



Universidade Federal de Pernambuco

Centro de Informática

Graduação em Ciência da Computação

**UMA ANÁLISE DA UTILIZAÇÃO DA TÉCNICA DE  
DECOMPOSIÇÃO DE NÚMEROS EM UM ESTIMADOR  
BAYESIANO PARA SISTEMAS RFID**

Renato Moura Dantas

TRABALHO DE GRADUAÇÃO

Recife  
10 de novembro de 2014

Universidade Federal de Pernambuco

Centro de Informática

Renato Moura Dantas

**UMA ANÁLISE DA UTILIZAÇÃO DA TÉCNICA DE  
DECOMPOSIÇÃO DE NÚMEROS EM UM ESTIMADOR  
BAYESIANO PARA SISTEMAS RFID**

Trabalho apresentado ao Programa de Graduação em Ciência da Computação do Centro de Informática da Universidade Federal de Pernambuco como requisito parcial para a obtenção do grau de Bacharel em Ciência da Computação.

Orientador: Paulo André da Silva Gonçalves

Recife

10 de novembro de 2014

## RESUMO

Ao longo das últimas décadas, os sistemas de identificação única por rádio frequência, ou RFID, vem se popularizando e ganhando força, tanto na academia, quanto no mercado. Diversos pesquisadores consideram o RFID o sucessor imediato do código que barras e uma ponte para o que se chama de internet das coisas(*internet of things*).

A utilização completamente difundida da tecnologia RFID limita-se à aquisição de uma boa taxa de colisões entre as etiquetas e o barateamento do custo das mesmas.

Neste trabalho, fez-se uma análise da utilização da técnica de decomposição de números em fatores primos e de uma abordagem própria para evitar o cálculo de fatoriais em estimadores bayesianos utilizados no protocolo DFSA em sistemas RFID. O algoritmo utilizado como base para os estudos é o Bayesiano[1]. Para realizar os experimentos foi desenvolvida uma ferramenta capaz de simular a execução desse algoritmo em um sistema RFID típico e coletar informações relevantes para análise. Por fim, faz-se uma análise dos dados obtidos, exaltando as vantagens e desvantagens de utilizar a decomposição ou a abordagem proposta de combinação, nesse cenário.

**Palavras-chave:** RFID, Protocolos anticolisão, DFSA, Bayes

## **ABSTRACT**

During the past decades, the Radio-frequency identification (RFID) has becoming more popular among students in universities so as in industries. Several researchers believe that the RFID is the successor of bar-code technology and a strong link to another paradigm of technology called internet of things.

This research presents an analyses of prime factorization and a new method to avoid factorial calculations, applied in Bayesian estimators used in DFSA protocol in RFID systems. The bayesian algorithm consist the base of this research. In order to explore more efficiently the experiments, the research shows several tools capable of simulate these algorithms in RFID environments and then collect relevant data for analyses proposes. At last, the data is processed and reveals the advantages and drawbacks of prime factorization and the combination method, for each simulation.

**Keywords:** Radio-frequency identification(RFID), Anticollision, DFSA, Bayes, Tag estimate

# SUMÁRIO

<b>CAPÍTULO 1 – INTRODUÇÃO</b> .....	<b>1</b>
1.1 Motivação .....	1
1.2 Objetivos .....	2
1.3 Organização .....	3
<b>CAPÍTULO 2 – FUNDAMENTOS DE RFID</b> .....	<b>4</b>
2.1 Funcionamento .....	4
2.1.1 Etiquetas Passivas .....	6
2.1.2 Etiquetas Semiativas .....	6
2.1.3 Etiquetas Ativas .....	7
2.2 Padronização .....	7
2.2.1 ISO .....	8
2.2.2 EPC Global.....	8
2.3 Aplicações .....	8
2.3.1 Pedágios .....	9
2.3.2 Controle de acesso .....	9
2.3.3 Linha de montagem industrial .....	10
2.4 Dificuldades da tecnologia RFID .....	10
<b>CAPÍTULO 3 – PROTOCOLOS ANTICOLISÃO</b> .....	<b>11</b>
3.1 Colisões .....	11
3.2 Protocolos anticolisão .....	11
3.2.1 Protocolos baseados em ALOHA .....	12
3.2.1.1 ALOHA puro .....	12
3.2.1.2 Slotted-ALOHA .....	13
3.2.1.3 Framed Slotted-ALOHA .....	14
3.2.1.4 Dynamic Framed Slotted-ALOHA .....	14
3.3 Estimadores para o DFSA .....	15
3.3.1 Lower Bound .....	16
3.3.2 Vogt .....	16
3.3.3 Chen .....	18
3.3.4 Bayesian .....	19
3.4 Melhorias propostas .....	21
3.4.1 Método da Decomposição .....	22
3.4.2 Método da Combinação .....	23
3.5 Resumo .....	24
<b>CAPÍTULO 4 – ANÁLISES E RESULTADOS</b> .....	<b>25</b>
4.1 Simulador .....	25
4.2 Validação do Simulador .....	26
4.3 Resultados e análises das abordagens propostas .....	27
4.4 Dificuldades .....	32
4.5 Resumo .....	32
<b>CAPÍTULO 5 – CONCLUSÕES</b> .....	<b>33</b>

# CAPÍTULO 1

## INTRODUÇÃO

Sistemas RFID estão cada vez mais difundidos entre usuários que necessitam de identificação única de objetos. Embora utilizado com grande frequência por grandes empresas de mercado, a tecnologia RFID ainda tem muitas lacunas de conhecimento e várias inovações vem sendo propostas nesse cenário. Segundo alguns visionários de mercado e negócios, os sistemas RFID prometem revolucionar a forma como objetos são vistos e como interagem atualmente com o homem.

### 1.1 Motivação

A tecnologia RFID (*Radio Frequency Identification* – Identificação por Rádio Frequência) tem como principal finalidade a identificação, controle e automatização de objetos ou pessoas.

Em geral, um sistema RFID típico é composto por uma ou mais etiquetas e um leitor. O leitor tem como função ler os dados e identificar unicamente cada etiqueta em seu alcance. As etiquetas possuem um identificador único que é lido pelo leitor e identifica unicamente o objeto ao qual está anexada a etiqueta. A partir dessa identificação, diversas informações podem ser adquiridas sobre o objeto.

Os sistemas RFID tem capacidade de se adequar a várias atividades cotidianas e estão sendo aos poucos incorporados no mercado. Seu uso envolve, por exemplo, o controle do estoque de uma grande cadeia de supermercados, o gerenciamento e a organização de livros em grandes bibliotecas, o controle de estoque e acesso a medicamentos em hospitais, rastreamento de animais em grandes fazendas, entre diversas outras.

Essa tecnologia tem potencial para substituir completamente o código de barras, atual forma de identificação de produtos em supermercados. Dentre as vantagens da tecnologia RFID, podemos citar:

- Não se faz necessária uma linha de visão entre o leitor e a etiqueta, como no sistema tradicional de código de barras.
- Várias etiquetas podem ser identificadas e lidas em um único ciclo de leitura.
- Etiquetas RFID tem a capacidade de armazenar mais informações além do identificador.

Possuir tantas facilidades exigem um custo extra, o que torna o RFID um sistema não completamente difundido atualmente.

Vários tipos de pesquisa vem sendo desenvolvida nos últimos anos, tanto com foco mercadológico, como com foco acadêmico. A tecnologia de identificação por rádio frequência tem se tornado foco de estudo nas maiores universidades do mundo e tem ganhado destaque como grande responsável para chegada da internet das coisas.

No processo de leitura das etiquetas, duas ou mais etiquetas podem transferir seus dados simultaneamente ou em tempos parcialmente iguais. A esse fenômeno damos o nome de colisão, e esse é um dos maiores problemas em aberto dos sistemas RFID. Com o intuito de controlar ou erradicar as colisões, foram escritos e propostos diversos algoritmos e métodos que tem como principal atividade evitar a ocorrência de colisões nas fases de leitura de sistemas RFID.

Entre os protocolos mais conhecidos e estudados no momento encontra-se o DFSA, ou *Dynamic Framed Slotted Aloha*, protocolo baseado em *Aloha* que é visto como grande proposta para solucionar o problema das colisões atualmente. Em suma, esse protocolo funciona dividindo o tempo de transmissão em frames, cada frame se divide em slots e cada etiqueta só transmite uma vez a cada frame em um slot escolhido randomicamente.

Para complementar o funcionamento do DFSA, existem algoritmos chamados estimadores, cuja função é estimar a quantidade de etiquetas que concorreram por slots no último frame a cada etapa do processo de leitura das mesmas.

Dentre os estimadores propostos mais conhecidos na literatura, podemos citar: Bayesian[1], o Chen[2], Vogt[3] e o Lower Bound também descrito por Vogt. O algoritmo [1] será descrito com mais detalhes ao longo do texto, além de ser usado como fonte de experimentos no objetivo principal desse trabalho.

## 1.2 Objetivos

O objetivo geral desse trabalho é realizar uma análise sobre o uso da decomposição em fatores primos e uma outra abordagem proposta para eliminar fatores infinitos no cálculo de probabilidades condicionais em estimadores que utilizam uma estimativa bayesiana. Como exemplo de estimador, utiliza-se o Bayesian[1] ao longo de todo o texto, mas a problemática se aplica também ao Chen[2], por exemplo.

Para realizar essa análise, foi implementada uma ferramenta capaz de simular o comportamento de um sistema RFID típico, mas com a possibilidade de alterar o estimador, e assim, capturar e analisar as diferenças entre o uso e a ausência da técnica de decomposição.

Para atingir os objetivos gerais desse projeto, foram definidos os seguintes objetivos específicos:

- Estudar e entender o funcionamento de sistemas RFID;
- Estudar e entender os protocolos anticolisão, focando o conhecimento no DFSA;
- Estudar de forma completa os estimadores citados anteriormente nesse trabalho;
- Desenvolver um simulador capaz de reproduzir os estimadores e coletar resultados das simulações.
- Comparar o desempenho do algoritmo Bayesian[1], seguindo algumas métricas comuns, fazendo uso da decomposição e sem o uso da mesma.

### **1.3 Organização**

Este trabalho está dividido em 5 capítulos. O segundo capítulo descreve em mais detalhes os sistemas e a tecnologia RFID, abordando os seus principais elementos, suas aplicações, etc. O terceiro capítulo aborda os protocolos anticolisão, demonstrando sua necessidade e sua evolução ao longo dos anos, com foco no DFSA. É nesse capítulo que são descritos os estimadores que são citados e utilizados nesse trabalho. O quarto capítulo descreve o processo seguido para a execução das simulações e analisa os resultados obtidos para avaliar o uso da técnica. O quinto capítulo mostra as conclusões desse trabalho.

## CAPÍTULO 2

### FUNDAMENTOS DE RFID

A tecnologia RFID vem crescendo bastante desde sua invenção na década de 50. Embora tenha sido criada em 50, somente na década de 80 a tecnologia começou a ser utilizada comercialmente[14]. O barateamento e resolução do problema de leitura de diversas etiquetas simultaneamente são fatores que ainda dificultam a inserção completa do RFID. Esse capítulo mostra os conceitos básicos dessa tecnologia, padrões existentes, além de exemplos de utilização.

#### 2.1 Funcionamento

Os sistemas RFID utilizam a comunicação através de ondas de rádio para transmitir informações contidas nas etiquetas em um meio sem fio. Essa transmissão, em geral, inicia no leitor, capaz de enviar sinais eletromagnéticos para as etiquetas, que, ao receberem o sinal do leitor, transmitem suas informações para o mesmo.

A tecnologia RFID é semelhante nesse processo ao código de barras, porem o código de barras utiliza a luz como meio e os sistemas RFID utilizam sinais de rádio frequência.

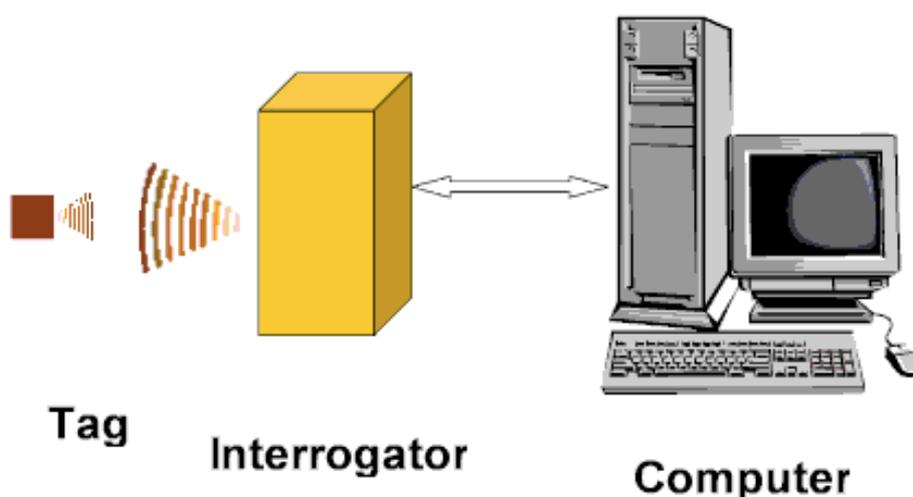


Figura 1. Principais elementos de um sistema RFID

Um sistema RFID típico tem três principais elementos. São eles:

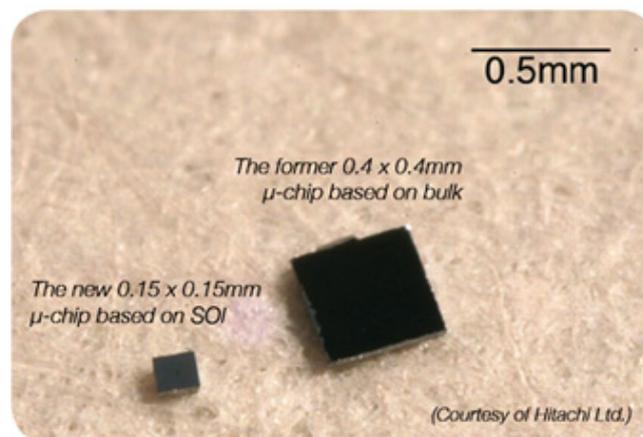
- O leitor;

- As etiquetas;
- Subsistema de processamento.

Os leitores são equipamentos construídos, em geral, por um módulo capaz de enviar e receber sinais de rádio frequência e um sistema de controle. Sua principal função nos sistemas RFID é garantir a identificação única de cada etiqueta em seu alcance.

O subsistema de processamento nada mais é que a base de informações de um sistema RFID. Em geral, sua função consiste em identificar quais dados que estão sendo lidos nas etiquetas são relevantes para o sistema em uso. Geralmente, o subsistema de processamento está contido e acoplado no leitor.

As etiquetas são os elementos que carregam o identificador único do objeto a ser identificado. Consistem, em geral, de um microchip combinado com uma antena. A antena da etiqueta capta os sinais provenientes do leitor, respondendo com seu identificador logo em seguida. O tamanho de uma etiqueta varia com a complexidade da mesma. Atualmente chips minúsculos estão sendo desenvolvidos pela *Hitachi* [9], empresa que dedica-se bastante na evolução da tecnologia RFID. A figura abaixo ilustra as dimensões de um modelo chamado Mu-chip, ou  $\mu$ -chip.



**Figura 2. Uma das menores etiquetas do mundo, da Hitachi**

As etiquetas RFID são normalmente classificadas em três categorias, de acordo com sua complexidade:

- Passivas;
- Semi-passivas;
- Ativas.

### 2.1.1 Etiquetas Passivas

As etiquetas passivas não possuem fonte de alimentação própria. Para conseguir transmitir seus dados, elas utilizam os sinais emitidos pelo leitor, transformando o sinal de rádio em energia, através de indução eletromagnética que ocorre na antena.

Esse tipo de etiqueta, por não possuir fonte de alimentação própria, necessita da presença do leitor para começar a transmitir. Em geral, o leitor inicia a comunicação, cedendo assim, energia para as etiquetas passivas.

As etiquetas passivas são as mais baratas e mais utilizadas no cenário atual. Essa redução de custo implica diretamente no poder computacional da etiqueta, fazendo com que sua estrutura seja simplificada para manter o baixo custo. O alcance das etiquetas passivas é relativamente pequeno, geralmente conseguindo transmitir somente até poucos metros do leitor.

Em suma, as vantagens desse tipo de etiqueta é sua leveza, seu custo reduzido e seu tamanho reduzido. A figura abaixo mostra um exemplo de etiqueta passiva.

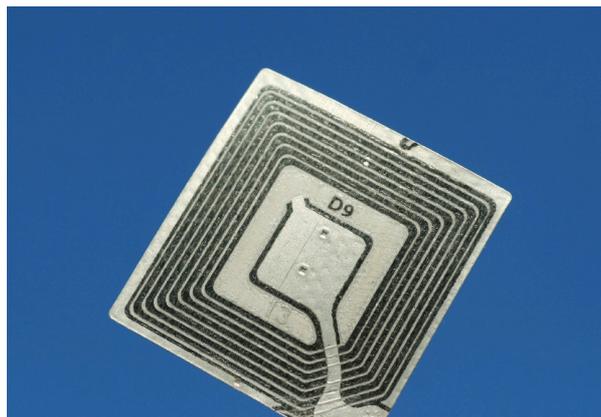


Figura 3. Etiqueta RFID passiva

### 2.1.2 Etiquetas semi-passivas

Etiquetas semi-passivas são aquelas que possuem uma fonte de alimentação própria(bateria), mas utilizam o mesmo somente para energizar o circuito interno. A forma de comunicação com o leitor é idêntica a forma como as etiquetas passivas realizam a comunicação, isto é, toda a energia necessária para a transmissão é obtida através de indução eletromagnética.

Embora a forma de transmissão seja idêntica, as etiquetas semi-passivas conseguem transmitir seus sinais a maiores distâncias. Esse fato acontece pois toda a energia conseguida através da indução eletromagnética é utilizada para a transmissão, visto que a bateria interna é utilizada para energizar o circuito interno.

### 2.1.3 Etiquetas ativas

As etiquetas ativas são etiquetas que possuem uma fonte de alimentação própria(bateria) e essa bateria é utilizada tanto para energizar o circuito interno, como fazem as etiquetas semiativas, mas também é utilizada para a comunicação com o leitor.

Devido a essa bateria mais poderosa, as etiquetas ativas tem um poder computacional mais desenvolvido, conseguindo transmitir seus dados por várias centenas de metros. Novamente devido a bateria que elas possuem, essas etiquetas conseguem enviar dados antes mesmo do leitor iniciar a comunicação, pois elas não necessitam da energia capturada do sinal do leitor.

Geralmente as etiquetas ativas são de leitura e escrita, isto é, podem ser inseridos dados nas mesmas com facilidade.

Devido ao uso exclusivo da bateria como fonte de energia, a vida útil desse tipo de etiqueta é curta, pois ao acabar a bateria nenhuma comunicação pode ser estabelecida e a etiqueta é inutilizada. A figura abaixo ilustra uma etiqueta ativa e todos os seus circuitos.

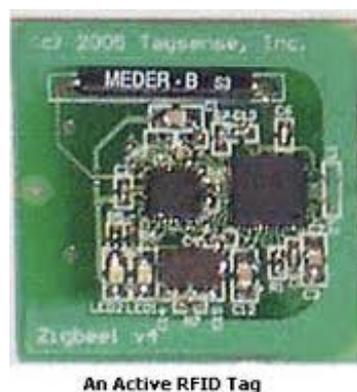


Figura 4. Circuito de etiqueta RFID ativa

## 2.2 Padronização

Padrões são essenciais em qualquer área de pesquisa com o intuito de facilitar a comunicação e interação entre os equipamentos de uma mesma tecnologia em escala global.

Grandes esforços vem sendo dispendidos na tentativa de padronizar por completo a utilização de sistemas RFID em todo o mundo. Dentre as organizações que estão nessa força tarefa, destacam-se a ISO(International Standard Organization), mundialmente conhecida por delimitar vários dos padrões mais conhecidos, e a EPCGlobal [13].

Os padrões para os sistemas RFID ainda estão sendo desenvolvidos, mas eles já estão sendo pensados para cobrir as áreas chave da tecnologia, como por exemplo:

- Padrões relacionados a codificação utilizada pelas etiquetas quando armazenam os dados;

- Protocolos de interface aérea, que definem a comunicação entre etiquetas e leitores;
- Padrões para testes e segurança da tecnologia.

### **2.2.1 ISO**

A ISO é uma das mais conhecidas entidades de regulamentação de padrões atualmente. Ela está presente desde a criação de padrões industriais quanto comerciais em quase todos os itens que utilizamos no dia a dia. Em sua série 18000, a ISO se envolveu na padronização da tecnologia RFID, embora essa padronização ainda esteja em desenvolvimento [13].

A grande maioria das regras e normas citadas pela ISO se referem a comunicação em meios sem fio, protocolos de segurança na tecnologia, isto é, fatores bem genéricos, mas que contribuem para que a tecnologia tenha potencial de crescimento sem ter que se limitar por normas e padrões.

Dentre os itens da série 18000, podemos citar:

- 18000-2: Interface aérea para 135 KHz;
- 18000-3: Interface aérea para 13.56 MHz;
- 18000-4: Interface aérea para 2.45GHz.

Como pode-se observar, são regras que mais dizem respeito ao meio de comunicação que à tecnologia em si.

### **2.2.2 EPC Global**

A EPCGlobal nasceu de um esforço conjunto de diversas empresas e o MIT(Massachussets Institute of Technology). A primeira proposta da EPCGlobal, o EPC(Eletronic Product Code) Gen 1, foi um padrão que teve uma aceitação altíssima, porém divergia em diversos pontos das normas da ISO. O EPC nada mais é que um código único embutido nas etiquetas como forma de controle por parte da EPCGlobal.

Em 2004, foi proposta a segunda geração do EPC que, através das críticas que tinham sofrido na primeira geração, se aproximou das normas da ISO, conseguindo assim, se tornar um padrão amplamente utilizado no mercado e na academia [13].

## **2.3 Aplicações**

Várias empresas de mercado já enxergaram o potencial da tecnologia RFID e fazem uso das facilidades que essa tecnologia pode trazer. As aplicações não são de um nicho específico, podendo ser observadas em quase todos os tipos de negócios, desde o controle de materiais em um hospital, ao rastreamento de animais em grandes fazendas.

Nessa seção serão citadas algumas das aplicações mais comuns, no cenário atual, utilizando RFID.

### 2.3.1 Pedágios

A tecnologia RFID já se tornou parte integrante da grande maioria das aplicações de pedágio no mundo todo. O uso da etiqueta é feita através de um adesivo afixado no vidro do carro. Ao passar pelo pedágio, o motorista não precisa parar para pagar o pedágio na hora, pois um leitor RFID captura informações da etiqueta e consegue com isso, identificar o carro e fazer a cobrança de uma forma menos intrusiva.

A figura abaixo ilustra o comportamento dessa utilização do RFID.

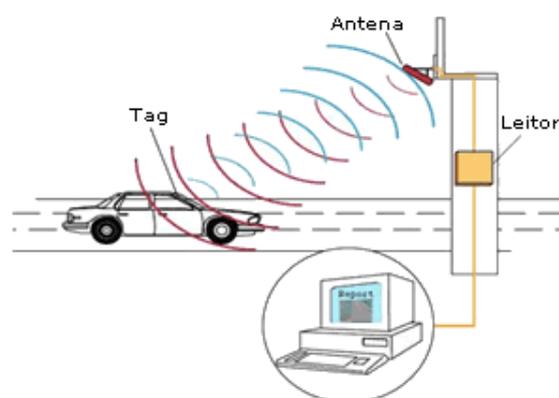


Figura 5. Pedágio utilizando RFID

### 2.3.2 Controle de acesso

Várias empresas já utilizam o controle de acesso através de RFID. Até pouco tempo atrás, na grande maioria dos cartões de acesso a grandes empresas era utilizada a tecnologia de código de barras, porém essa tecnologia introduz muito risco pois pode-se, por exemplo, tirar fotocópia do código de barras e outra pessoa poderia ter acesso ao prédio.

Com a inserção da tecnologia RFID, ficou bastante mais seguro o controle de acesso a ambientes restritos. Outra facilidade é que, diferente do código de barras, que se desgasta com o uso contínuo, a tecnologia RFID com etiquetas passivas tem um tempo de vida útil altíssimo, diminuindo bastante a necessidade de trocas periódicas de cartão.

A figura abaixo ilustra o que foi descrito acima.

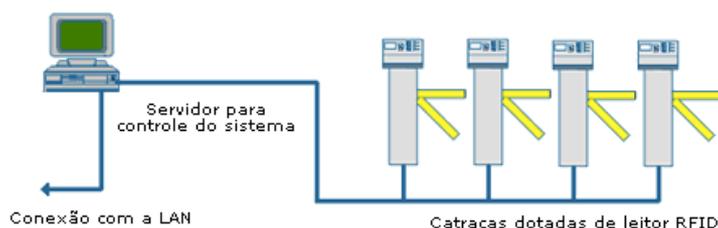


Figura 6. Controle de acesso via cartões RFID

### **2.3.3 Linha de montagem industrial**

Nessa aplicação da tecnologia RFID, a grande vantagem estabelecida com a inserção da tecnologia é a capacidade de monitorar todas as peças envolvidas na construção do carro até o produto final. Além de ser utilizada na fabricação, essas etiquetas poderiam ser usadas na manutenção dos mesmos, guardando um registro de cada vez que certa peça foi trocada ou consertada, exibindo assim, possíveis fragilidades em certos formatos de peças.

Uma outra funcionalidade que essas etiquetas poderiam ter em veículos seria o controle de peças durante um furto, podendo, por exemplo, desligar a passagem de gasolina para o motor, ou travar todas as portas do carro. Um exemplo de uso dessa tecnologia é a *Hyundai Motor Company*, que já utiliza RFID na fabricação de alguns de seus modelos de veículos[13].

### **2.4 Dificuldades da tecnologia RFID**

Apesar de bastante promissora, a tecnologia RFID ainda enfrenta empecilhos e problemas que a impedem de se destacar no mercado mundial. O fator que mais atrapalha nessa inserção, é a eficiência de protocolos anticolisão no processo de leitura, fazendo com que a tecnologia ainda seja um pouco lenta. Outro fator que atrapalha bastante a comercialização é a falta de um padrão global para regular as atividades da tecnologia RFID.

Outros problemas não menos importantes, mas que estão fora do escopo desse texto estão relacionados à segurança. Atualmente envolver criptografia nos dados das etiquetas diminui bastante a eficiência de leitura das mesmas, mas caso essa segurança não seja colocada, os dados podem ser facilmente capturados por qualquer usuário com um leitor de etiquetas.

Como dito anteriormente, a segurança não será foco desse texto, mas o problema de colisões será detalhado e algumas proposições serão feitas no intuito de otimizar esse processo.

## CAPÍTULO 3

# PROTOS COLS ANTICOLISÃO

Este capítulo descreve com maiores detalhes o evento de colisão entre etiquetas, como funcionam os protocolos anticolisão, sempre com ênfase no DFSA, protocolo utilizado durante toda a pesquisa para a escrita desse trabalho. Esses protocolos tem como principal função a diminuição da quantidade de colisões geradas na leitura de etiquetas.

### 3.1 Colisões

O processo de leitura de etiquetas, em geral, inicia com o leitor enviando sinal para as etiquetas requisitando que as mesmas enviem seus respectivos identificadores únicos. No entanto, por diversas vezes, mais de uma etiqueta recebe o mesmo sinal do leitor e transmite suas informações ao mesmo tempo, caracterizando o que chama-se na literatura de colisão.

As colisões são vistas como um dos principais problemas em aberto na utilização de sistemas RFID [2][3], pois acarretam em uma leitura lenta e ineficiente. As colisões acarretam uma má utilização do canal de comunicação, desperdício de energia, visto que é necessária uma retransmissão após serem detectadas as colisões.

O problema das colisões desencadeou um estudo sobre protocolos que pudessem evitar ou ao menos diminuir a ocorrência das mesmas. Esses protocolos que foram criados são os chamados protocolos anticolisão, e tem como função evitar as colisões de todas as maneiras.

### 3.2 Protocolos anticolisão

Protocolos anticolisão, como descrito sucintamente acima, são os algoritmos que buscam evitar ao máximo a ocorrência de colisões, otimizando assim a utilização de recursos e o tempo de leitura das etiquetas.

Vários protocolos desse tipo já existem na literatura das redes sem fio, porém, devido à natureza das etiquetas, isto é, o gasto de bateria deve ser evitado ao máximo, esses protocolos não se aplicam com naturalidade nos sistemas RFID. Para tal finalidade, foram propostos diversos protocolos, sua maioria com inspiração em protocolos já existem no mundo da internet sem fio.

Existem duas principais vertentes nos protocolos anticolisão para sistemas RFID. São elas:

- Baseados em ALOHA;
- Baseados em Árvore.

Os protocolos baseados em ALOHA são, em geral, probabilísticos, e funcionam tentando evitar ao máximo a probabilidade de ocorrer uma colisão. Os protocolos baseados em árvore trabalham construindo árvores onde não ocorrem colisões, sendo assim, algoritmos determinísticos.

Durante toda a pesquisa para elaboração desse trabalho, o foco definido sempre foi o protocolo DFSA, que pertence à classe dos protocolos ALOHA, portanto, os protocolos em árvore não serão descritos em detalhes nesse trabalho.

### **3.2.1 Protocolos baseados em ALOHA**

O protocolo *ALOHA* foi adaptado das redes sem fio para os sistemas RFID, e desde então diversas melhorias tem sido propostas, evoluindo o protocolo até o estado atual, o DFSA. O protocolo *ALOHA* inicial, ou *ALOHA* puro, é um protocolo baseado em divisão do tempo para os participantes do canal de comunicação, isto é, é um protocolo TDMA(*Time-Division Multiple Access*).

#### **3.2.1.1 ALOHA puro**

No ALOHA puro uma etiqueta transmite assim que ela possui dados para transmitir e está energizada. Uma das propriedades do *ALOHA* tradicional que poucas vezes é vista nos sistemas RFID é a característica onde o usuário do canal tem a capacidade de começar a se comunicar antes mesmo do leitor enviar qualquer tipo de comando ou sinal. Esse comportamento não é comumente visto em sistemas RFID pois para que ele ocorra, as etiquetas devem ter acesso ao meio para poder se comunicar. Nas etiquetas passivas e semiativas essa premissa não é verdadeira. Elas só transmitem após o leitor enviar algum tipo de sinal.

Se duas ou mais etiquetas transmitirem informações através do canal de comunicação no mesmo instante de tempo, ocorrerão colisões. As colisões no *ALOHA* puro podem ser do tipo parciais ou totais. Serão totais quando o espaço de tempo escolhido pelas etiquetas é exatamente igual, e parcial, caso haja a interseção no tempo, mas somente em partes desse tempo. No caso de colisão, as etiquetas são instruídas a parar a transmissão e só recomeçar após um tempo randômico.

A figura abaixo ilustra o comportamento do *ALOHA* puro com 3 etiquetas transmitindo. Pode-se notar uma colisão total entre a etiqueta 1 e a 2, e uma colisão parcial entre as etiquetas 2 e 3.



Figura 7. Funcionamento do ALOHA puro

O maior problema e necessidade de mudança do ALOHA puro é a grande probabilidade de ocorrência de colisões, visto que colisões parciais tem maior probabilidade de ocorrer. A primeira evolução dos protocolos ALOHA foi justamente para resolver essa dificuldade que envolvia as colisões parciais.

### 3.2.1.2 Slotted-ALOHA

Com a necessidade de resolver o problema das colisões parciais, os pesquisadores da época pensaram na solução proposta como *Slotted-ALOHA*. O protocolo consiste em utilizar um protocolo TDMA similar ao ALOHA puro, porém com a alteração de que as etiquetas só poderão transmitir dentro de slots de tempo pré-determinados. Com essa modificação, todas as colisões que ainda existirem serão totais, diminuindo o desgaste do uso do canal com as colisões parciais.

O funcionamento básico do *Slotted-ALOHA* acontece da seguinte forma: As etiquetas só poderão transmitir no início de cada slot, eliminando as colisões parciais. Caso ainda ocorra colisão, as etiquetas esperam um tempo randômico e transmitem novamente, até que não restem mais etiquetas não-identificadas. Para que isso ocorra corretamente, é necessário que todas as etiquetas saibam quando começa e quando termina cada slot, sendo essa atividade responsabilidade do leitor.

A figura abaixo ilustra o funcionamento do *slotted-ALOHA*.

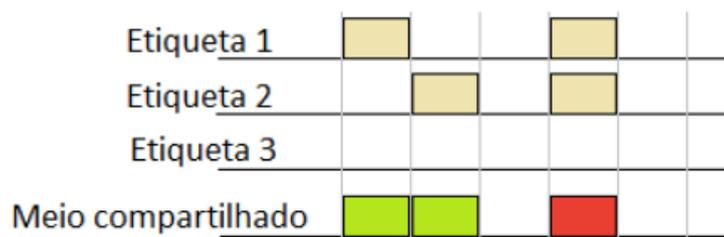


Figura 8. Funcionamento do Slotted-ALOHA

### 3.2.1.3 Framed Slotted-ALOHA

O *Framed Slotted-ALOHA*, ou FSA, é uma evolução em cima do *Slotted-ALOHA*. A principal diferença é que os slots, presentes no *Slotted-ALOHA*, agora são divididos em frames, isto é, “pacotes” de slots.

O principal problema que motivou essa proposta foi a observação de que etiquetas com uma maior frequência, tendiam a transmitir mais vezes que as etiquetas com menores frequências. Esse comportamento ocasionava mais colisões, visto que uma etiqueta poderia tentar transmitir a cada 3 slots, por exemplo.

Com a inserção do conceito de Frame, cada etiqueta só pode transmitir em um slot por frame. Caso haja colisão, terá que esperar o próximo Frame ser iniciado, selecionar um slot aleatoriamente e tentar a transmissão novamente. Esse processo é repetido até que todas as etiquetas sejam identificadas com sucesso.

Embora contribua para a solução do problema de etiquetas com diferentes frequências, o *Framed Slotted-ALOHA* inseriu um problema complexo com a ideia de Frame: O tamanho do Frame. Esse é um dos maiores problemas dos sistemas RFID atualmente. A problemática gira em torno da quantidade de etiquetas e tamanho do frame. Caso existam muitas etiquetas e o tamanho do frame seja pequeno, existirão muitas colisões. Caso o frame seja grande para a quantidade de etiquetas, o canal de comunicação estará subutilizado constantemente, atrasando o processo de leitura.

A figura abaixo ilustra o funcionamento do *Framed Slotted-ALOHA*.

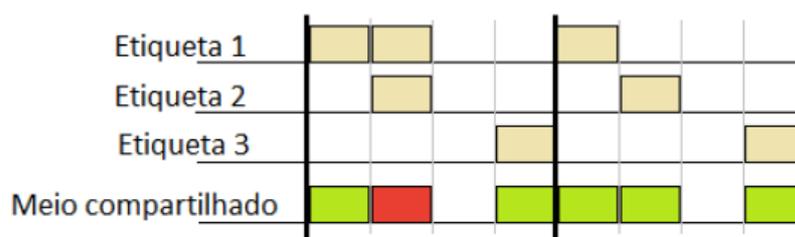


Figura 9. Funcionamento do *Framed-Slotted-ALOHA*

Para tentar resolver esse problema do tamanho ideal do frame, foi proposto o DFSA, *Dynamic Framed Slotted-ALOHA*.

### 3.2.1.4 Dynamic Framed Slotted-ALOHA

O *Dynamic Framed Slotted-ALOHA*, ou DFSA, é o protocolo anticolisão mais avançado no momento[2]. Ele consiste em variar o tamanho do frame, seguindo a lógica do FSA. Para tal, a cada frame executado, o leitor capta as informações de quantidade de slots em colisão,

quantidade de slots vazios e quantidade de slots com sucesso, para, com esses dados, conseguir estimar o tamanho de frame ideal para o próximo frame a ser lançado.

Em geral, a estimativa do tamanho ideal para o próximo frame se dá em função da quantidade de etiquetas que ainda não foram reconhecidas pelo leitor. Essa estimativa se baseia na quantidade de etiquetas restantes para conseguir obter a máxima utilização do canal de comunicação. Esta estimativa de quantidade de etiquetas é feita por algoritmos específicos, chamados de estimadores, que serão expostos em detalhes na próxima seção.

Em suma, a cada frame executado, o leitor captura a quantidade de slots em colisão, sucesso e vazios que ocorreram no frame, e com esses valores consegue estimar a quantidade de etiquetas restantes que ainda não foram identificadas. Com o valor da estimativa de etiquetas restantes, ele consegue estimar o tamanho ideal para o próximo frame, de modo que minimize a quantidade de slots vazios e a quantidade de slots em colisão.

A figura abaixo ilustra esse comportamento descrito acima. No início o tamanho do frame é pequeno para a quantidade de etiquetas, mas ao final do frame ele é ajustado para obter uma melhor performance.

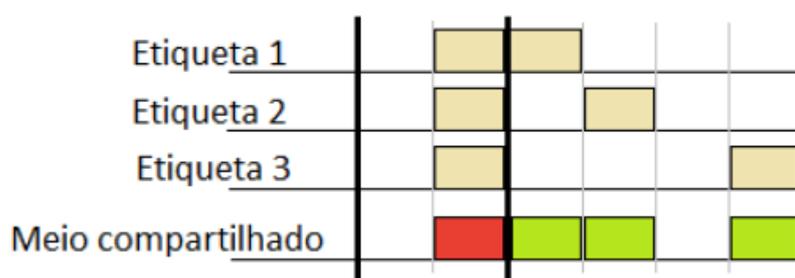


Figura 10. Funcionamento do *Dynamic-Framed-Slotted-ALOHA*

### 3.3 Estimadores para o DFSA

Os algoritmos estimadores, como citado anteriormente, são parte integrante do DFSA, sendo descritos com mais detalhes nessa seção. Ao executar o ciclo de leitura de etiquetas, o DFSA utiliza algum algoritmo estimador para determinar o tamanho que o próximo frame terá.

Existem vários algoritmos que foram pensados exclusivamente para resolver esse problema de estimativa da quantidade de etiquetas restantes e de tamanho ótimo para o próximo frame do DFSA. Em geral, calcula-se que em cada slot em colisão existem pelo menos 2 etiquetas que não transmitiram seu conteúdo, mas cada estimador tem sua forma própria de estimar essa quantidade.

Como dito anteriormente, calcula-se primeiro a quantidade de etiquetas restantes, para com esse dado, calcular o tamanho ideal para o próximo frame.

Existem vários estimadores na literatura, mas abordaremos somente o Lower Bound[3], como forma de entender o papel de um estimador, o Bayesian[1], que será o

algoritmo utilizado para fazer o estudo aprofundado do uso da decomposição em fatores primos, que é o objetivo desse trabalho. Mostraremos também o Chen[2], que é um algoritmo que também utiliza probabilidades condicionais em sua estrutura e pode ser um trabalho futuro analisar seu desempenho com a técnica de decomposição, e o Vogt[3], que é um dos estimadores mais utilizados na literatura como comparador, visto que a maioria dos pesquisadores da área já conhecem o seu comportamento.

### 3.3.1 Lower Bound

O Lower Bound é, com certeza, o estimador mais básico para o protocolo DFSA. A ideia central do Lower Bound é que para ocorrer uma colisão em um slot, pelo menos duas etiquetas enviaram seus dados para esse mesmo slot. Considere  $\langle c_0, c_1, ck \rangle$  o resultado de um ciclo no DFSA, onde  $c_0$  é a quantidade de slots vazios,  $c_1$  é a quantidade de slots com sucesso e  $ck$  a quantidade de slots em colisão, o número mínimo de etiquetas que concorreram por slots nesse frame,  $N$ , é:

$$N = (2 * ck) + c_1 \quad (1)$$

Com o número de etiquetas que disputaram por slots em mãos, o Lower Bound faz a estimativa do próximo frame da seguinte forma:

$$N_{(F+1)} = N_F - c_1 \quad (2)$$

Na equação acima,  $F$  representa o frame que foi concluído e  $F + 1$ , o próximo frame a ser executado. Segundo (2), vemos que o tamanho do próximo frame no Lower Bound é determinado como o tamanho do frame anterior menos a quantidade de etiquetas que transmitiram com sucesso.

O Lower Bound foi proposto por Vogt em [3] e é utilizado literalmente como fronteira para novas propostas. Se o algoritmo tiver um desempenho pior que o Lower Bound, em geral, é repensada sua concepção.

### 3.3.2 Vogt

Este estimador foi proposto por Vogt em [3]. O algoritmo basicamente busca encontrar o ponto onde a diferença entre os valores obtidos ao fim do frame atual e os valores esperados é a menor possível.

$$f(L, E, S, C) = \min (n) \left| \begin{pmatrix} a_0^{L,n} \\ a_1^{L,n} \\ a_{\geq 2}^{L,n} \end{pmatrix} - \begin{pmatrix} E \\ S \\ C \end{pmatrix} \right| \quad (3)$$

Segundo (3), vemos que o algoritmo encontra o valor que minimiza a diferença entre o vetor de valores esperados, onde  $L$  é o tamanho do frame atual,  $E$  é a quantidade de slots vazios,  $S$  a quantidade de slots com sucesso na transmissão e  $C$  a quantidade de slots em colisão. Os valores esperados para cada um desses citados anteriormente podem ser vistos no outro vetor, e são calculados da seguinte forma:

$$a_0^{L,n} = L(1 + \frac{1}{L})^n \quad (4)$$

$$a_1^{L,n} = L * n(\frac{1}{L})(1 + \frac{1}{L})^{n-1} \quad (5)$$

$$a_{\geq 2}^{L,n} = L - a_0^{L,n} - a_1^{L,n} \quad (6)$$

Na época em que foi desenvolvido, o algoritmo do Vogt tinha uma limitação na determinação do tamanho do frame. Embora essa estimativa seja calculada em função do número de etiquetas, somente potencias de 2(dois) poderiam ser utilizadas como tamanho de um frame, por isso foi desenvolvida a tabela abaixo após experimentações.

**Tabela 1. Tamanho do frame em função do  $n$  estimado**

<b>Valor estimado</b>	<b>Tamanho do <i>frame</i></b>
<b>[1,9]</b>	16
<b>[10,27]</b>	32
<b>[17,56]</b>	64
<b>[51,129]</b>	128
<b>[112,∞]</b>	256

A tabela 1 mostra como Vogt descreveu a estimativa de tamanho de frame em função do número de etiquetas estimadas. Nota-se que existem interseções entre os valores de etiquetas. Esse comportamento é explicado pelo autor como uma variação possível [3], visto que esses valores foram determinados experimentalmente, e não de forma rígida através de equações matemáticas.

### 3.3.3 Chen

Este estimador foi proposto por Chen em [2] e se baseia no uso de probabilidade máxima *a posteriori* para a determinação do número de etiquetas que disputaram slots no último frame executado.

Considerando  $n$  etiquetas e um tamanho de frame  $L$ , em um slot qualquer do frame, o número de etiquetas que estarão alocadas nesse slot segue uma distribuição binomial com uma probabilidade de ocupação de  $1/L$ . A probabilidade de encontrar  $k$  etiquetas em um slot específico do frame pode ser calculada por:

$$B(r) = \binom{n}{r} \left(\frac{1}{L}\right)^r \left(1 - \frac{1}{L}\right)^{n-r} \quad (7)$$

Segundo (7), podemos calcular de forma análoga a probabilidade de vazio, sucesso ou colisão para um slot específico por:

$$P_e = B(0) = \left(1 - \frac{1}{L}\right)^n \quad (8)$$

$$P_s = B(1) = \frac{n}{L} \left(1 - \frac{1}{L}\right)^{n-1} \quad (9)$$

$$P_c = 1 - P_e - P_s \quad (10)$$

O ponto chave desse algoritmo é o cálculo da probabilidade de obter exatamente  $E$  slots vazios,  $S$  slots com sucesso,  $C$  slots em colisão em um frame de tamanho  $L$ . Esse problema se comporta como uma distribuição multinomial, que pode ser descrita com a seguinte equação:

$$P(E, S, C) = \frac{L!}{E! S! C!} P_e^E P_s^S P_c^C \quad (11)$$

Já temos os valores das probabilidades individuais vistas na equação acima em (8), (9) e (10). Substituindo os valores teremos a equação completa que ilustra a estimativa do número de etiquetas no algoritmo do Chen. Vale ressaltar que buscamos o valor de  $n$  onde essa probabilidade é máxima:

$$N_{est} = n, \text{ tal que } P(n|E, S, C) \text{ é máxima} \quad (12)$$

### 3.3.4 Bayesian

Este estimador foi proposto por Wu and Zeng em [1], e tem como principal diferença frente aos outros estimadores discutidos nesse trabalho, o uso exclusivo de um processo Bayesiano para a determinação do número de etiquetas e tamanho ótimo do próximo frame.

Seja  $c$  uma observação dos resultados obtidos pelo leitor a cada frame da quantidade de slots vazios, slots em colisão e slots com sucesso, isto é,  $c = \langle c_0, c_1, c_k \rangle$ . Seja também  $n$  o número de etiquetas e  $L$  o tamanho do frame. Muitas teorias estatísticas e métodos estatísticos consideram  $c$  como uma variável aleatória de valor randômico e  $n$  uma variável aleatória determinística de valor desconhecido. Dado que  $c \sim P(c|n)$ , o algoritmo [1] se utiliza dessa propriedade para propor uma diferença significativa frente aos outros estimadores bayesianos: Os autores consideram  $n$  não como uma variável determinística de valor desconhecido, mas sim como uma variável aleatória de valor randômico.

Ao se utilizar dessa proposição, o método Bayesiano, como é chamado pelos autores, atualiza a probabilidade *a priori*, isto é,  $P(c|n)$ , com a probabilidade *a posteriori*, isto é,  $P(n|c)$ , e assim consegue determinar um valor de  $n$  mais próximo do real.

Dado que as probabilidades de um slot estar vazio, com sucesso ou em colisão, são, respectivamente:

$$P_e = B(0) = \left(1 - \frac{1}{L}\right)^n \quad (13)$$

$$P_s = B(1) = \frac{n}{L} \left(1 - \frac{1}{L}\right)^{n-1} \quad (14)$$

$$P_c = 1 - P_e - P_s \quad (15)$$

A probabilidade de  $n$  dada uma observação  $c$ , onde  $c = \langle E, S, C \rangle$  é:

$$P(E, S, C) = \frac{L!}{E! S! C!} P_e^E P_s^S P_c^C \quad (16)$$

A fórmula descrita em (16) mostra uma probabilidade condicional de  $n$  dado  $c$ , isto é, mede a probabilidade do número de etiquetas ter um valor  $n$ , dado que o a tripla de valores de slots vazios, em colisão e com sucesso no último frame é  $c$ . A equação (16) segue uma distribuição multinomial onde  $L$  é o tamanho do frame,  $c$  é a tripla com os resultados do último frame e as probabilidades individuais estão descritas em (13), (14) e (15). Portanto, a probabilidade condicional de  $n$  dado  $c$ , é uma equação cujo resultado é facilmente encontrado através da resolução dessas equações mais simples.

Essa equação é descrita no trabalho original para ser usada exatamente como está descrita em (16), e assim como no algoritmo Chen, citado anteriormente nesse capítulo, a manipulação desses fatoriais é um grande problema dos métodos bayesianos, e será explorada em detalhes na próxima sessão desse trabalho.

A estimativa de etiquetas restantes no processo de leitura das etiquetas descrita no algoritmo segue a seguinte equação:

$$\hat{n} = \arg \min_{\tilde{n} \in \omega} \sum_{n=1}^{+\infty} J(\tilde{n}, n) p(n|c) \quad (17)$$

Na equação (17), o termo  $p(n|c)$  é a probabilidade de  $n$  dado  $c$ , cujo valor pode ser obtido com o uso da equação (16). O somatório de 1 a infinito tem o propósito de percorrer toda a curva gerada pelos outros termos multiplicados, e assim, conseguir determinar o valor que minimiza essa curva, isto é, o valor que buscamos para a estimativa do número de etiquetas e o termo  $J(\tilde{n}, n)$  é uma função de risco.

No artigo original, os autores delimitam 3 tipos de função de risco que foram usadas para medir a eficiência do algoritmo, são elas: *mean-square*, *absolute-error probability* e *posterior-error probability*. Ao longo dos experimentos desse trabalho, utilizamos a *mean-*

*square* e a *absolute-error*, com um foco acentuado para o uso da *absolute-error*. Ainda sobre a equação (17), o intervalo de valores em  $\omega$  é descrito por:

$$\omega = \{\tilde{n} | E + 2 * C \leq \tilde{n} \leq N\} \quad (18)$$

E as funções de risco são descritas como:

1) Mean-Square:

$$J(\tilde{n}, n) = (\tilde{n} - n)^2 \quad (19)$$

2) Absolute-Error:

$$J(\tilde{n}, n) = |\tilde{n} - n| \quad (20)$$

O tamanho ideal do próximo frame é definido através de experimentos, onde o autor mostra que o valor ideal para o próximo frame é descrito como:

$$L = \lfloor k * n \rfloor \quad (21)$$

Ainda segundo experimentos, notou-se que a utilização máxima do canal é obtida quando  $k \approx 1.7$ .

### 3.4 Melhorias Propostas

O Teorema de Bayes, famoso corolário utilizado em diversas áreas do conhecimento com ênfase na estatística, enuncia, de forma bastante simplificada, que a probabilidade de um evento *A* dado um evento *B*, depende não apenas do relacionamento entre *A* e *B*, mas também da probabilidade individual de cada evento ocorrer. Esse teorema é a base para os algoritmos estimadores que utilizam probabilidade *a posteriori* para a obtenção da estimativa de etiquetas [1][2][12].

Como dito, a utilização do Teorema de Bayes é uma prática bastante comum e cotidiana no meio acadêmico, porém algumas desvantagens são trazidas de seu uso. Ao implementar e analisar mais profundamente o algoritmo Bayesiano[1], notou-se que um problema bem

comum atinge toda essa cadeia de algoritmos que utilizam abordagens bayesianas na estimativa de etiquetas.

Ao trabalhar com esse teorema, comumente nos deparamos com a equação (16) do método bayesiano, que é idêntica à equação (11), proposta por Chen em [2]. Essa equação é a resolução da probabilidade condicional de dois eventos distintos e estará presente em todos os algoritmos estimadores que utilizem a abordagem bayesiana, como é o caso de [1], [2] e [12], por exemplo.

Na implementação do algoritmo de [1] observou-se que ao atingir um número de slots superior a 140 (cento e quarenta), o valor de  $L!$  é imediatamente caracterizado como um fator infinito. Isso acontece pois em todas as linguagens de programação a representação de números tem um limite. Embora esse limite seja, em geral, mais que suficiente para aplicações cotidianas, em aplicações científicas, esse valor está bem aquém do necessário para representação de certos valores no cálculo das soluções. A operação de fatorial de números, com frequência ultrapassa esse limite.

Foi com o objetivo de sanar esse problema em toda a família de algoritmos que se baseiam em processo bayesiano para estimativa de quantidade de etiquetas em um frame, que foram desenvolvidas duas abordagens que eliminam o problema de fatores infinitos, independentemente da linguagem de programação escolhida, pois as soluções apresentadas foram atuam no campo da matemática.

As duas soluções idealizadas nesse trabalho estão descritas em mais detalhes nas sessões seguintes.

### **3.4.1 Método da Decomposição**

A primeira das soluções propostas para o problema é a ideia mais simples a ser tomada: Decompor os números grandes em fatores primos, de forma a trabalhar com um conjunto de números pequenos, em vez de trabalhar diretamente com os números grandes, que podem facilmente “estourar” a representação dos mesmos via linguagens de programação.

Para fazer isso, utilizou-se a propriedade matemática que enuncia que todo número inteiro positivo e maior que um, pode ser escrito de forma única como o produto de vários números primos. Essa técnica é conhecida como decomposição em fatores primos.

A decomposição foi desenvolvida de forma simples, utilizando-se de um array, isto é, uma lista com tamanho definido, onde cada posição representa um número primo e à medida que o número vai sendo fatorado, o valor de cada posição do array vai sendo incrementada ou decrementada para controlar o expoente dos fatores primos.

Como exemplos de decomposição em fatores primos, temos:

- 1) O número 2205 é decomposto em  $3^2 * 5 * 7^2$ ;

- 2) O número 175 é decomposto em  $5^2 * 7$ ;
- 3) O número 149 é decomposto como 149, visto que é um número primo.

Na implementação da solução, inicialmente é feita a decomposição do numerador, que no caso da equação (16) é o fator  $L!$ . Com isso, teremos uma lista com os fatores primos que representam  $L!$  e repetimos o mesmo processo para o denominador, isto é,  $E! S! C!$ . Em seguida, é feita a simplificação de fatores efetuando a subtração dos expoentes dos fatores no numerador, pelos expoentes dos fatores dos denominadores. O exemplo abaixo ilustra o método:

$$\frac{12}{4} = \frac{2^2 * 3^1}{2^2} = \frac{2^2 * 2^{-2} * 3^1}{1} = \frac{2^0 * 3^1}{1} = 3$$

No exemplo acima, o fator 2 do numerador é simplificado com o fator 2 do denominador, gerando como resultado da equação o valor 3. Esse processo ocorre na implementação utilizando-se de um *array*, onde a posição guarda o expoente dos fatores, seguindo a mesma lógica do exemplo acima.

### 3.4.2 Método da Combinação

Nessa segunda abordagem proposta, utiliza-se manipulação matemática da própria equação (16) para eliminar os fatores infinitos do cálculo de fatoriais. A utilização desse método é bem mais eficiente, computacionalmente falando, que a abordagem de decomposição.

Para explicar esse método, utilizaremos somente a fração da equação (16), que é onde o “estouro” realmente acontece nos métodos bayesianos.

Sabendo que:

$$L = C_0 + C_1 + C_k \quad (22)$$

Então:

$$C_k = L - C_0 - C_1 \quad (23)$$

Substituindo na fração original, temos:

$$\frac{L!}{C_0! C_1! C_k!} = \frac{L!}{C_0! C_1! (L - C_0 - C_1)!} \quad (24)$$

Para chegar ao resultado que buscamos, temos que manipular o lado direito da equação (24), para que ao final, tenhamos diminuído a complexidade de cada cálculo, fazendo com que os fatores infinitos não mais aconteçam. Chegamos nesse resultado dividindo a expressão inicial em duas combinações onde o  $L!$  é fracionado, e assim, não gera mais fatores infinitos.

A manipulação matemática necessária segue abaixo:

$$\frac{L!}{c_0! c_1! c_k!} = \frac{L!}{c_0! c_1!(L-c_0-c_1)!} = \frac{L!}{c_0!} * \frac{1}{c_1!(L-c_0-c_1)} * \frac{(L-c_0)!}{(L-c_0)!} \quad (25)$$

$$\frac{L!}{c_0! c_1! c_k!} = \frac{L!}{c_0!(L-c_0)!} * \frac{(L-c_0)!}{c_1!((L-c_0)-c_1)} \quad (26)$$

Note que o último fator de (25) é criado por nós, entra simplesmente para que a equação chegue ao resultado que queremos. Sabendo que  $\binom{n}{r} = \frac{n!}{r!(n-r)!}$ , chegamos ao seguinte resultado:

$$\frac{L!}{c_0! c_1! c_k!} = \binom{L}{c_0} * \binom{(L-c_0)}{c_1} \quad (27)$$

Essa modificação garante que não mais existirão fatores infinitos no cálculo da probabilidade da equação (18), exceto no cenário onde o tamanho do frame seja mais que 1000 slots e a quantidade de slots vazios seja exatamente metade do tamanho do frame, ou seja, 500. Em nossas simulações não houve nenhum “estouro”.

### 3.5 Resumo

Os algoritmos estimadores, como citado anteriormente, são parte integrante do DFSA, e foram descritos com mais detalhes nessa seção. Ao executar o ciclo de leitura de etiquetas, o DFSA utiliza um algoritmo estimador, que, ao estimar o número de etiquetas que estão ativas no sistema RFID, determina o tamanho ideal para o próximo frame.

Ao longo do capítulo foram mostrados vários estimadores, dentre os tipos de estratégias citados temos os estimadores baseados em árvore, aos quais o escopo desse trabalho impossibilitou uma análise mais profunda, e os estimadores baseados em *ALOHA*, cuja evolução foi descrita em detalhes.

Destacamos novamente a importância de seleção de um bom estimador para um sistema RFID funcionar de forma correta e eficiente, pois sem um estimador adequado, o tempo de espera pode invalidar o uso da tecnologia, como discutido no início do capítulo.

Sendo assim, é de vital importância que os estimadores sejam estimulados a lidar com todos os cenários mais comuns onde, em geral, sistemas RFID serão utilizados.

Ainda nesse capítulo foram apresentadas duas abordagens de solução para o problema de fatores infinitos: A primeira delas, faz a decomposição de números grandes em um produto de números primos, eliminando a chance de ocorrência de fatores que “estourem” a representação de números em qualquer linguagem de programação. A segunda, elimina esses fatores infinitos manipulando a própria equação de probabilidade, de forma a simplificar o cálculo, fazendo com que os fatores infinitos sejam erradicados.

Nota-se por fim, que os estimadores de etiquetas são o trunfo do DFSA, sendo de fundamental importância que sejam evoluídos continuamente pela academia.

## CAPÍTULO 4

### ANÁLISES E RESULTADOS

Com o intuito de avaliar, analisar o desempenho e provar o conceito das abordagens propostas, foi desenvolvido um simulador que implementa o algoritmo original dos autores, assim como descrito no artigo [1], o algoritmo com o método da decomposição e o algoritmo com o método da combinação. Assim, ficou possível fazer análises mais profundas sobre qual algoritmo tem melhor desempenho nas métricas mais comuns em sistemas RFID.

#### 4.1 Simulador

O simulador desse trabalho foi escrito na linguagem Java e foi desenvolvido de forma a ser extremamente simples de mudar qualquer parâmetro da simulação. Embora tenha essa facilidade porque o código está extremamente parametrizado, não foi desenvolvida uma interface gráfica, visto que todos os usuários que utilizaram o simulador foram pessoas técnicas da área de TI, e não se fez necessário o desenvolvimento de uma interface gráfica para esse público-alvo.

Como saída do simulador temos 5 tipos de gráficos, cada um com um foco específico de análise. São eles:

- Número total de slots: Esse gráfico faz uma relação entre a quantidade de etiquetas e o número total de slots utilizados para ciclos de leitura completos em cada algoritmo.
- Número de slots em colisão: Esse gráfico analisa a quantidade de slots em colisão de acordo com o número de etiquetas em ciclos completos de leitura.
- Número de slots vazios: Esse gráfico analisa a quantidade de slots vazios de acordo com o número de etiquetas em ciclos completos de leitura.
- Número total de ciclos: Esse gráfico faz a relação entre a quantidade de etiquetas e o número de ciclos gastos para a leitura completa das etiquetas simuladas.
- Erro absoluto médio: Esse gráfico mede o erro durante o processo de leitura completo das etiquetas.

## 4.2 Validação do simulador

Como forma de validar e garantir a corretude do simulador e da implementação do algoritmo estudado, foram gerados gráficos reproduzindo os cenários descritos pelos autores no artigo original. É necessário que os resultados sejam muito semelhantes, para termos a certeza que o algoritmo foi implementado de forma correta.

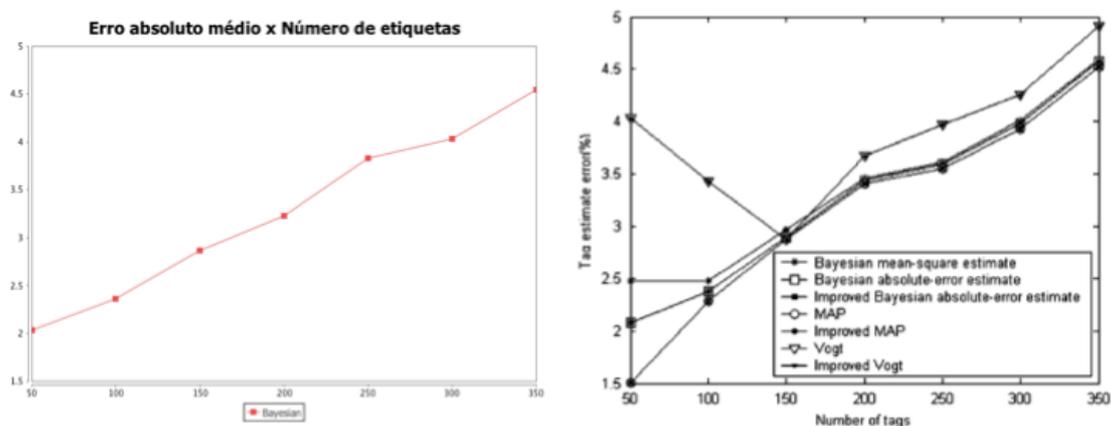


Figura 11. Validação do simulador - Gráfico de erro médio com função de risco *absolute-error*

A figura acima(11) mostra a comparação entre um gráfico gerado pelo simulador à esquerda e um gráfico retirado de [1] à direita. Ambos os gráficos fazem uma análise do erro na estimativa de etiquetas(em porcentagem) pelo aumento do número de etiquetas. O cenário de teste foi retirado do artigo original, isto é, frame inicial de 128 slots e o número de etiquetas varia de 50 a 350 em saltos de 50 etiquetas. A única diferença na simulação gerada e o gráfico do artigo original é que utilizamos 1000 experimentos em cada valor de etiquetas, no artigo original foram feitos 500 experimentos para cada salto na quantidade de etiquetas.

Note que o gráfico gerado pelo simulador só contém uma das séries trazidas no gráfico original. Isso ocorre pois nesse gráfico, para facilitar a comparação visual, só foram gerados os resultados do algoritmo *absolute-error*, que nas séries do artigo original, é representado pelo símbolo de um quadrado.

Note também que por se tratar de um processo não-determinístico, alguns valores podem diferir levemente do original, não descaracterizando a corretude do simulador.

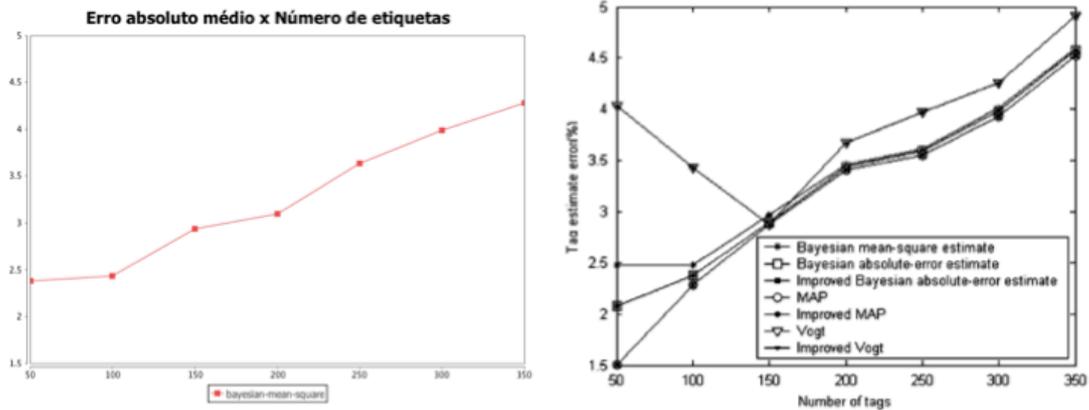


Figura 12. Validação do simulador - Gráfico de erro médio com função de risco *mean-square*

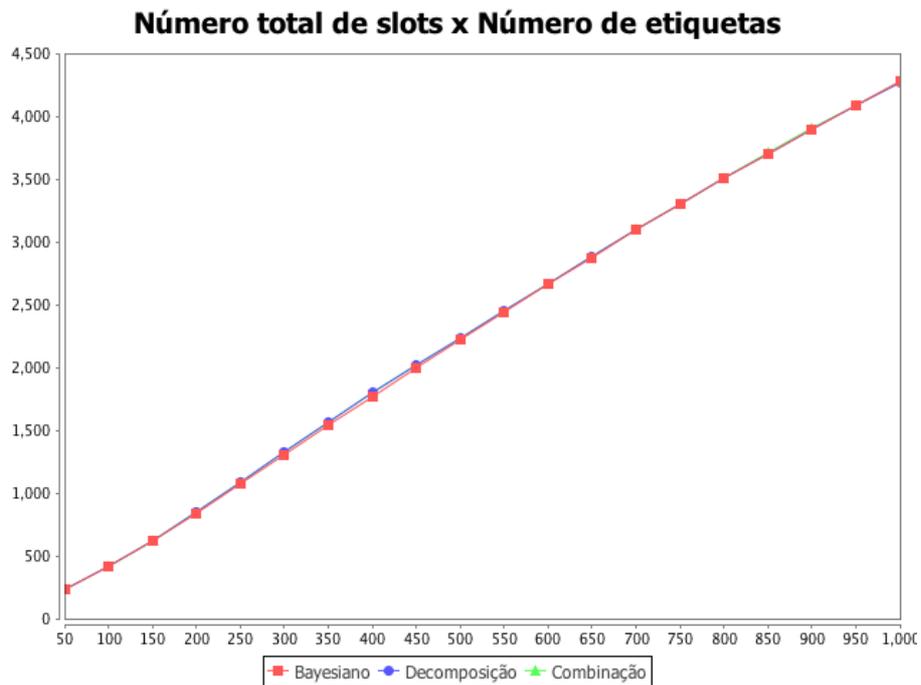
A figura (12) mostra novamente a comparação entre um gráfico gerado pelo simulador, e um gráfico retirado do artigo original. Seguindo o mesmo cenário descrito para a figura (11), isto é, avaliar o erro absoluto médio em porcentagem com frame inicial de 128 slots e o número de etiquetas variando de 50 a 350 em saltos de 50 etiquetas, conseguimos provar a partir da análise desse gráfico que para a função de risco *mean-square*, o simulador também está correto.

Novamente vale ressaltar que por se tratar de um procedimento não-determinístico, alguns valores podem oscilar levemente, mas no geral, o gráfico representa de forma bastante confiável o resultado do artigo original.

### 4.3 Resultados e análises das abordagens propostas

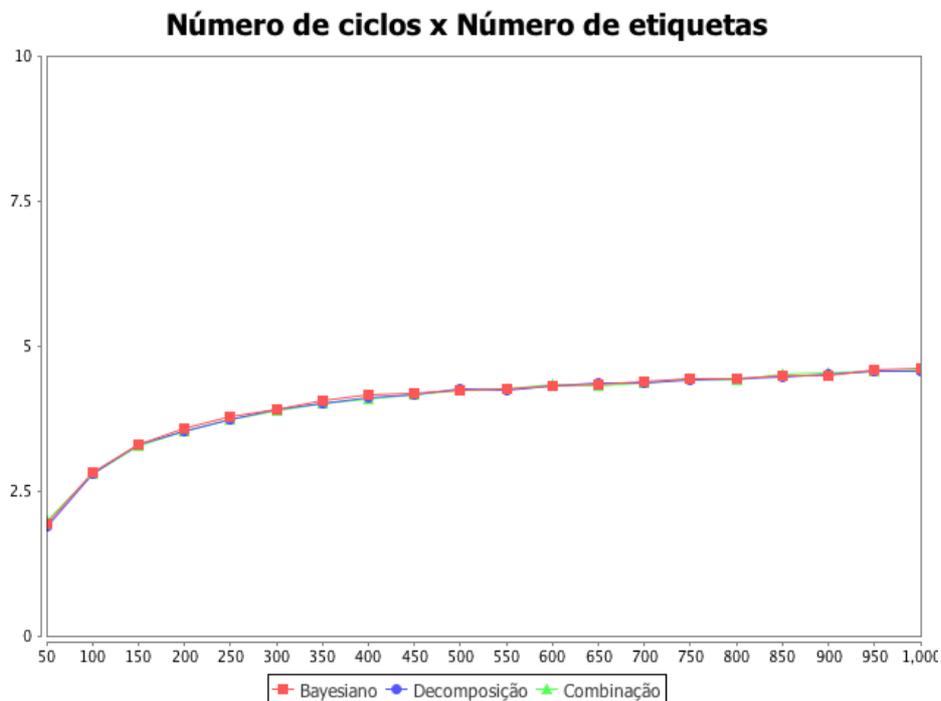
Para avaliar o desempenho das duas abordagens propostas, para a solução do problema de fatores infinitos no cálculo de fatoriais, foram gerados vários gráficos que ilustram esse desempenho.

Os valores utilizados em todos esses experimentos foram: Tamanho inicial do frame de 128 slots, variação da quantidade de etiquetas de 50 a 1000, com o passo de 50, isto é, saltos de 50 no número de etiquetas. Para cada passo, foram realizadas 1000 simulações, com o intuito de deixar a curva o mais suave possível. Para critérios de comparação, foi escolhida a função de risco *absolute-error*, exceto nos gráficos de erro percentual, onde a função de risco *mean-square* também é exibida, como forma de ilustrar as abordagens utilizando as duas funções. A seguir, são exibidos e comentados cada gráfico gerado.



**Figura 13. Comparativo de número total de slots x Número de etiquetas**

A figura (13) mostra o primeiro dos gráficos gerados nas simulações. O gráfico mostra a relação do uso de slots no processo de leitura completo, e varia de acordo com a quantidade de etiquetas. Nota-se analisando a imagem, que as curvas praticamente se sobrepõem, mostrando que nenhuma das abordagens se sobressai em relação às outras no critério de uso de slots totais durante a leitura.



**Figura 14. Comparativo da quantidade de ciclos x Número de etiquetas**

A figura (14) mostra a relação entre a quantidade de ciclos utilizados no processo de leitura completo das etiquetas em relação à quantidade de etiquetas inicial. Novamente, vemos através do gráfico que as abordagens se sobrepõem, não gerando qualquer vantagem real para nenhuma das abordagens analisadas.

Esse resultado mostra que mesmo a abordagem original do autor[1] levando a fatores infinitos, ele consegue, utilizando o *lower bound*[3] nesses casos, ter um desempenho idêntico ao desempenho observado com a eliminação desses fatores infinitos.

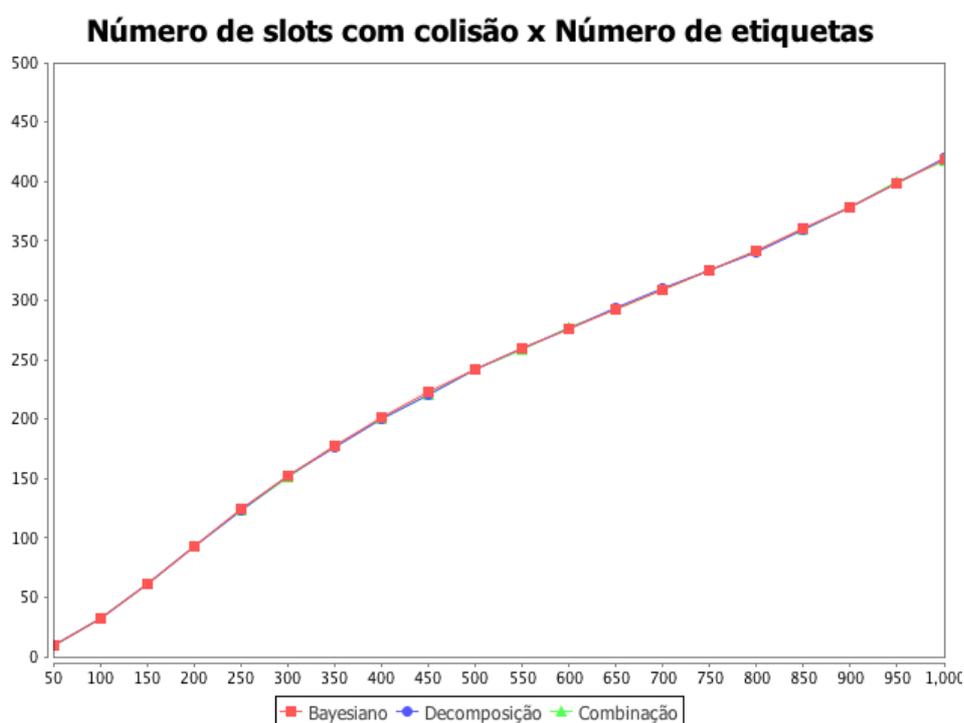


Figura 15. Comparativo da quantidade de slots com colisão x Número de etiquetas

A figura (15) mostra a relação entre a quantidade de slots em colisão gerados durante os ciclos de leitura completa em função do número de etiquetas inicial. Novamente pode-se observar que as curvas são praticamente idênticas, não se sobressaindo nenhuma abordagem analisada. Novamente vemos que o fato de gerar fatores infinitos na leitura das etiquetas, não acarreta em um desempenho ruim do algoritmo descrito no artigo original[1].

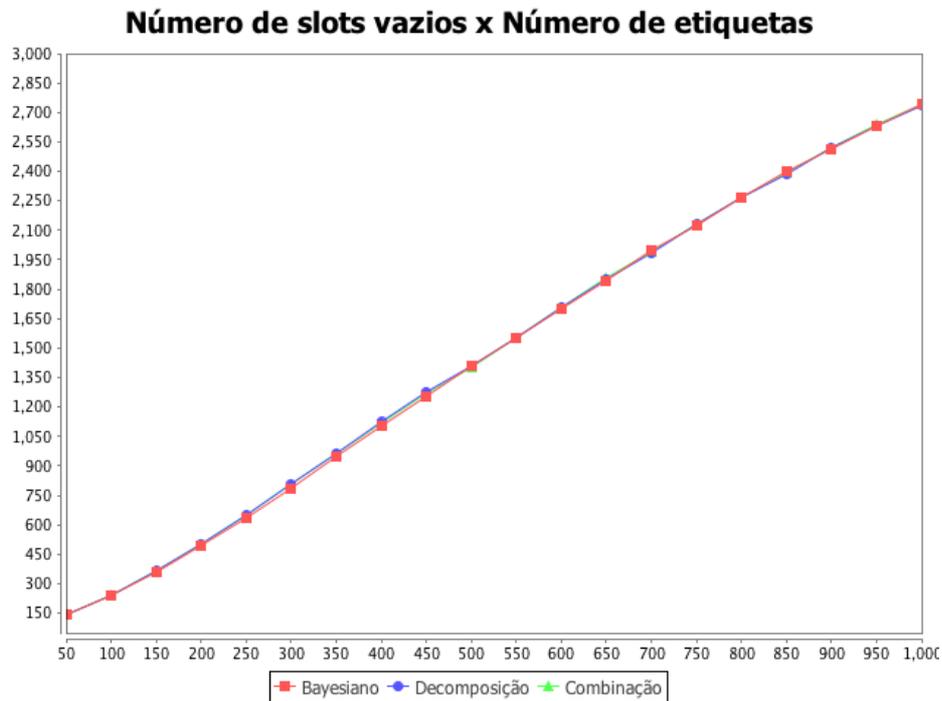


Figura 16. Comparativo do número de slots x Número de etiquetas

A figura (16) ilustra a comparação entre o número de slots vazios gerados no processo de leitura completo em relação ao número de etiquetas inicial. Novamente os resultados não mostram qualquer vantagem para nenhuma abordagem analisada. Vê-se que o algoritmo original gera menos slots vazios, de forma bem sutil, na faixa de 250 a 500 etiquetas, mas no geral, os algoritmos geram resultados idênticos.

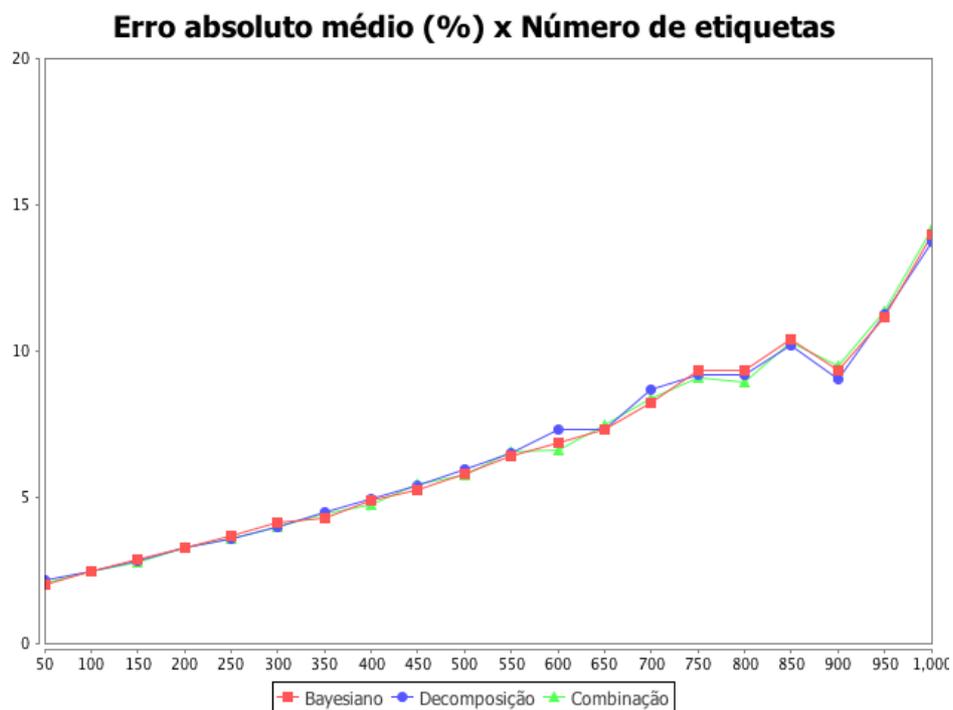


Figura 17. Erro absoluto médio x Número de etiquetas - Função de risco *absolute-error*

A figura (17) mostra a comparação entre o erro médio percentual em relação à quantidade de etiquetas com a função de risco *absolute-error*. Novamente vemos curvas bem semelhantes, porém nesse caso pode-se dizer que o algoritmo da combinação obteve resultados melhores que os demais na faixa de valores de etiquetas entre 550 e 850, e um resultado idêntico aos outros nas outras faixas estudadas. O algoritmo da decomposição oscila com o algoritmo original[1] e tem resultados piores entre 550 e 750, mas exibe resultados melhores entre 750 e 900. Com a análise desse gráfico, pode-se afirmar que com a abordagem da combinação, conseguimos obter um melhor resultado que o resultado original.

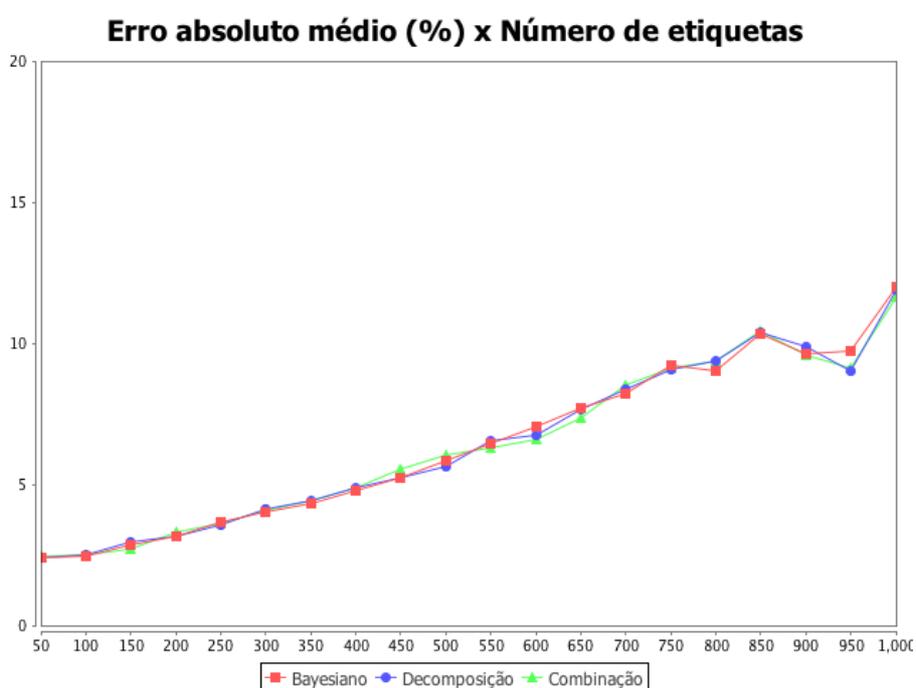


Figura 18. Erro absoluto médio x Número de etiquetas - Função de risco *mean-square*

A imagem (18) mostra a comparação entre o erro médio percentual e a quantidade de etiquetas inicial com o uso da função de risco *mean-square*. Mesmo utilizando uma função de risco diferente, o resultado é bem semelhante aos demais. Observa-se uma leve superioridade do algoritmo da decomposição frente aos outros. O ponto máximo de diferença acontece com a quantidade de etiquetas inicial igual a 950, onde o algoritmo da decomposição tem pouco menos que 1% a menos de erro. Avaliando que esse erro médio foi executado 1000 vezes, é um bom resultado, mesmo que numericamente pequeno. O algoritmo da combinação também se mostra ligeiramente mais eficaz que o algoritmo original durante quase toda a curva apresentada.

## 4.4 Dificuldades

Ao longo desse trabalho, várias dificuldades surgiram, mas conseguimos resolvê-las de forma que não impactassem no projeto final. Como primeiro ponto, podemos citar a complexidade do próprio algoritmo Bayesiano[1], que fez com que o entendimento completo do algoritmo levasse um bom tempo.

Em segundo lugar, ao longo de todo o artigo original, várias constantes e até mesmo variáveis ficam sem uma determinação clara de qual valor devemos utilizar para replicar o algoritmo. Esse caso já foi descrito na própria descrição do artigo, mas foram coisas que realmente tornaram o processo oneroso.

Por fim, devido à complexidade computacional dos algoritmos estudados e o número de execuções necessárias para as simulações, o tempo de simulação foi a maior dificuldade desse trabalho. Todas as simulações foram rodadas em um computador pessoal, fazendo com que as simulações chegassem a durar horas. Para diminuir um pouco esse tempo, várias técnicas de programação foram utilizadas para otimização do tempo de execução, como por exemplo, programação dinâmica no cálculo das probabilidades.

## 4.5 Resumo

Neste capítulo foi apresentado o simulador utilizado para investigar e analisar as duas abordagens propostas para o algoritmo bayesiano[1]. Primeiramente, para garantir a corretude do simulador, foram gerados gráficos com o mesmo cenário de gráficos exibidos em [1]. Como os gráficos estão muito semelhantes, comprovamos que a implementação tanto do simulador quanto do algoritmo tal qual descrito pelos autores está correto.

Como forma de analisar e avaliar mais facilmente as fraquezas e potencialidades de cada método, os resultados foram todos colocados na forma de gráficos e os resultados comentados logo abaixo cada imagem. Várias análises foram feitas ao longo do capítulo, mas no geral, os três algoritmos possuem os mesmos resultados tendo em vista o mesmo cenário.

Por fim, listamos as dificuldades encontradas ao longo do desenvolvimento desse trabalho, assim como os recursos que utilizamos para ultrapassar essas dificuldades.

## CAPÍTULO 5

### CONCLUSÕES

Sistemas RFID já são uma realidade na atualidade e estão cada vez mais sendo aprimorados por estudos acadêmicos e até mesmo financiados por empresas que possuem interesse nos possíveis resultados dessa tecnologia.

Quando completamente difundida, essa tecnologia trará inúmeros benefícios para a vida humana e estima-se que será um dos pilares para o que se chama atualmente de internet das coisas. Várias aplicações já utilizam o RFID como forma de controle e para facilitar o gerenciamento de seus insumos, como o agronegócio, produção de uvas, criação de animais, pedágios, entre vários outros.

No âmbito geral, o RFID é visto como o sucessor imediato do código de barras, e já vem sendo implantado com sucesso em aeroportos, livrarias e grandes redes de supermercados. Com o potencial da tecnologia, resta aos pesquisadores resolver e eliminar problemas que ainda impedem o uso completamente difundido da tecnologia, como por exemplo, a ocorrência das colisões no processo de leitura das etiquetas.

Esse trabalho tem foco justamente na solução desse problema, as colisões. Ao longo de todo o texto foram explicados e detalhados algoritmos e protocolos que possibilitam uma diminuição da ocorrência de colisões mas muito ainda precisa ser feito. Devido à latente necessidade, muitos pesquisadores estão se dedicando completamente à tarefa de achar soluções para o problema das colisões em sistemas RFID.

Como foco principal de análise desse trabalho, verificou-se o desempenho sob várias métricas de duas abordagens que solucionam o problema de fatores infinitos na estimativa da quantidade de etiquetas em algoritmos que utilizam o Teorema de Bayes como aspecto principal nessa estimativa. Todos os resultados analisados foram obtidos com a construção de um simulador capaz de executar exatamente os mesmos cenários descritos em [1] e comparar com o resultado utilizando as abordagens propostas.

Ao fim da análise, viu-se que as abordagens propostas geram um menor erro médio percentual na estimativa de etiquetas, e portanto, são uma boa melhoria no algoritmo descrito em [1]. Todos os resultados foram descritos no formato de gráficos para facilitar a análise e estão todos comentados. Pôde-se observar que variando a função de risco utilizada, a melhor solução também muda, porém, sempre são as abordagens propostas melhores que o desempenho do original na métrica do erro médio percentual.

Por fim, pode-se afirmar que em poucos anos os sistemas RFID estarão completamente difundidos em nossas atividades diárias e serão transparentes para os usuários comuns.

## REFERÊNCIAS BIBLIOGRÁFICAS

[1] H. Wu and Y. Zeng, “Bayesian Tag Estimate and Optimal Frame Length for Anti-Collision Aloha RFID System,” *IEEE Transactions on Automation Science and Engineering*, vol. 7, no. 4, pp. 963–969, October 2010.

[2] W.-T. Chen, “An accurate tag estimate method for improving the performance of an RFID anticollision algorithm based on dynamic frame length ALOHA,” *IEEE Trans. Autom. Sci. Eng.*, vol. 6, no. 1, pp. 9–15, Jan. 2009.

[3] H. Vogt, “Efficient Object Identification with Passive RFID Tags,” in *Proceedings of the First International Conference on Pervasive Computing*. London, UK: Springer-Verlag, 2002, pp. 98–113.

[4] F. C. Schoute, “Dynamic frame length ALOHA,” *IEEE Trans. Commun.*, vol. 31, no. 4, pp. 565–568, Apr. 1983.

[5] J.-B. Eom and T.-J. Lee, “Accurate Tag Estimation for Dynamic Framed-slotted ALOHA in RFID Systems,” *IEEE Communications Letters*, vol. 14, pp. 60–62, January 2010.

[6] K. Finkenzerler, *RFID Handbook: Fundamentals and Applications in Contactless Smart Cards and Identification*, 2nd ed. New York, NY, USA: John Wiley & Sons, Inc., 2003.

[7] D.-H. Shih, P.-L. Sun, D. C. Yen, and S.-M. Huang, “Taxonomy and Survey of RFID Anti-collision Protocols,” *Computer Communications*, vol. 29, no. 11, pp. 2150–2166, 2006.

[8] J. Andrade and P. Gonçalves, “Uma Função de Cálculo de Tamanho de Frames para o Protocolo DFSA em Sistemas RFID,” in *XVI Workshop de Gerência e Operação de Redes e Serviços (WGRS)*, Campo Grande, MS, Brazil, May 2011, pp. 61–74.

[9] THAMGAM, C. V. World's smallest and thinnest RFID tag is powder made by Hitachi, 2007. Disponível em: [http://digitaljournal.com/article/244321/World\\_s\\_smallest\\_and\\_thinnest\\_RFID\\_tag\\_is\\_powder\\_made\\_by\\_Hitachi](http://digitaljournal.com/article/244321/World_s_smallest_and_thinnest_RFID_tag_is_powder_made_by_Hitachi). Último acesso em: 8 Novembro 2014.

[10] J.-R. Cha and J.-H. Kim, “Novel Anti-collision Algorithms for Fast Object Identification in RFID System,” in *Proceedings of the 11th International Conference on Parallel and Distributed Systems*, vol. 2, July 2005, pp. 63–67.

[11] J. F. Kurose and K. W. Ross, *Redes de Computadores e a Internet. Uma Abordagem Top-down*, 5th ed. Pearson Addison Wesley, 2010.

[12] TONG, Q., ZOU, X., and TONG, H. 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, pp. 384–388, 2009.

[13] A. C. Rizzo, Uma análise dos estimadores Tong e Chen para o protocolo anticolisão DFSA em sistemas RFID, Projeto de conclusão de curso, UFPE, Julho 2012

[14] ROBERTI, M. The History of RFID Technology, 2007 Disponível em: <<http://www.rfidjournal.com/article/view/1338/1>>. Último acesso em: 8 Novembro 2014.