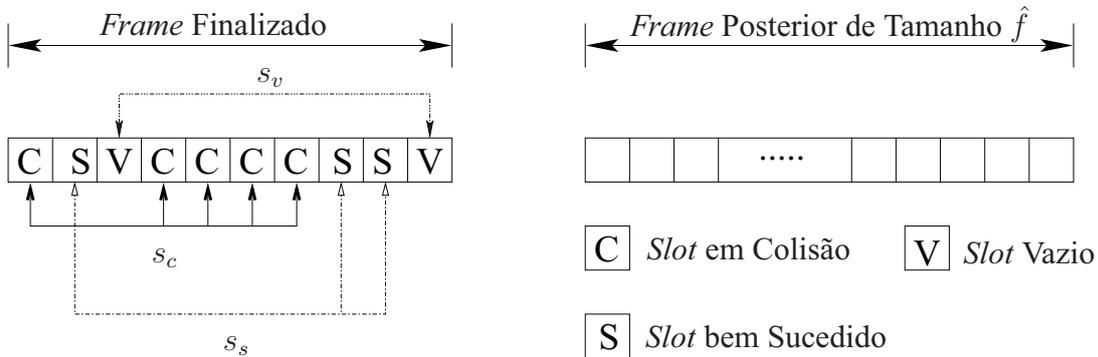


Figura 1. *Frames* de interesse no DFSA.

2.1. Lower Bound

O estimador *Lower Bound* foi inicialmente proposto em [Vogt 2002]. A ideia por trás desse estimador é simples: há ao menos duas etiquetas envolvidas em uma colisão. Então, a menor quantidade possível de etiquetas que competiram em um *frame finalizado* é igual a quantidade total de *slots* com transmissões bem sucedidas somada com o dobro da quantidade de *slots* em colisão. A quantidade total estimada de etiquetas que restam a ser identificadas (*backlog*) é simplesmente o dobro da quantidade de *slots* em colisão. Dessa forma, esse é o valor do tamanho do próximo *frame* a ser utilizado em busca da maximização da eficiência do sistema [Schoute 1983]. Portanto, o estimador *Lower Bound* define:

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

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

Teoricamente, bons estimadores não deveriam ter erros de estimação piores do que os do *Lower Bound*, pois os valores estimados por este último para a quantidade de etiquetas restantes são sempre os menores dentro de uma gama de possibilidades. Por causa disso, o estimador *Lower Bound* é comumente utilizado em avaliações de desempenho do DFSA para fins de comparação com a qualidade de estimação de outros estimadores.

2.2. Schoute

Schoute [Schoute 1983] computa o tamanho do *frame posterior* como sendo o resultado da multiplicação do número total de *slots* em colisão no *frame finalizado* por um fator igual a 2,39. Esse fator é o número esperado de etiquetas que transmitirão em cada *slot* em colisão no *frame posterior*. Assim sendo, o método de Schoute define:

$$\hat{f} = 2,39 \cdot s_c . \quad (3)$$

A Equação (3) é obtida considerando-se um processo de chegadas do tipo Poisson, sendo ela uma aproximação subótima caso o processo de chegada de pacotes respeite outras distribuições. Para se obter a estimativa \hat{n} , basta somar \hat{f} com o número total de etiquetas identificadas com sucesso no *frame finalizado*. Logo,

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

2.3. Vogt

O método de Vogt [Vogt 2002] considera que a alocação de transmissões de etiquetas em um *slot* é um problema de ocupação. Tal problema lida com a alocação aleatória de “bolas” em uma certa quantidade de “sacos”, onde as “bolas” podem ser vistas como etiquetas e os “sacos” como *slots* em um *frame* de interesse. Dado que existem L *slots* em um *frame* e n etiquetas que competiram por *slots* nele, a probabilidade de haver r etiquetas em um *slot* é binomialmente distribuída com parâmetros n e $1/L$ conforme representa a seguinte equação:

$$B_{n, \frac{1}{L}}(r) = \binom{n}{r} \left(\frac{1}{L}\right)^r \left(1 - \frac{1}{L}\right)^{n-r} . \quad (5)$$

Adicionalmente, em um *frame* com L *slots*, a quantidade esperada de *slots* contendo transmissões de r etiquetas é dada por:

$$a_r^{L,n} = LB_{n, \frac{1}{L}}(r) = L \binom{n}{r} \left(\frac{1}{L}\right)^r \left(1 - \frac{1}{L}\right)^{n-r} . \quad (6)$$

O método de Vogt é baseado na desigualdade de Chebyshev, a qual afirma que o resultado de um experimento envolvendo uma variável aleatória X é, provavelmente, próximo ao valor esperado de X [Vogt 2002]. Utilizando esse conceito, Vogt propõe uma função de estimativa do número de etiquetas que busca a minimização da distância entre o

vetor $\langle s_v, s_s, s_c \rangle$ e o vetor contendo os valores esperados para s_v , s_s e s_c . Essa função de estimativa é dada por:

$$\hat{n}(L, s_v, s_s, s_c) = \min_n \left| \begin{pmatrix} a_0^{L,n} \\ a_1^{L,n} \\ a_{\geq 2}^{L,n} \end{pmatrix} - \begin{pmatrix} s_v \\ s_s \\ s_c \end{pmatrix} \right|, \quad (7)$$

onde \hat{n} é o valor de n que minimiza o módulo da diferença dos dois vetores representados nessa equação.

O estimador de Vogt também define uma função de cálculo do tamanho do próximo *frame* com base no número estimado \hat{n} de etiquetas. Os resultados possíveis para a função proposta são apresentados na Tabela 1. Caso $\hat{n} \in [17, 27]$ tanto $\hat{f} = 32$ quanto $\hat{f} = 64$ são escolhas adequadas.

\hat{f}	$\hat{n} \in [x, y]$
16	[1, 9]
32	[10, 27]
64	[17, 56]
128	[51, 129]
256	[112, ∞]

Tabela 1. Tamanho de *frames* para o método de Vogt.

É importante ressaltar que de forma diferente de outras propostas apresentadas, o estimador de Vogt foi desenvolvido considerando-se limitações do sistema RFID *I-Code*. Nesse sistema, o tamanho dos *frames* estão limitados a uma potência de 2 com tamanho máximo possível igual a 256 *slots*. A maioria das pesquisas atuais envolvendo estimadores para o DFSA utilizam o método de Vogt sem tais limitações. Para isso, o tamanho do próximo *frame* no DFSA é calculado simplesmente como:

$$\hat{f} = \hat{n} - s_s. \quad (8)$$

2.4. Eom-Lee

O método Eom-Lee [Eom and Lee 2010] propõe o uso de um algoritmo iterativo para se estimar a quantidade de etiquetas competindo por *slots* em um *frame* e o tamanho \hat{f} de seu próximo *frame*. Primeiramente, Eom-Lee define L como sendo o tamanho do *frame* que será analisado para se estimar o tamanho do *frame* imediatamente posterior. O valor de L é assumido ser igual ao número estimado de etiquetas que competiram no *frame* multiplicado por um fator β que deverá ser determinado. Assim sendo, o valor de L pode ser representado por:

$$L = \beta \cdot \hat{n}. \quad (9)$$

Também é assumido que o número de etiquetas competindo em um *slot* em colisão é igual a γ . Considerando o tamanho do próximo *frame* igual ao *backlog*, o valor de \hat{f} pode ser calculado como:

$$\hat{f} = \hat{n} - s_s = \gamma \cdot s_c. \quad (10)$$

A fim de se obter a estimativa \hat{n} do número de etiquetas, o problema em questão passa a ser a determinação do valor de γ ou de β . Em [Eom and Lee 2010], é demonstrado que γ e β podem ser relacionados através da Equação (11) ao se considerar que a probabilidade de r etiquetas, dentre o universo total de etiquetas, transmitirem em um mesmo *slot* pode ser aproximada por uma distribuição binomial e que a equação de cômputo de tal probabilidade pode ser aproximada por uma distribuição de Poisson com média n/L para L suficientemente grande.

$$\gamma = \frac{1 - e^{-\frac{1}{\beta}}}{\beta(1 - (1 + \frac{1}{\beta})e^{-\frac{1}{\beta}})}. \quad (11)$$

Encontrar uma solução fechada para se determinar os valores de γ e β a partir da Equação (11) é um desafio. Por causa disso, o método Eom-Lee utiliza um algoritmo iterativo para encontrar tais valores. Considere γ_k e β_k sendo, 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:

$$\beta_k = \frac{L}{\gamma_{k-1} \cdot s_c + s_s}, \quad (12)$$

$$\gamma_k = \frac{1 - e^{-\frac{1}{\beta_k}}}{\beta_k(1 - (1 + \frac{1}{\beta_k})e^{-\frac{1}{\beta_k}})}. \quad (13)$$

No primeiro passo, 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 Equações (12) e (13), respectivamente. Quando $|\gamma_{k^*-1} - \gamma_{k^*}|$ for menor que um limiar pré-definido $\epsilon_{threshold}$, o processo iterativo é interrompido. 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 *frame* e a quantidade estimada \hat{n} de etiquetas são obtidos, respectivamente, pelas Equações (14) e (15), onde β_{k^*} é a aproximação mais recente para o valor de β .

$$\hat{f} = \gamma_{k^*} \cdot s_c. \quad (14)$$

$$\hat{n} = \frac{\hat{f}}{\beta_{k^*}}. \quad (15)$$

3. Avaliação e Análise dos Trabalhos Relacionados

Esta seção apresenta uma avaliação independente do desempenho de todos os estimadores apresentados na Seção 2 e analisa os resultados obtidos. Para isso, foi desenvolvido um simulador de eventos discretos na linguagem C++. Nas simulações apresentadas, considera-se um sistema RFID básico com um leitor e uma determinada quantidade de

etiquetas a serem identificadas utilizando-se o protocolo DFSA. Em todas as simulações apresentadas neste artigo, a taxa de transmissão é de 40 kbps e o identificador das etiquetas possui 128 *bits*.

Adicionalmente, as extensões *Early-End* e *Muting* também são utilizadas em todas as simulações neste artigo. A extensão *Muting* permite ao leitor silenciar, até o término do processo de identificação, uma etiqueta identificada com sucesso. Assim sendo, essa etiqueta deixa de competir por *slots*, reduzindo a quantidade de colisões e, por consequência, melhorando a eficiência do protocolo. A extensão *Early-End* permite que o leitor “encurte” a duração de um *slot* vazio. Como o leitor informa às etiquetas não só o início de um *frame* como também o início de cada *slot*, ele pode verificar rapidamente se *bits* de informações começaram a ser recebidos ou não após o envio de um comando de início de *slot*. Ao se passar um tempo pré-determinado após o início de um *slot*, se o leitor não detectar o início da transmissão de ao menos uma etiqueta, o *slot* é encerrado prematuramente. Com isso, o desperdício de tempo gerado por *slots* vazios ao longo do processo de identificação é reduzido.

As métricas de desempenho estudadas foram: o número total de *slots* utilizados no processo de identificação, o tempo total para a identificação de todas as etiquetas, o número total de *slots* vazios, o número total de *slots* em colisão e a média do erro absoluto de estimação por *frame* ao longo do processo de identificação. O erro absoluto de estimação é definido como o módulo da diferença entre o número real e o número estimado de etiquetas em um *frame* de interesse. Para cada estimador, os resultados apresentados foram obtidos a partir da média dos resultados de 2.000 simulações. A fim de se estudar apenas o impacto dos estimadores no desempenho do processo de identificação, considera-se um canal de comunicação livre de erros. Em particular ao estimador Eom-Lee, adota-se o parâmetro $\epsilon_{threshold}$ igual a 0,001. O tamanho do *frame* inicial é de 64 *slots*.

3.1. Modelagem do Canal de Comunicação

Neste trabalho, o canal de comunicação foi modelado a partir da dados fornecidos pela especificação EPCglobal Class-1 Gen-2 [EPC Global 2008]. Essa especificação define as temporizações para os diferentes tipos de *slots* no canal de comunicação e as regras de comunicação entre etiquetas e leitores. O leitor deve se referir à especificação usada para maiores informações sobre como calcular os tempos T_1 a T_5 explicitados nesta seção.

Considere a simbologia $T \Rightarrow R$ como sendo a representação de uma comunicação de etiqueta para leitor e a simbologia $R \Rightarrow T$ como sendo a representação de uma comunicação de leitor para etiqueta. Com base na especificação EPCglobal Class-1 Gen-2, modelou-se o canal de comunicação do seguinte modo: toda comunicação (*i.e.* comandos e respostas) é precedida de um período de sinalização, independente de seu sentido. Na comunicação $R \Rightarrow T$, a sinalização pode ser de dois tipos: *preâmbulo* e *frame-sync*, onde o *preâmbulo* é utilizado apenas quando o leitor envia um comando de início de *frame* e o *frame-sync* é utilizado para qualquer outro tipo de comando. Na comunicação $T \Rightarrow R$, a sinalização é feita através de um *preâmbulo* de tamanho igual a 6 *bits* conforme define a norma utilizada.

Imediatamente após o comando de início de *frame*, o leitor envia um comando de início de *slot*. A duração de um comando de início de um *slot* é igual a T_1 . Este va-

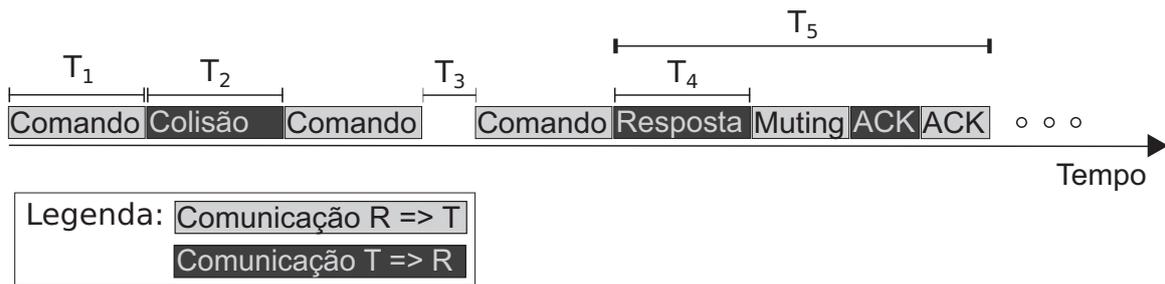


Figura 2. Modelo do canal de comunicação.

lor representa a soma da duração da sinalização com o tempo de transmissão e atraso de propagação. Após o envio desse comando, três situações podem acontecer: a ausência de respostas, respostas múltiplas ou uma única resposta. Um *slot* em colisão possui duração T_2 . Essa duração é composta pelo tempo de propagação somado ao tempo de transmissão da mensagem das etiquetas e à duração do preâmbulo usado pelas mesmas. Caso o leitor transmita um comando de início de *slot*, se ele não começar a receber os *bits* iniciais de ao menos uma resposta, a duração do *slot* é encurtada. Um *slot* de tempo vazio, possui duração igual a $T_3 = \max(RT_{Cal}, 10T_{pri})$, conforme parâmetros definidos na especificação [EPC Global 2008]. A duração de um *slot* bem sucedido é igual a T_4 e considera a soma dos atrasos de transmissão e propagação da mensagem da etiqueta. O tempo T_5 representa o tempo necessário para identificar uma etiqueta e silenciá-la.

A Figura 2 ilustra os três casos de *slots* possíveis em um determinado *frame* e os intervalos de tempo mais importantes. Nessa ilustração, o leitor envia um comando de início de *slot* para as etiquetas das quais as respostas colidem em um *slot*. Em seguida, o leitor envia um novo comando de início de *slot* para o qual não há respostas. Por fim, o leitor envia um comando de início de *slot* para o qual apenas uma etiqueta responde. Após a resposta da etiqueta, o leitor envia o comando *Muting*, a etiqueta responde com um *ACK* e o leitor confirma a recepção desse *ACK* com outro *ACK*. Nesse ponto, a etiqueta é considerada identificada.

3.2. Resultados e Análises

A Figura 3(a) mostra o erro absoluto médio de estimação para cada um dos estimadores quando o número total de etiquetas varia de 100 a 1.000. O erro absoluto médio do *Lower Bound* e do Schoute tem crescimento exponencial lento em função do aumento do número de etiquetas a serem identificadas. Dentre todos os estimadores avaliados, o Eom-Lee é o melhor de todos até pouco mais de 900 etiquetas. Em torno desse valor, o Vogt passa a ser melhor. É importante observar que no intervalo de 100 a 300 etiquetas, o erro absoluto médio do estimador Vogt é muito próximo ao observado para o estimador Eom-Lee. Já entre 300 e 500 etiquetas, o erro absoluto médio do estimador Vogt cresce rapidamente. Contudo, observa-se que a partir de 700 etiquetas até 900 etiquetas, o erro absoluto médio do Vogt volta a se aproximar cada vez mais do erro observado com o uso do estimador Eom-Lee.

A Figura 3(b) mostra o número total médio de *slots* usados em função da quantidade de etiquetas a serem identificadas. Até pouco menos de 900 etiquetas, o estimador Eom-Lee produziu os melhores resultados, embora seja verificado que os demais estimadores tenham produzido resultados próximos para determinadas quantidades de etiquetas.

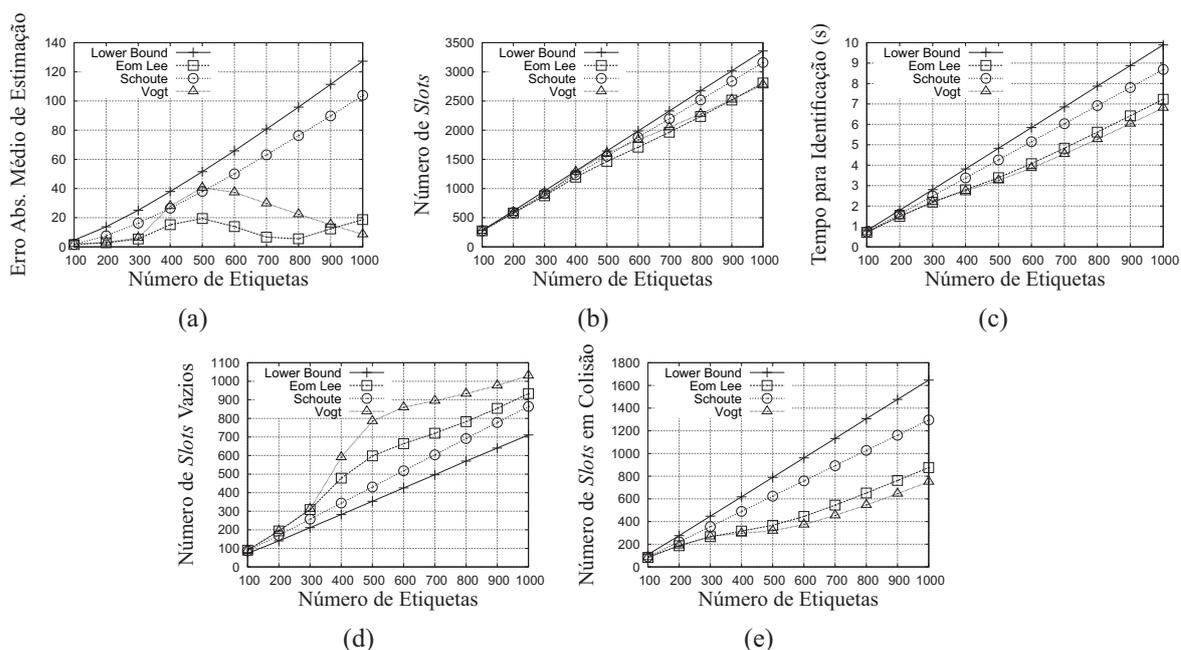


Figura 3. Desempenho dos Estimadores.

Note que a diferença do erro absoluto médio de estimação entre o pior e o melhor estimador não é tão grande, principalmente, até 500 etiquetas. Essa diferença não trouxe impacto significativo no desempenho dos estimadores em relação à quantidade total de *slots* usados ao longo do processo de identificação até 500 etiquetas. A partir de 500 etiquetas, a diferença no erro absoluto médio de estimação entre o pior e o melhor estimador aumenta, impactando, cada vez mais, na quantidade média total de *slots* usados no processo de identificação. Em particular, observa-se que os resultados com o estimador Vogt começam a convergir para os resultados obtidos com o estimador Eom-Lee a partir de 500 etiquetas. Já entre 800 e 1.000 etiquetas, esses dois estimadores possuem desempenho similar em termos do número total médio de *slots* usados.

A Figura 3(c) mostra a média do tempo total de identificação em função do número total de etiquetas. Em particular, observa-se que sob o ponto de vista dessa métrica, os estimadores Vogt e Eom-Lee são os melhores e permitem um tempo de identificação comparável até 400 etiquetas. A partir desse valor, o Vogt leva uma ligeira vantagem sobre o Eom-Lee. Isso é explicado pelo impacto da extensão *Early-End* associado à quantidade de *slots* vazios e em colisão gerada no processo de identificação. Note que o Vogt permite a ocorrência de mais *slots* vazios (Figura 3(d)) e, portanto, a ocorrência de menos colisões (Figura 3(e)) em relação ao Eom-Lee a partir de 300 etiquetas. Note também que o custo “temporal” de um *slot* em colisão é significativamente maior em relação ao custo “temporal” de um *slot* vazio no DFSA com a extensão *Early-End*. Assim, a partir de 400 etiquetas, o Vogt consegue produzir uma melhor relação custo/benefício entre a quantidade de *slots* vazios e em colisão, permitindo um menor tempo total de identificação.

Pelo exposto, algumas conclusões importantes podem ser tiradas quando o DFSA é utilizado em conjunto com a extensão *Early-End*, mas não se considera o impacto da mesma no cálculo do tamanho dos *frames*: 1) uma melhoria na quantidade total de *slots*

utilizados pode não implicar necessariamente em uma redução no tempo total do processo de identificação; 2) buscar apenas um menor erro de estimação não é uma condição suficiente para se minimizar o tempo total de identificação; 3) uma redução adequada na quantidade de *slots* em colisão com um aumento adequado na quantidade de *slots* vazios em cada *frame* gerado contribui para a minimização do tempo total de identificação; 4) superestimar de forma adequada o *backlog* contribui para a minimização do tempo total de identificação; e 5) a métrica de avaliação de desempenho mais importante a ser considerada é o tempo total de identificação de etiquetas.

4. A Função de Cálculo de Tamanho de *Frames* Proposta

De acordo com as conclusões da seção anterior, percebe-se que nenhuma das funções de cálculo de tamanho de *frames*, até então, explora explicitamente o impacto da extensão *Early-End* a fim de se buscar uma minimização do tempo total de identificação de etiquetas. Assim sendo, esta seção apresenta uma proposta de função de cálculo de tamanho de *frames* que explora o impacto dessa extensão. A estratégia proposta consiste em superestimar adequadamente o *backlog* para o cálculo do tamanho de cada novo *frame* gerado, permitindo uma quantidade “razoável” de *slots* vazios e, uma conseqüente redução de *slots* em colisão. Como o tamanho do próximo *frame* no DFSA é proporcional ao *backlog* estimado, é proposto que o tamanho \hat{f} do próximo *frame* calculado pelo estimador seja ajustado por um fator multiplicativo δ_i . Onde i representa o tamanho do *frame* inicial. Assim sendo, é proposta a seguinte função que relaciona a função de cálculo do estimador com esse fator de ajuste:

$$F(\delta_i, \hat{f}) = \delta_i \hat{f} . \quad (16)$$

O valor de δ_i a ser utilizado na Equação (16) precisa ser determinado para cada estimador, buscando-se a minimização do tempo total do processo de identificação de etiquetas. Um estudo sobre o impacto do valor de δ_i no tempo total de identificação de etiquetas é apresentado a seguir na Seção 4.1.

4.1. Determinação do Parâmetro δ_i

Alguns valores adequados de δ_i para uso na função de cálculo proposta foram determinados através de simulações. Nessas simulações, foi avaliado o tempo total de identificação de etiquetas em função de δ_i considerando-se os métodos Eom-Lee, Vogt e Schoute para um *frame* inicial de 64 *slots* e para um *frame* inicial de 128 *slots*. O valor de δ_i foi variado de 1 a 6 em passos de 0,2. Todos os métodos foram avaliados considerando-se um total de 100, 300, 500, 700 e 1.000 etiquetas a serem identificadas. Os resultados apresentados são médias obtidas a partir de 2.000 simulações. Os parâmetros do canal de comunicação são idênticos aos descritos na Seção 3.

As Figuras 4 e 5 mostram a influência do fator de ajuste no tempo total de identificação de etiquetas para um *frame* inicial de 64 *slots* e de 128 *slots*, respectivamente. Em particular, observa-se que a minimização do tempo total de identificação depende do valor do fator de ajuste, o qual depende também da quantidade total de etiquetas a serem identificadas. O problema em questão é encontrar um valor adequado para o fator de ajuste que permita a obtenção de ganhos no tempo total de identificação para qualquer quantidade de etiquetas no intervalo [100, 1.000].

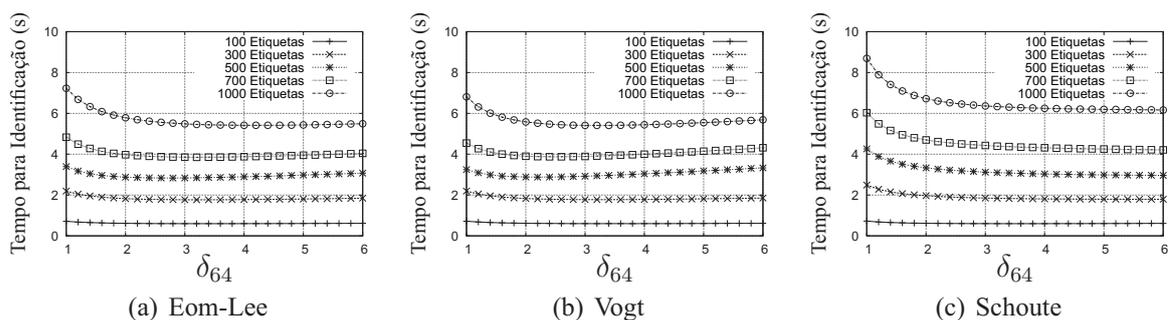


Figura 4. Influência do fator δ no desempenho do DFSA para cada estimador.

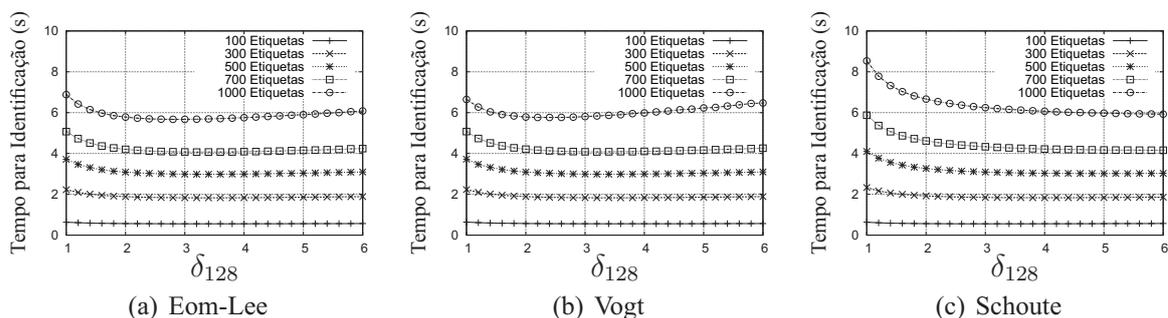


Figura 5. Influência do fator δ no desempenho do DFSA para cada estimador.

No caso do Eom-Lee, o valor de δ_{64} (Figura 4(a)) que minimiza o tempo total de identificação para 300 e 500 etiquetas é igual a 3,0. Esse valor também permite uma redução no tempo total de identificação para o caso de haver 100, 700 ou 1.000 etiquetas a serem identificadas, apesar do tempo total não ser minimizado. Contudo, observa-se que para $\delta_{64} > 3,0$ ou não há ganhos significativos de desempenho ou o desempenho passa a ser pior do que quando $\delta_{64} = 3,0$. No caso do *frame* inicial ser de 128 *slots*, o valor do fator de ajuste (Figura 5(a)) que minimiza o tempo total de identificação para 500 e 700 etiquetas é igual a 3,2. Esse valor também permite uma redução no tempo total de identificação para 100, 300 e 1.000 etiquetas, embora o tempo total de identificação não seja minimizado. Apesar de $\delta_{128} = 3,2$ não ser um valor ótimo para 100, 300 e 1.000 etiquetas, o módulo da diferença do tempo total de identificação com tal valor e o valor do δ_{128} ótimo para tais quantidades de etiquetas é, respectivamente, igual a 0,47 *ms*, 1,1 *ms* e 6,72 *ms*. Logo, a diferença não é significativa. De acordo com as análises feitas, $\delta_{64} = 3,0$ e $\delta_{128} = 3,2$ são valores adequados para o fator de ajuste.

Com relação ao Vogt, o valor de δ_{64} (Figura 4(b)) que minimiza o tempo total de identificação para 100 e 300 etiquetas é igual a 3,2. Com esse valor, o tempo total de identificação também é reduzido quando há 500, 700 ou 1.000 etiquetas a serem identificadas, embora o tempo total de identificação não seja minimizado. O módulo da diferença do tempo total de identificação entre o caso de $\delta_{64} = 3,2$ e os fatores ótimos para as quantidades de etiquetas de 100, 300 e 1.000 é, respectivamente, igual a 63,7 *ms*, 354,66 *ms* e 0,3 *ms*. Mais uma vez, as diferenças não são significativas. No caso do *frame* inicial ser de 128 *slots*, o valor do fator de ajuste (Figura 5(b)) que minimiza o tempo total de identificação para 300 e 700 etiquetas também é igual a 3,2. Esse valor também permite uma redução no tempo total de identificação para o caso de haver 100,

500 ou 1.000 etiquetas a serem identificadas, mesmo não minimizando o tempo total. O módulo da diferença do tempo total de identificação obtido ao se utilizar $\delta_{128} = 3,2$ e os fatores ótimos para quantidades de etiquetas de 100, 500 e 1.000 é, respectivamente, igual a $0,472 \text{ ms}$, $1,062 \text{ ms}$ e $75,018 \text{ ms}$. Com base nas análises apresentadas, $\delta_{64} = 3,2$ e $\delta_{128} = 3,2$ são valores adequados para o fator de ajuste.

No caso do Schoute, $\delta_{64} = 6,0$ (Figura 4(c)) produz o menor tempo total de identificação para 300, 500, 700 e 1.000 etiquetas. Para uma quantidade de etiquetas igual a 100, o valor de δ_{64} que minimiza o tempo total de identificação é igual a 3,6. Contudo, para essa quantidade de etiquetas, um $\delta_{64} = 6,0$ ainda produz ganhos de desempenho em relação ao caso de se utilizar um valor de $\delta_{64} = 1,0$. No caso do *frame* inicial ser de 128 *slots*, $\delta_{128} = 5,8$ minimiza o tempo total de identificação para 700 etiquetas. O módulo da diferença do tempo total de identificação entre o caso de $\delta_{128} = 5,8$ e os fatores ótimos para 100, 300, 500 e 1.000 etiquetas é, respectivamente, igual a $14,591 \text{ ms}$, $23,405 \text{ ms}$, $5,996 \text{ ms}$ e $8,313 \text{ ms}$. Mais uma vez, a diferença de tempo observada não é significativa. Com base no exposto, $\delta_{64} = 6,0$ e $\delta_{128} = 5,8$ são valores adequados para o fator de ajuste.

5. Avaliação de Desempenho com a Função Proposta

Esta seção apresenta uma avaliação de desempenho do DFSA com e sem o uso da função de cálculo proposta. Mais uma vez são empregadas as extensões *Early-End* e *Mutting*. As métricas de desempenho avaliadas em função do número total de etiquetas foram: o número total de *slots* vazios, o número total de *slots* em colisão e o tempo total para a identificação das etiquetas. Os parâmetros do canal de comunicação são idênticos aos descritos na Seção 3. Os resultados apresentados são médias obtidas a partir de 2.000 simulações.

As Figuras 6, 7 e 8 mostram os resultados obtidos para um *frame* inicial de 64 *slots* utilizando-se os estimadores Eom-Lee, Vogt e Schoute, respectivamente. Note que para todos os casos avaliados, a função proposta permite uma melhoria no tempo total de identificação de etiquetas já que ela explora melhor o impacto da extensão *Early-End*. Tal melhoria em relação ao Eom-Lee, ao Vogt e ao Schoute é, respectivamente, de até 24%, 20% e 29%. A correta exploração da extensão se traduz em um aumento adequado da quantidade de *slots* vazios e, em consequência, em uma redução adequada da quantidade de *slots* em colisão por *frame* estimado. Note também que embora a quantidade total de *slots* utilizados no processo de identificação aumente, não há impacto negativo no tempo total de identificação das etiquetas. Os resultados obtidos para um *frame* inicial de 128

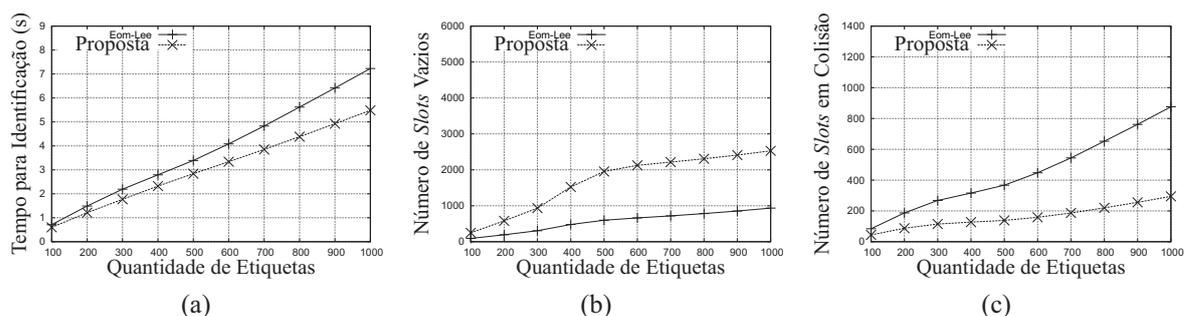


Figura 6. Estimador Eom-Lee e *frame* inicial de 64 *slots*.

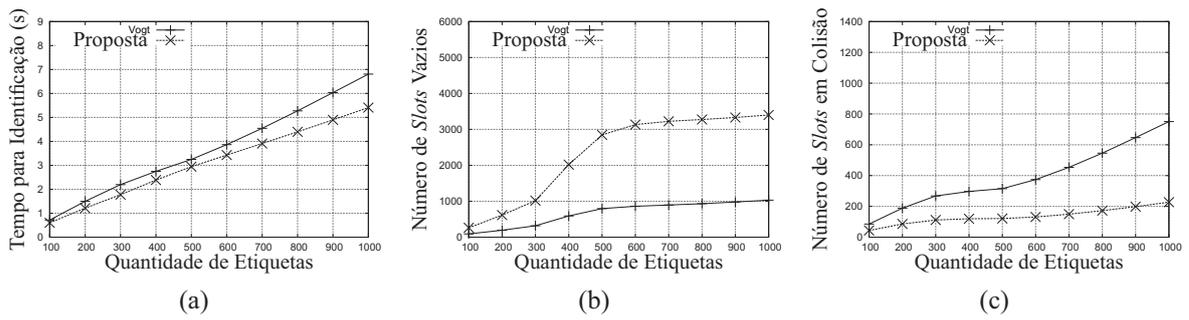


Figura 7. Estimador Vogt e *frame* inicial de 64 slots.

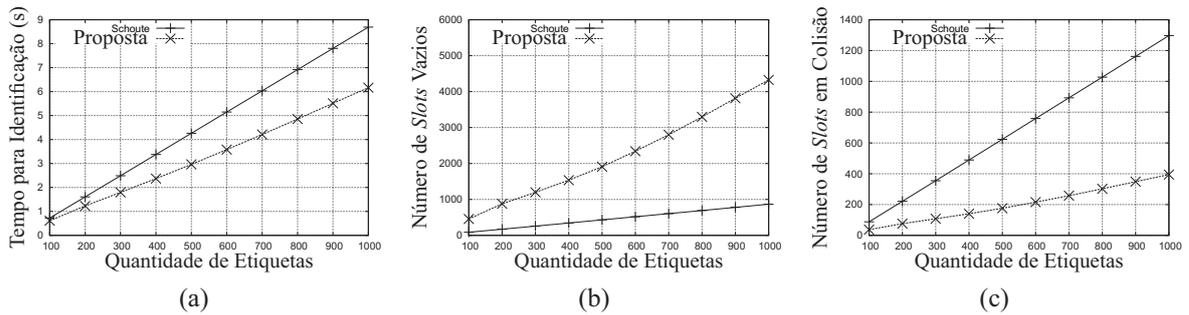


Figura 8. Estimador Schoute e *frame* inicial de 64 slots.

slots foram omitidos por limitações de espaço, porém a abordagem proposta continua trazendo melhorias significativas no tempo total de identificação de etiquetas.

Os fatores de ajuste encontrados na Seção 4.1 para cada estimador não diferiram significativamente ao se mudar o tamanho do *frame* inicial de 64 para 128 slots. Além disso, os valores de δ_{64} e δ_{128} encontrados são subótimos para várias quantidades de etiquetas. Mesmo assim, o tempo total de identificação ao se utilizar um fator ótimo não se mostrou significativamente diferente do tempo obtido com os fatores subótimos indicados. Tudo isso sugere que, para ambos os tamanhos de *frame* inicial estudados, um mesmo fator de ajuste pode ser utilizado sem prejuízo perceptível no tempo total de identificação. A avaliação dessa hipótese é apresentada na Figura 9. Ela mostra o tempo total de identificação para o Eom-Lee e para o Schoute com a função proposta quando se considera um *frame* inicial igual a 128 slots e os fatores de ajuste δ_{64} e δ_{128} . Note que não há diferenças perceptíveis no tempo total de identificação.

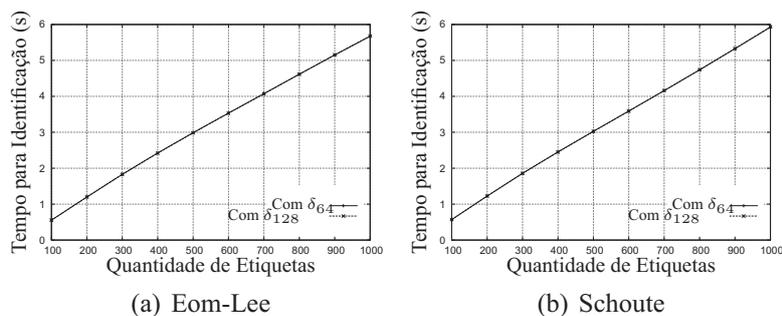


Figura 9. Impacto dos fatores de ajuste para um *frame* inicial de 128 slots.

6. Conclusões

Esse artigo apresentou um estudo do protocolo DFSA com os principais estimadores de tamanho de *frame* propostos na literatura. Dentre as principais contribuições encontram-se conclusões importantes tiradas ao se analisar o DFSA com a extensão *Early-End*, mas sem considerar o impacto da mesma no cálculo do tamanho dos *frames*: 1) uma melhoria na quantidade total de *slots* utilizados pode não implicar necessariamente em uma redução no tempo total do processo de identificação; 2) um menor erro de estimação não é uma condição suficiente para se minimizar o tempo total de identificação; 3) uma redução adequada na quantidade de *slots* em colisão com um aumento adequado na quantidade de *slots* vazios em cada *frame* gerado contribui para a minimização do tempo total de identificação; e 4) superestimar de forma adequada o *backlog* contribui para a minimização do tempo total de identificação.

A partir dessas conclusões, foi apresentada mais uma contribuição, sendo ela a proposta de uma função de cálculo de tamanho de *frames* para o DFSA que explora o impacto da extensão *Early-End*. A função proposta relacionou a função de cálculo de um estimador escolhido com um fator de ajuste. Valores adequados para esse fator foram determinados de acordo com os estimadores e com os tamanhos de *frame* inicial estudados. Os resultados mostraram que a abordagem proposta permitiu que o tempo total de identificação de etiquetas no DFSA com *Early-End* fosse melhorado quando comparado com o tempo total obtido ao se utilizar a função original do estimador isoladamente. Em particular, a melhora observada no tempo total de identificação de etiquetas no caso de uso da função proposta em conjunto com o Eom-Lee, o Vogt e o Schoute foi de até 24%, 20% e 29%, respectivamente.

Referências

- 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.
- 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. (2008). *EPC Radio-Frequency Identity Protocols Class-1 Generation 2 UHF RFID Protocol for Communications at 860 MHz - 960 MHz*. Version 1.2.0.
- 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.
- 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 *Proc. of the WRI World Congress on Computer Science and Information Engineering*, pages 384–388.
- Vogt, H. (2002). Efficient Object Identification with Passive RFID Tags. In *Proceedings of the First International Conference on Pervasive Computing*, pages 98–113, London, UK. Springer-Verlag.