



UNIVERSIDADE FEDERAL DO RIO GRANDE DO NORTE  
CENTRO DE TECNOLOGIA

PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA E  
DE COMPUTAÇÃO



***Group Sequential Communication (GSC):  
Especificação e Análise de Desempenho de um  
Mecanismo de Comunicação de Tempo Real  
Compatível ao Padrão IEEE 802.11/11e  
Aplicado à Automação Industrial***

**Raimundo Viégas Junior**

Orientador: Prof. Dr. Luiz Affonso H. Guedes de Oliveira (DCA/UFRN)

Co-orientador: Prof. Dr. Francisco Vasques (FEUP/UP)

**Tese de Doutorado** apresentada ao Programa de Pós-Graduação em Engenharia Elétrica e de Computação da UFRN (área de concentração: Engenharia de Computação) como parte dos requisitos para obtenção do título de Doutor em Ciências.

Número de ordem PPgEE: D55  
Natal, RN, fevereiro de 2010

Divisão de Serviços Técnicos

Catálogo da publicação na fonte. UFRN / Biblioteca Central Zila Mamede

Viégas Junior, Raimundo.

Group Sequential Communication (GSC): Especificação e Análise de Desempenho de um Mecanismo de Comunicação de Tempo Real Compatível ao Padrão IEEE 802.11/11e e Aplicado à Automação Industrial / Raimundo Viégas Junior. - Natal, RN, 2010

121 f.

Orientador: Luiz Affonso H. Guedes de Oliveira.

Co-orientador: Francisco Vasques.

Tese (Doutorado) - Universidade Federal do Rio Grande do Norte. Centro de Tecnologia. Programa de Pós-Graduação em Engenharia Elétrica e de Computação.

1. Automação Industrial - Tese. 2. IEEE 802.11e - Tese. 3. Informática Industrial - Tese. 4. Comunicação em Tempo Real - Tese. 5. Qualidade de Serviço - Tese. 6. Redes Sem Fio - Tese. I. Oliveira, Luiz Affonso H. Guedes de. II. Vasques, Francisco. III. Universidade Federal do Rio Grande do Norte. IV. Título.

RN/UF/BCZM

CDU 681.5(043.2)

***Group Sequential Communication (GSC):***  
**Especificação e Análise de Desempenho de um**  
**Mecanismo de Comunicação de Tempo Real**  
**Compatível ao Padrão IEEE 802.11/11e**  
**Aplicado à Automação Industrial**

**Raimundo Viégas Junior**

Tese de Doutorado aprovada em 11 de fevereiro de 2010 pela banca examinadora composta pelos seguintes membros:

---

Prof. Dr. Luiz Affonso H. Guedes de Oliveira (orientador) ..... DCA/UFRN

---

Prof. Dr. Francisco Vasques (co-orientador) ..... FEUP/UP

---

Prof. Dr. Carlos Barros Montez ..... DAS/UFSC

---

Prof. Dr. Luiz Felipe de Queiroz Silveira ..... DI/IFRN

---

Prof. Dr. Adilson Barboza Lopes ..... DIMAP/UFRN

---

Prof. Dr. Marcos Cesar Madruga Alves Pinheiro ..... DIMAP/UFRN



*A Deus, por mais esta oportunidade  
de crescimento como ser humano.*



---

# Agradecimentos

---

A minha esposa Simone Rose e aos meus filhos Gabriel Victor, Micael Augusto e Samuel Felipe, pelo amor e força que me foi de fundamental importância para obtenção de êxito nesta missão.

Aos meus pais, Raimundo Viégas e Joana Célia pelo grande exemplo de vida, e aos meus irmãos; Ruy Felipe, José Augusto e Renata pelo apoio.

Ao meu orientador, Prof. Dr. Luiz Affonso Guedes pela amizade e valiosa orientação na condução do desenvolvimento deste trabalho.

Aos professores, Francisco Vasques e Paulo Portugal pela supervisão técnica do estágio de doutorado sanduíche realizado na Faculdade de Engenharia da Universidade do Porto (FEUP) e sugestões que enriqueceram o desenvolvimento deste trabalho.

Aos colegas do Laboratório de Informática Industrial (L.I.I), em especial a Ricardo Valentim, Carlos Viegas, Marcelo Nobre, Ivanovitch Silva e Vinicius Machado pela troca de experiências no decorrer do doutorado.

Aos colegas professores da Faculdade de Computação da Universidade Federal do Pará (FACOMP-UFGA) e Departamento de Controle e Automação da Universidade Federal do Rio Grande do Norte (DCA-UFRN) pelo incentivo.

À CAPES, pelo apoio financeiro.





---

# Resumo

---

Esta tese propõe a especificação e análise de desempenho de um mecanismo de comunicação de tempo real compatível com o Padrão IEEE 802.11/11e, chamado *Group Sequential Communication (GSC)*. O *GSC* apresenta um melhor desempenho quando comparado ao mecanismo *HCCA* para tratar pequenos pacotes de dados, além de adotar uma abordagem descentralizada do controle de acesso ao meio baseado no conceito produtor/consumidor. O objetivo principal da proposta é a redução de *overheads* da rede, oriundos de quadros de *Polling*, *ACK* e *QoS Null* trocados entre o controlador híbrido e as estações no *HCCA* padrão. O mecanismo *GSC* elimina o uso de quadros de *Polling* utilizados pelo escalonador do *HCCA*, através de um procedimento de *Virtual Token Passing* entre os membros do grupo de tempo real, a quem é garantida alta prioridade de acesso ao meio de forma sequencial. A fim de melhorar a confiabilidade da proposta em ambientes ruidosos é apresentado um esquema de recuperação de erro chamado algoritmo de segunda chance. Este esquema é baseado em uma estratégia de reconhecimento em bloco das mensagens enviadas, com possibilidade de serem retransmitidas quando não recebidas com sucesso. Desta forma, o mecanismo *GSC* mantém o tráfego das estações de tempo real entre os diversos dispositivos compatível ao Padrão IEEE 802.11/11e, com o uso otimizado da banda e variações mínimas de atraso médio entre as entregas dos pacotes de dados pertencentes à rede sem fio. Para propósito de validação da proposta, os mecanismos *GSC* e *HCCA* foram implementados através de *softwares* de simulação de redes desenvolvido em C/C++ e os resultados de desempenho foram comparados. Os experimentos mostram a eficiência do mecanismo *GSC*, principalmente em cenários de comunicações industriais.

**Palavras-chave:** IEEE 802.11e, Informática Industrial, Comunicação em Tempo Real, Qualidade de Serviço, Redes sem Fios.



---

# Abstract

---

This thesis proposes the specification and performance analysis of a real-time communication mechanism for IEEE 802.11/11e standard. This approach is called Group Sequential Communication (GSC). The GSC has a better performance for dealing with small data packets when compared to the HCCA mechanism by adopting a decentralized medium access control using a publish/subscribe communication scheme. The main objective of the thesis is the HCCA overhead reduction of the Polling, ACK and QoS Null frames exchanged between the Hybrid Coordinator and the polled stations. The GSC eliminates the polling scheme used by HCCA scheduling algorithm by using a Virtual Token Passing procedure among members of the real-time group to whom a high-priority and sequential access to communication medium is granted. In order to improve the reliability of the mechanism proposed into a noisy channel, it is presented an error recovery scheme called second chance algorithm. This scheme is based on block acknowledgment strategy where there is a possibility of retransmitting when missing real-time messages. Thus, the GSC mechanism maintains the real-time traffic across many IEEE 802.11/11e devices, optimized bandwidth usage and minimal delay variation for data packets in the wireless network. For validation purpose of the communication scheme, the GSC and HCCA mechanisms have been implemented in network simulation software developed in C/C++ and their performance results were compared. The experiments show the efficiency of the GSC mechanism, especially in industrial communication scenarios.

**Keywords:** IEEE 802.11e, Industrial Informatics, Real-Time Communications, Qos, Wireless Networks.



---

# Sumário

---

<b>Sumário</b>	<b>i</b>
<b>Lista de Figuras</b>	<b>iii</b>
<b>Lista de Tabelas</b>	<b>v</b>
<b>Lista de Publicações</b>	<b>vii</b>
<b>Lista de Símbolos e Abreviaturas</b>	<b>ix</b>
<b>1 Introdução</b>	<b>1</b>
1.1 Redes Sem Fio Aplicada à Indústria . . . . .	1
1.2 Motivação . . . . .	5
1.3 Objetivo Geral . . . . .	6
1.4 Objetivos Específicos . . . . .	6
1.5 Contribuições da Tese . . . . .	7
1.6 Organização do Trabalho . . . . .	7
<b>2 Redes Padrão IEEE 802.11/11e</b>	<b>9</b>
2.1 Arquiteturas . . . . .	9
2.2 Camada Física do Padrão IEEE 802.11 . . . . .	12
2.2.1 IEEE 802.11a . . . . .	13
2.2.2 IEEE 802.11b . . . . .	14
2.2.3 IEEE 802.11g . . . . .	15
2.2.4 IEEE 802.11n . . . . .	15
2.3 Subcamada de Controle de Acesso ao Meio ( <i>MAC</i> ) . . . . .	16
2.3.1 Função de Coordenação Distribuída ( <i>DCF</i> ) . . . . .	16
2.3.2 Função de Coordenação Pontual ( <i>PCF</i> ) . . . . .	25
2.4 Emenda IEEE 802.11e . . . . .	29
2.4.1 Função de Coordenação Híbrida ( <i>HCF</i> ) . . . . .	29
2.4.2 Acesso Baseado em Contenção ( <i>EDCA</i> ) . . . . .	31

2.4.3	Acesso Controlado ao Canal ( <i>HCCA</i> ) . . . . .	33
2.4.4	Controle de Admissão . . . . .	37
2.4.5	Escalonador de Referência . . . . .	38
2.4.6	Melhorias Obtidas pela Emenda IEEE 802.11e . . . . .	41
2.4.7	Motivação para Mudanças no Mecanismo <i>HCCA</i> . . . . .	42
2.4.8	Considerações sobre as Redes Padrão IEEE 802.11/11e . . . . .	45
<b>3</b>	<b>Trabalhos Relacionados</b>	<b>47</b>
3.1	Comunicação de Tempo Real para Redes Sem Fio . . . . .	47
3.1.1	Soluções Baseadas no <i>DCF/EDCA</i> . . . . .	51
3.1.2	Soluções Baseadas no <i>HCCA</i> . . . . .	54
3.1.3	Soluções Baseadas em Modificações no <i>HCCA</i> . . . . .	57
3.1.4	Soluções para Aplicações Industriais . . . . .	59
3.1.5	Considerações sobre Tempo Real para Redes Sem Fio . . . . .	62
<b>4</b>	<b>O Mecanismo <i>GSC</i></b>	<b>65</b>
4.1	Visão Geral . . . . .	65
4.1.1	Especificação . . . . .	70
4.1.2	Quadro de <i>Beacon</i> . . . . .	78
4.1.3	Inserção de Estações . . . . .	81
4.1.4	Remoção de Estações . . . . .	82
4.2	Modelagem do Sistema . . . . .	83
4.2.1	Modelagem do Tráfego . . . . .	84
4.2.2	Controle de Admissão . . . . .	86
4.2.3	Modelo de Erro do Canal . . . . .	89
4.2.4	Análise de Erros . . . . .	91
4.2.5	Aspectos Relacionados à Robustez . . . . .	93
<b>5</b>	<b>Análise de Desempenho do <i>GSC</i></b>	<b>97</b>
5.1	Descrição dos Cenários de Comunicação Industrial . . . . .	98
5.1.1	Avaliação dos Mecanismos <i>GSC</i> e <i>HCCA</i> . . . . .	103
5.1.2	Simulação da Rede Utilizando Canal Livre de Ruídos . . . . .	104
5.1.3	Simulação da Rede Utilizando Canal com Ruídos . . . . .	109
<b>6</b>	<b>Conclusão e Trabalhos Futuros</b>	<b>113</b>
	<b>Referências Bibliográficas</b>	<b>115</b>

---

# Lista de Figuras

---

2.1	Rede Modo Infra-estruturado. . . . .	10
2.2	Rede Modo <i>Ad-hoc</i> . . . . .	10
2.3	Camadas Física e de Enlace com Respectivas Subcamadas. . . . .	12
2.4	Espaçamento Entre Quadros do Padrão IEEE 802.11. . . . .	17
2.5	Exemplo de Operação <i>DCF</i> . . . . .	18
2.6	Problema da Estação Oculta. . . . .	20
2.7	Problema da Estação Exposta. . . . .	20
2.8	Comunicação entre Estações Utilizando o Mecanismo <i>RTS/CTS</i> . . . . .	22
2.9	Formato do Quadro Padrão IEEE 802.11/11e. . . . .	23
2.10	Formato Geral do Quadro <i>MAC</i> do IEEE 802.11/11e. . . . .	24
2.11	Coexistência da Função <i>PCF</i> e <i>DCF</i> no Superquadro. . . . .	25
2.12	Transmissões Realizadas Durante o <i>CFP</i> . . . . .	27
2.13	Exemplo de Diminuição do <i>CFP</i> . . . . .	27
2.14	Arquitetura da Camada <i>MAC</i> do IEEE 802.11e. . . . .	30
2.15	Esquema Básico do <i>EDCA</i> . . . . .	32
2.16	Comparação entre <i>DCF</i> e <i>EDCA</i> . . . . .	33
2.17	Superquadro Padronizado pela Emenda IEEE 802.11e. . . . .	34
2.18	Formato do Quadro <i>TSPEC</i> segundo o Padrão IEEE 802.11e. . . . .	35
2.19	Formato do Campo TS Info Segundo o Padrão IEEE 802.11e. . . . .	36
2.20	Operação do Escalonador de Referência. . . . .	40
2.21	Mecanismo de <i>Polling</i> Utilizado Pelo <i>HCCA</i> para um Fluxo de Dados. . . . .	43
2.22	Desempenho do Mecanismo de <i>Polling</i> do <i>HCCA</i> . . . . .	45
3.1	Abordagens Para Comunicação de Tempo Real em <i>WLAN</i> . . . . .	50
4.1	Cenário Proposto para o <i>GSC</i> . . . . .	66
4.2	Funcionamento Básico do Mecanismo <i>GSC</i> . . . . .	68
4.3	Economia de Banda do Mecanismo <i>GSC</i> . . . . .	68
4.4	Esquema de Recuperação de Erro do Mecanismo <i>GSC</i> . . . . .	69
4.5	Formato do Quadro de <i>Block Acknowledgment</i> . . . . .	73

4.6	Intervalo de Repetição do Mecanismo <i>GSC</i> . . . . .	74
4.7	Formato do Quadro de <i>Beacon</i> . . . . .	78
4.8	Máquina de Estados do IEEE 802.11. . . . .	81
4.9	Topologia de Rede sem Fio. . . . .	84
4.10	Avaliação do Limite Superior/Inferior para o Intervalo de Serviço. . . . .	86
4.11	Modelo do Canal de <i>Gilbert-Elliot</i> Representado por Cadeia de <i>Markov</i> . . . . .	90
5.1	Cenário de Supervisão. . . . .	100
5.2	Diagrama Temporal do <i>GSC/HCCA</i> para Cenário de Supervisão. . . . .	101
5.3	Cenário de Controle. . . . .	102
5.4	Diagrama Temporal do <i>GSC/HCCA</i> para Cenário de Controle. . . . .	102
5.5	Porcentagem do SI para Canal Livre de Ruídos. . . . .	105
5.6	Latência Média de Transmissão para Canal Livre de Ruídos. . . . .	106
5.7	Tamanho dos Pacotes Versus Numero de Estações de Tempo Real. . . . .	107
5.8	Duração do <i>CFP</i> em função da variação do <i>PER</i> . . . . .	110
5.9	Latência Média de Transmissão para <i>PER</i> Variável. . . . .	110



---

# Lista de Tabelas

---

2.1	Taxas de Transmissão do Padrão IEEE 802.11a. . . . .	13
2.2	Canal e Frequência para o Padrão IEEE 802.11b/g. . . . .	14
2.3	Comparativo entre os Padrões da Família IEEE 802.11. . . . .	16
2.4	Mapeamento das Prioridades de Usuário para as Categorias de Acesso. . .	31
2.5	Tempo de Transmissão do Pacote de Dados de 40 bytes. . . . .	43
4.1	Corpo do Quadro de <i>Beacon</i> . . . . .	78
4.2	Corpo do Quadro de Requisição de Associação. . . . .	82
5.1	Parâmetros da Camada Física do IEEE 802.11b. . . . .	103
5.2	Parâmetros de Simulação da Rede. . . . .	104
5.3	Tempo de Transmissão do <i>HCCA</i> e <i>GSC</i> . . . . .	108
5.4	Parâmetros de Simulação para Cenário com Ruído . . . . .	109
5.5	Taxa de Sucesso dos Pacotes Transmitidos. . . . .	111



---

# Lista de Publicações

---

No decorrer do curso de doutorado foram elaboradas seis publicações em conferências e um artigo submetido à revista científica. Estas publicações estão ordenadas em ordem da mais importante e recente até a mais antiga, revelando a evolução gradual dos estudos até consolidação da proposta sugerida neste trabalho.

- 1- VIÉGAS JR, R.; PORTUGAL, P.; GUEDES, L. A.; VASQUES, F.; MORAES, R. *Specification and Performance Analysis of a HCCA-Based Approach Specifically Suited For IEEE 802.11e Real-Time Communication*. Paper code: COMNET-D-09-3958, COMPUTER NETWORKS journal, Elsevier Publisher, Status: Under Revision, 2010.
- 2- VIÉGAS JR, R.; PORTUGAL, P.; GUEDES, L. A.; VASQUES, F. *A Proposal of Real-Time Publish-Subscribe Scheme Compatible with 802.11e Wireless Networks*. 35th Annual Conference of the IEEE Industrial Electronics Society - IECON 2009, Porto - Portugal, 2009.
- 3- VIÉGAS JR, R.; MORAES, R.; GUEDES, L. A.; VASQUES, F. *GSC: A Real-Time Communication Scheme for IEEE 802.11e Industrial Systems*. 7th IFAC International Conference on Fieldbuses & Networks in Industrial & Embedded System, 2007, Toulouse - França, 2007. p. 111-118.
- 4- VIÉGAS JR, R.; VALENTIM, R.; TEIXERA, D.; GUEDES, L. A. *Performance Analysis of Protocols: UDP and RAW Ethernet to Real-Time Networks*. International Workshop on Telecommunications - IWT 2007. Santa Rita do Sapucaí - MG - Brasil : Instituto Nacional de Telecomunicações - Inatel, 2007.
- 5- VIÉGAS JR, R.; VALENTIM, R.; TEIXERA, D.; GUEDES, L. A. *Analysis of Protocols to Ethernet Automation Networks*. IEEE International Joint Conference - SICE - ICCAS 2006, Busan - Coréia, 2006. v. 01. p. 4981-4985.

- 6- VIÉGAS JR, R.; VALENTIM, R.; TEIXERA, D.; GUEDES, L. A. *Performance Measurements of Protocols to Ethernet Real-Time Applications*. 11th IEEE International Conference on Emerging Technologies and Factory Automation - ETFA 2006, Praga - Rep. Theca, 2006. v. 01. p. 1352-1355.
  
- 7- VIÉGAS JR, R.; VALENTIM, R.; FONSECA, A.; GUEDES, L. A. *Análise de Desempenho de Protocolos em Redes Ethernet para Aplicações de Tempo Real*. GCMM'2006 - Congresso Global em Engenharia de Manufatura e Administração - São Paulo - Brasil. ISSN/ISBN: 8589549356.

---

# Lista de Símbolos e Abreviaturas

---

AC	Access Category
ACK	Acknowledgment
AIFS	Arbitration Interframe Space
AIFSN	Arbitration Interframe Space Number
AP	Access Point
APSD	Automatic Power Save Delivery
ARROW	Adaptive Resource Reservation Over WLANs
BA	Block Acknowledgment
BB	Black-Burst
BER	Bit Error Rate
BPSK	Binary Phase-Shift Keying
BSA	Basic Service Area
BSS	Basic Service Set
CAP	Controlled Access Phase
CBR	Constant Bit Rate
CCA	Clear Channel Assessment Signal
CCK	Complementary Code Keying
CF	Coordination Function
CFP	Contention Free Period

CP	Contention Period
CRC	Cyclic Redundancy Character
CSMA	Carrier Sense Multiple Access
CSMA/CA	Carrier Sense Multiple Access/Collision Avoidance
CSMA/CD	Carrier Sense Multiple Access/Collision Detection
CTS	Clear to Send
CW	Contention Window
CWmax	Contention Windows Maximum
CWmin	Contention Windows Minimum
DCF	Distributed Coordination Function
DFIR	Diffused Infrared
DIFS	DCF Inter Frame Spacing
DLP	Direct Link Protocol
DS	Distribution System
DSSS	Direct Sequence Spread Spectrum
EB	Energy-Burst
EDCA	Enhanced Distributed Channel Access
ESS	Extended Service Set
FCS	Frame Check Sequence
FEC	Forward Error Correction
FHSS	Frequency Hopping Spread Spectrum
FIFO	First In, First Out
Gid	Group Identification

GSC	Group Sequential Communication
HC	Hybrid Coordination
HCCA	HCF - Controlled Channel Access
HCF	Hybrid Coordination Function
HR/DSSS	High Rate Direct Sequence Spread Spectrum
IBSS	Independent BSS
IEEE	Institute of Electrical and Electronic Engineers
IFS	Interframe Spaces
IP	Internet Protocol
ISM	Industrial, Scientific and Medical
LLC	Logical Link Control
LR-WPAN	Low-Rate Personal Area Network
MAC	Medium Access Control
MIMO	Multiple-Input Multiple-Output
MPDU	MAC Package Data Unit
MSDU	MAC Service Data Unit
MSI	Maximum Service Interval
NAV	Network Allocation Vector
NDIS	Network Driver Interface Specification
NPHCCA	Non-Polling Based HCCA
OFDM	Orthogonal Frequency Division Multiplexing
OTcl	Object Tool Command Language
PC	Point Coordinator

PCF	Point Coordination Function
PER	Packet Error Rate
PHY	PHY layer
PIFS	PCF Inter Frame Spacing
PLC	Programmable Logic Controller
PLCP	Physical Layer Convergence Procedure
PMD	Physical Medium Dependent
PVB	Parcial Virtual Bitmap
QAM	Quadrature Amplitude Modulation
QAP	QoS Access Point
QBSS	QoS Basic Service Set
QoS	Quality of Service
QPSK	Quadrature Phase-Shift Keying
QSTA	QoS Station
RTBlockAck	Real-Time Block Acknowledgment
RTH	Real-Time HCCA
RTS	Request to Send
SC	Serial Counter
SCADA	Supervisory Control and Data Aquisition
SETT-EDD	Scheduling Based on Estimated Transmission Times - Earliest Due Date
SFD	Start Frame Delimiter
SI	Service Interval
SIFS	Short Inter Frame Spacing



SP	Service Period
ST	aSlotTime
STA	Station
TBTT	Target Beacon Transmission Time
TC	Traffic Category
TDMA	Time Division Multiple Access
TIM	Traffic Information Map
TS	Traffic Stream
TSPEC	Traffic Specification
TXOP	Transmission Opportunity
VBR	Variable Bit Rate
VoIP	Voice over Internet Protocol
VTP	Virtual Token Passing
WLAN	Wireless Local Area Network
WMM	Wireless MultiMedia
WPAN	Wireless Personal Area Network



---

# Capítulo 1

## Introdução

---

Este capítulo apresenta uma introdução às redes sem fio aplicado à indústria, destacando suas vantagens e os desafios tecnológicos que devem ser superados para a efetiva viabilização desta solução para automação de processos industriais. Ao final deste capítulo serão apresentados a motivação, o objetivo geral, os objetivos específicos e as contribuições da tese, além da organização do trabalho.

### 1.1 Redes Sem Fio Aplicada à Indústria

Atualmente, as tecnologias de comunicação sem fio estão sendo empregadas em um número cada vez maior de aplicações em áreas que necessitam de comunicação de tempo real, que é o caso de aplicações industriais [Willig et al. 2005] [Trsek et al. 2006] [Cena et al. 2008]. Essas tecnologias oferecem muitas vantagens sobre as redes industriais cabeadas tradicionais, tais como: mobilidade, flexibilidade, escalabilidade, rápida e fácil instalação e custo reduzido para ambientes dinâmicos que exigem frequentes movimentações, adições e alterações dos dispositivos de comunicação [Ye et al. 2000] [Willig et al. 2005]. No entanto, em comparação com os seus correspondentes cabeados, o projeto e execução de redes sem fio para a indústria enfrentam determinados desafios, incluindo interferências no canal de comunicação, largura de banda, potência de transmissão, segurança das informações e controle de acesso ao meio [Willig 2008].

As principais áreas de interesse em tecnologias de redes sem fios para ambientes industriais são as que não necessitam de nenhum tipo de licenciamento de frequências, a exemplo da utilização pública das bandas Médicas, Científicas e Industrial ou *ISM (Industrial, Scientific and Medical)* [Willig et al. 2005]. Essas soluções incluem tecnologias de redes pessoais ou *WPAN (Wireless Personal Area Network)*, definida pelo Instituto de Engenheiros Eletricistas e Eletrônicos (*Institute of Electrical and Electronic Engineers - IEEE*) tais como IEEE 802.15.1 e IEEE 802.15.4, bem como a tecnologia de redes locais

sem fio ou *WLAN* (*Wireless Local Area Network*) pertencente ao padrão IEEE 802.11 [Willig et al. 2005] [Willig 2008].

O Padrão IEEE 802.15.1 também conhecido por *Bluetooth* foi projetado inicialmente para suportar redes simples de dispositivos e periféricos pessoais como celulares, *PDA*s, computadores e *mouses* e tem o objetivo de substituir cabos utilizados em dispositivos móveis por transmissão sem fio de curto alcance [Willig et al. 2005]. Através do *Bluetooth* são formados grupos de 2 a 8 dispositivos sem fios, organizados em uma *piconet*. Diversas *piconets* podem ser interligadas, formando-se uma *scatternet*. A coexistência de diversas *piconets* ocorre devido a cada uma delas utilizar uma faixa diferente de frequências, pois o *Bluetooth* divide a faixa de 2,4 GHz em 79 canais de 1 MHz [Willig et al. 2005]. Em cada *piconet* existe apenas um nó que assume o papel de mestre da rede enviando mensagens de *polling* para os demais dispositivos com uma taxa suportada máxima próxima a 723 Kbps [Castano et al. 2003].

O Padrão IEEE 802.15.4 descreve a camada física e de controle de acesso ao meio para redes de sensores de baixa velocidade com pequeno consumo de energia. Um Padrão implementado sobre o IEEE 802.15.4 é o *ZigBee* e compreende a camada de rede e aplicação [Willig 2008]. Desenvolvido como alternativa para comunicação em redes que não necessitem de soluções mais complexas para seu controle, trata-se de uma tecnologia simples, que utiliza um protocolo de comunicação com características específicas e utiliza pequenos pacotes de dados com curto alcance, projetado para oferecer flexibilidade no controle de vários tipos de dispositivos com taxas de até 250 Kbps.[Willig 2008].

Como alternativa às tradicionais redes industriais cabeadas e a utilização de soluções como *Bluetooth* e *ZigBee*, surgem implementações que fazem uso das redes sem fio compatível ao padrão IEEE 802.11 (*WLAN*). Atualmente esse padrão é a base do *Wi-Fi*<sup>1</sup> que é a tecnologia de rede local sem fio mais utilizada no mundo [Al-Karaki & Chang 2004] [Moyne & Tilbury 2007] [Gao et al. 2005], que possui vários fornecedores a baixo custo.

A *WLAN* é definida pela especificação do padrão IEEE 802.11 [IEEE 1999a] e tem o método original de acesso ao meio baseado no protocolo *Carrier Sense Multiple Access with Collision Avoidance* (*CSMA/CA*), que é utilizado no Controle de Acesso ao Meio (*Medium Access Control - MAC*) através da Função de Coordenação Distribuída (*Distributed Coordination Function - DCF*), onde todas as estações competem entre si pelo meio de transmissão. Opcionalmente, pode-se utilizar a Função de Coordenação Pontual (*Point Coordination Function - PCF*), que utiliza um mecanismo de interrogação (*Polling*) das estações através de um coordenador de rede residente no Ponto de Acesso (*Access Point*

---

<sup>1</sup>Wi-Fi é uma marca registrada pela Wi-Fi Alliance para tecnologia de redes sem fios baseadas no padrão IEEE 802.11.

- AP). Porém estes mecanismos presentes no Padrão IEEE 802.11 não garantem qualidade de serviço devido principalmente pela não diferenciação de tipo de tráfego na rede. Logo, não suportam os requisitos das redes de automação industrial que exigem suporte de tempo real [Sauter & Vasques 2006]. Todavia, apresenta-se como uma tecnologia bastante interessante para este contexto, devido ao bom desempenho (taxas de transmissão que variam de 1 Mbps a 54 Mbps, dependendo do padrão utilizado), baixo custo e alta interoperabilidade [Willig et al. 2005] [Ni 2005].

Atualmente as redes padrão IEEE 802.11 podem tolerar erros, mas não impõem qualquer restrição temporal rígida para o tráfego de pacotes e apresentam maiores desafios em relação às redes cabeadas, onde podemos destacar:

- A largura de banda de uma rede sem fio é limitada, pois o canal de comunicação tem que ser compartilhado por vários dispositivos que dividem o mesmo meio de transmissão [Willig 2008].
- Em redes sem fio não é possível transmitir e monitorar o canal de comunicação simultaneamente, pois o custo desta implementação é inviável [Grilo et al. 2003]. Esta é uma característica que tem repercussão direta no uso do protocolo de controle de acesso ao meio para aplicações que necessitam de confiabilidade e determinismo para garantir restrições de tempo real.
- As redes cabeadas podem ser organizadas em diferentes topologias de acordo com as ligações físicas entre as estações, diferente das redes sem fios que o sinal de uma estação propaga-se para todas as direções, equivalendo-se a uma topologia de barramento [Willig et al. 2005].
- As redes sem fios estão expostas às taxas de erros no canal de comunicação muito maiores e variáveis no tempo do que as redes cabeadas, devido a fatores como interferências, atenuação e efeitos da reflexão e da refração das ondas eletromagnéticas [Willig et al. 2005].

Algumas destas dificuldades já foram superadas em parte pelas pesquisas desenvolvidas tanto no nível da camada física quanto no nível da camada de enlace de dados. Atualmente já existem redes de comunicação sem fio capazes de suportar tanto o tráfego síncrono, como o tráfego assíncrono proveniente de aplicações convencionais, admitindo diferentes níveis de prioridade [Trsek et al. 2006]. No entanto, o controle de acesso ao meio e os erros no canal de comunicação continuam a ser um obstáculo à utilização

das redes sem fios em ambientes industriais, devido à baixa tolerância a falhas associada às restrições temporais exigidas pelas aplicações industriais de tempo real [Moyné & Tilbury 2007].

Devido às características diversificadas dessas aplicações industriais, provavelmente será necessário operar com redes múltiplas *WLAN/WPAN* simultaneamente em áreas diferentes ou sobrepostas na mesma planta industrial. Devido a este fator, a coexistência de redes múltiplas de mesmo ou variados tipos necessita ser investigada com relação à padronização das comunicações, mantendo-se os padrões de conectividade com os diversos tipos de protocolos de comunicação disponível no mercado [Willig et al. 2005] [Moraes et al. 2007]. Porém, garantir as restrições temporais para o tráfego de tempo real em ambientes industriais abertos que são uma característica de redes sem fio é uma tarefa difícil. A principal dificuldade é que se não for acertado primeiramente um conjunto de protocolo de comunicação comum aos dispositivos envolvidos, a restrição temporal imposta não pode ser prevista na configuração e nem pode ser controlada efetivamente em tempo de execução do sistema [Moraes et al. 2007].

A fim de agregar qualidade de serviço (*Quality of Service - QoS*) ao padrão IEEE 802.11 houve a necessidade de criar uma emenda chamada de IEEE 802.11e [IEEE 2005], na qual é definida uma função de coordenação híbrida (*Hybrid Coordination Function - HCF*) que combina e melhora os dois métodos de acesso legado (*DCF/PCF*) da IEEE 802.11, fornecendo serviços de *QoS* priorizado e parametrizado totalmente compatíveis.

Estes mecanismos do IEEE 802.11e têm um método de acesso baseado em contenção chamado *Enhanced Distributed Channel Access (EDCA)* que opera conjuntamente com um método baseado em interrogação (*Polling*), chamado *HCF - Controlled Channel Access (HCCA)*, destinado para comunicação síncrona periódica [Mangold et al. 2003].

A coordenação entre os dois métodos de acesso é feito pelo coordenador híbrido (*Hybrid Coordination - HC*), que geralmente está localizado no ponto de acesso (*AP*) da rede [Gao et al. 2005]. Através desta emenda, pretende-se garantir a qualidade de serviço em redes sem fio para aplicações de tempo real.

O mecanismo *HCCA* tem um bom desempenho para tratar com pacotes multimídia, porém possui algumas deficiências, dentre elas podem-se citar as seguintes:

- Quando uma estação ao ser interrogada, não tiver pacotes a transmitir, obrigatoriamente a mesma enviará um pacote nulo (*QoS Null*), desperdiçando largura de banda da rede.
- O *overhead* da rede está vinculado ao tamanho do pacote de dados e ao número de pacotes transmitidos em um fluxo negociado com o HC. Isto quer dizer que pacotes

pequenos transmitidos em fluxo único, que geralmente são utilizados em automação industrial, têm um grande impacto no desempenho da rede.

- Quando o mecanismo *HCCA* opera em canal ruidoso, que é a realidade para redes sem fio, o sistema de retransmissão padrão que utiliza o quadro de *ACK* pode ter um desempenho fraco (não cumprimento de *deadlines* das mensagens para sistemas de tempo real, por exemplo) dependendo da natureza do ruído.

Devido a estes problemas de desempenho do *HCCA*, principalmente quando se transmite pacotes pequenos com rígido *deadline*, propomos um mecanismo de comunicação em tempo real chamado *GSC* (*Group Sequential Communication*) como uma alternativa ao *HCCA*. Este mecanismo de comunicação permite a coexistência de dispositivos IEEE 802.11 [IEEE 1999a] e IEEE 802.11e [IEEE 2005] aos dispositivos de *GSC*, priorizando o tráfego de tempo real (síncrono) das estações *GSC* em relação à comunicação esporádica (assíncrona) das estações genéricas que compartilham o mesmo meio de comunicação com melhor desempenho para aplicações típicas de redes industriais.

## 1.2 Motivação

A crescente demanda por aplicações populares de tempo real como sistemas multimídia (Voz e Vídeo), estimularam o desenvolvimento da emenda IEEE 802.11e que agregou qualidade de serviço ao Padrão IEEE 802.11.

Este novo recurso abriu também uma nova perspectiva para aplicações que utilizam a tecnologia IEEE 802.11 para uso em automação industrial, porém as atuais técnicas de controle de acesso ao meio resolvem em parte os problemas do não determinismo, mas o desempenho é baixo principalmente para aplicações que transmitem pacotes pequenos e que possam sofrer erros de transmissão oriundos de ruídos no canal de comunicação, apesar dos esforços em pesquisa na camada física para garantir a confiabilidade nas transmissões de dados na rede.

Portanto, para assegurar uma boa confiabilidade que satisfaça aos requisitos de tempo real, necessita-se de uma camada de acesso ao meio capaz de se comportar de forma determinística e de identificar e tratar os erros oriundos da camada física dentro de limites temporais (*deadline*) necessários para implementar comunicação compatível de tempo real aos ambientes industriais.

Atualmente os trabalhos desenvolvidos na área de comunicações de tempo real para redes sem fio são focados em tráfego de sistemas multimídia, que têm requisitos que di-

ferem da automação industrial. Estes fatores motivam o desenvolvimento e a implementação de um mecanismo de comunicação de tempo real compatível com a emenda IEEE 802.11e voltado para aplicação em redes industriais operando em modo infra-estruturado, utilizando mecanismo de controle de acesso ao meio alternativo ao *HCCA* com suporte a ruídos no canal de comunicação.

### 1.3 Objetivo Geral

Propor um novo mecanismo de controle de acesso ao meio para comunicação de tempo real compatível com o padrão IEEE 802.11/11e, que atenda adequadamente às restrições das aplicações industriais: baixo *overhead* para transmissão de pacotes pequenos e mecanismo de retransmissão de pacotes perdidos devido a ruídos no canal. Esse mecanismo foi denominado *GSC (Group Sequential Communication)*, em virtude da forma como o mecanismo trabalha.

### 1.4 Objetivos Específicos

Os objetivos específicos deste trabalho de doutorado foram delineados de modo a atingir o objetivo geral, que consiste em cumprir tarefas relevantes os quais podemos citar as seguintes:

- Estudo detalhado dos atuais mecanismos de controle de acesso ao meio em redes sem fio que abrangem o Padrão IEEE 802.11/11e.
- Pesquisa de novas estratégias de controle de acesso ao meio com suporte a tempo real utilizadas para redes industriais sem fio, incorporando o estado da arte a este trabalho de pesquisa.
- Especificação de um novo mecanismo com melhor desempenho para controle de acesso ao meio compatível ao Padrão IEEE 802.11/802.11e, onde se possa coexistir na mesma rede estações de tempo real e estações não tempo real (genéricas) em ambientes industriais.
- Implementação computacional de uma ferramenta de simulação de redes para os modelos *GSC* e *HCCA*.
- Análise do desempenho do mecanismo proposto com relação ao *HCCA*, utilizando-se para isso o recurso de simulação computacional.
- Análise da viabilidade de implementação do mecanismo proposto.



## 1.5 Contribuições da Tese

A principal contribuição desta tese está baseada na especificação e análise de desempenho de um novo mecanismo de comunicação de tempo real baseado no modelo produtor/consumidor compatível com o Padrão IEEE 802.11/11e, destinado a tratar com pequenos pacotes em redes sem fio estruturadas. Esta proposta foi denominada de *Group Sequential Communication (GSC)* e teve seu funcionamento aferido através da implementação de um simulador de redes desenvolvido em C/C++, em que foi possível comparar o desempenho do mecanismo proposto versus o *HCCA*. As simulações mostraram a eficácia do mecanismo GSC, principalmente em cenários de aplicações em automação industrial.

## 1.6 Organização do Trabalho

O restante da tese está estruturado em mais cinco capítulos, os quais são descritos da seguinte maneira:

- Capítulo 2 apresenta uma revisão bibliográfica nos fundamentos do padrão IEEE 802.11 englobando em detalhes a camada física e de enlace, principalmente a subcamada de controle de acesso ao meio, onde são descritas as principais mudanças propostas pela emenda IEEE 802.11e para agregar qualidade de serviço a rede.
- Capítulo 3 faz uma investigação minuciosa sobre o estado da arte na área do controle de acesso ao meio (MAC) compatível com o Padrão IEEE 802.11/11e.
- Capítulo 4 apresenta o mecanismo *Group Sequential Communication (GSC)* para o controle de acesso ao meio em redes sem fio compatíveis com o padrão IEEE 802.11/11e, com seu respectivo mecanismo de retransmissão denominado de algoritmo de segunda chance. O *GSC* se caracteriza como a contribuição central desta tese e foi concebido para melhorar o desempenho da comunicação de tempo real que usam pacotes pequenos, como é o caso de aplicações industriais.
- Capítulo 5 apresenta a análise de desempenho entre os mecanismos *GSC* e *HCCA* em dois cenários típicos de aplicações industriais. Os índices de desempenho analisados foram a latência, o tempo máximo de *CFP* e o percentual de perdas de pacote. Os resultados foram obtidos via simulação computacional, cujos os modelos *GSC* e *HCCA* foram implementados em linguagem C/C++ e baseados nas implementações para o *Network Simulator 2 (NS2)* em [Cicconetti et al. 2005b] e [Demarch 2007].
- Capítulo 6 apresenta as conclusões e os trabalhos futuros.



---

# Capítulo 2

## Redes Padrão IEEE 802.11/11e

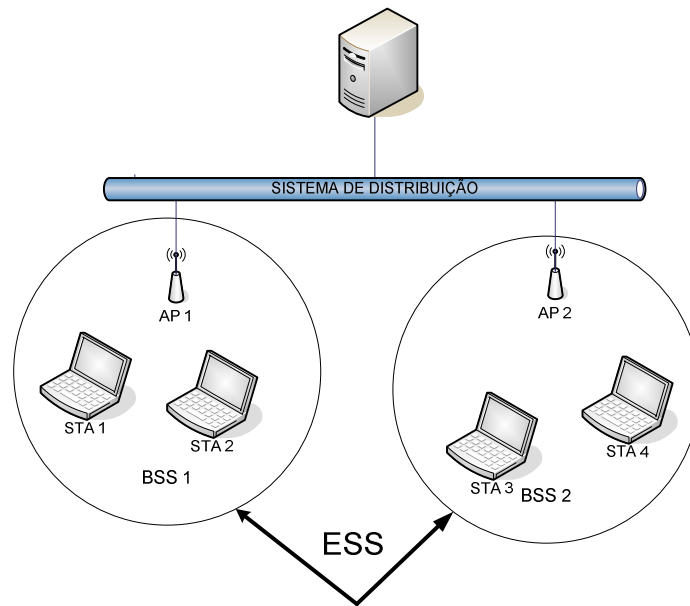
---

Este capítulo se destina a apresentar o Padrão 802.11/11e e está subdividido em duas partes: na primeira parte é apresentado o Padrão IEEE 802.11 [IEEE 1999a] [IEEE 2007], caracterizando arquiteturas, camadas físicas (*PHY layer*) e subcamada de controle de acesso ao meio (*MAC*), enquanto na segunda parte são descritas as principais mudanças propostas pela emenda IEEE 802.11e [IEEE 2005] para agregar qualidade de serviço.

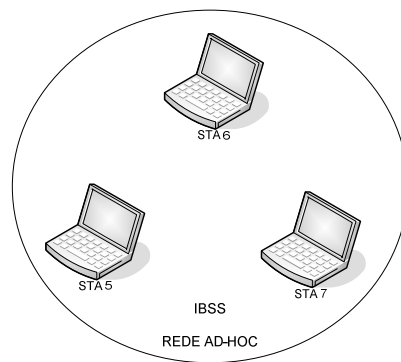
### 2.1 Arquiteturas

Segundo o Padrão IEEE 802.11 [IEEE 1999a] que trata das camadas de enlace (subcamada de Controle de Acesso ao Meio - *Medium Access Control - MAC*) e Física (*PHY*), uma rede sem fio é um grupo de duas ou mais estações (*Station - STA*) sob o domínio de uma mesma função de coordenação (*Coordination Function - CF*) formando um conjunto básico de serviço (*Basic Service Set - BSS*). A área de cobertura de uma *BSS* é designada pela área básica de serviço (*Basic Service Area - BSA*) limitada pelo alcance de propagação do seu sinal no meio de comunicação.

As redes Padrão IEEE 802.11 podem operar em dois modos distintos, chamados de *ad-hoc* e infra-estruturado. No modo *ad-hoc* há uma área básica de serviço (*BSA*), em que as estações se comunicam exclusivamente umas com as outras sem a necessidade de uma infra-estrutura formando um conjunto básico de serviço independente (*Independent BSS - IBSS*). Já o modo de operação infra-estruturado, chamado de conjunto básico de serviço (*BSS*), requer a presença de uma estação de coordenação no *BSS*, denominada de ponto de acesso ou *Access Point (AP)*, que serve de ponte entre um *BSS* e o sistema de distribuição (*Distribution System - DS*), possibilitando a comunicação entre as estações de distintos *BSS's* a outras redes. Os modos *ad-hoc* e infra-estruturado são ilustrados nas Figuras 2.1 e 2.2, respectivamente.



**Figura 2.1:** Rede Modo Infra-estruturado.



**Figura 2.2:** Rede Modo *Ad-hoc*.

O sistema de distribuição (*DS*) permite interligar vários *BSSs* formando um conjunto estendido de serviço (*Extended Service Set - ESS*), que se apresenta como um único *BSS* ampliado para a subcamada lógica (*Logical Link Control - LLC*) das estações que fazem parte do *ESS*. Geralmente utiliza-se uma rede local Ethernet como *DS* [Tanenbaum 2003], embora qualquer rede possa ser utilizada, pois o padrão IEEE 802.11 não detalha a implementação do sistema de distribuição [Farooq & Rauf 2006].

A principal função do *AP* (utilizado somente no modo infra-estruturado) é a integração dos serviços existentes na rede cabeada à rede sem fio [Gast 2002]. Existem várias

características importantes a se destacar em uma rede sem fio, dentre elas podemos citar as seguintes:

- a) **Privacidade** - Nas redes cabeadas somente as estações fisicamente conectadas ao meio de transmissão (cabos) estão aptas a receber os pacotes de dados, mas nas redes sem fio qualquer estação que seja compatível como o Padrão IEEE 802.11 pode monitorar o tráfego sem que seja descoberta, bastando para isso estar dentro do limite de alcance da rede [Gast 2002]. Para aumentar o nível de segurança da rede IEEE 802.11 foi especificado o protocolo WEP (*Wired Equivalent Privacy*) pelo IEEE 802.11i [IEEE 2004].
- b) **Autenticação** - Como as redes sem fio não podem oferecer o mesmo nível de segurança física das redes cabeadas, deve-se depender de rotinas de autenticação adicionais para assegurar que os usuários que acessem a rede estejam autorizados a fazê-lo. A autenticação é um pré-requisito necessário para a associação porque somente os usuários autenticados são autorizados a usar a rede. O padrão IEEE 802.11 suporta vários mecanismos de autenticação, inclusive o de chave compartilhada (*shared key*), que utiliza o protocolo de segurança WEP [IEEE 2004] [Gast 2002].
- c) **Associação** - Define que uma estação só poderá transmitir seus dados se a mesma estiver associada a um ponto de acesso (*AP*). Desta forma será possível ao sistema de distribuição identificar a qual *AP* pertence a estação. Uma estação não pode se associar a dois ou mais *AP* de forma simultânea, sendo que o processo de associação sempre é iniciado pela estação após o envio de uma mensagem de associação [Gast 2002].
- d) **Gestão de Consumo de Energia** - Proporciona supervisão do consumo de energia das estações. O *AP* guarda os dados destinados às estações que estão operando em modo de conservação de energia e transmite em *broadcast* as informações nos quadros de *Beacon* para identificar estas estações. As estações que operam no modo de conservação de energia “acordam” periodicamente para escutar o quadro de *Beacon*. Quando uma estação recebe indicação de que o ponto de acesso tem quadros armazenados para ela, a referida estação requisita o envio dos quadros ao *AP* [Gast 2002].
- e) **Roaming** - É descrito como um processo em que uma estação sem fio desloca-se de uma área de cobertura (*BSA*) para outra, sem perder a conexão. Porém aplicações que necessitam de requisitos de tempo real (ex: Sistema multimídia) podem ser penalizados com o não cumprimento dos máximos *delays* na rede [Gast 2002].

## 2.2 Camada Física do Padrão IEEE 802.11

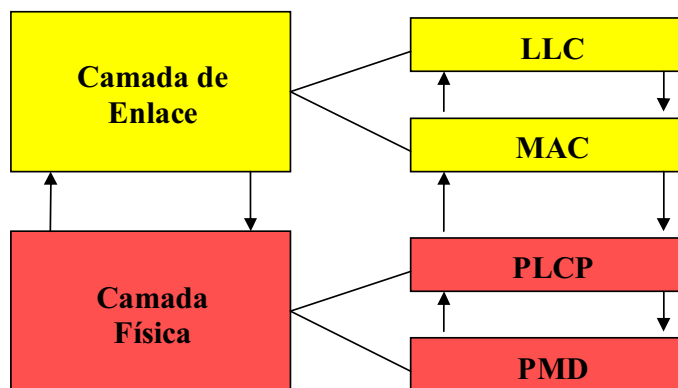
O padrão IEEE 802.11 [IEEE 1999a] [IEEE 2007] define três tipos de camada física (*PHY*), duas delas operam na banda *ISM* de 2,4 GHz utilizando espalhamento espectral. A primeira utiliza saltos em frequência (*Frequency Hopping Spread Spectrum - FHSS*) e a segunda por sequência direta (*Direct Sequence Spread Spectrum - DSSS*). Um terceiro tipo de camada física foi definido para operação na banda de infravermelho (*Diffused Infrared - DFIR*). Todas estas camadas físicas suportam *throughputs* de 1 Mbit/s a 2 Mbit/s e possuem sinal de verificação de canal livre (*Clear Channel Assessment Signal - CCA*), que é usado pela subcamada *MAC* para revelar se está livre o meio de transmissão.

A subcamada de controle de acesso ao meio (*MAC*) do Padrão IEEE 802.11 opera de forma independente da camada física. A camada física é dividida em duas subcamadas: sub-camada *PLCP* (*Physical Layer Convergence Procedure*) e subcamada *PMD* (*Physical Medium Dependent*).

A subcamada *PLCP* interliga a subcamada *MAC* ao canal de comunicação através do mapeamento dos quadros *MAC* em formato adequado à transmissão dos dados na rede, utilizando as funções da subcamada *PMD*. A subcamada *PMD* determina os tipos e métodos de transmissão e recepção dos dados no canal de comunicação.

A camada física do Padrão IEEE 802.11 recebe o quadro *MAC* e adiciona no início deste o preâmbulo e o cabeçalho da subcamada *PLCP*, que somente são transmitidos à taxa mínima da rede (1 Mbit/s), independentemente da largura de banda utilizada, permitindo a coexistência de estações que suportam taxas máximas de transmissão diferentes.

Figura 4.2 fornece uma representação visual das camadas física e de enlace e como elas estão organizadas em subcamadas.



**Figura 2.3:** Camadas Física e de Enlace com Respectivas Subcamadas.

Adicionalmente ao Padrão IEEE 802.11, outras emendas<sup>2</sup> foram desenvolvidas para aumentar as taxas de transmissão, agregar qualidade de serviço ou segurança, por exemplo. Provendo assim, interoperabilidade entre equipamentos de fabricantes diferentes. A seguir apresentaremos uma descrição resumida das especificações das camadas físicas implementadas nas tecnologias IEEE 802.11.

### 2.2.1 IEEE 802.11a

O Padrão IEEE 802.11a [IEEE 1999b] especifica a operação da camada física em banda *ISM* de frequências de 5 GHz, baseada na técnica de modulação OFDM (*Orthogonal Frequency Division Multiplexing*). Possui taxas de transmissão e recepção de dados de 6, 9, 12, 18, 24, 36, 48 e 54 Mbps, sendo que as taxas de 6, 12 e 24 Mbps são obrigatórias. São utilizadas 52 subportadoras: 48 para dados e 04 para sinais piloto. As 48 subportadoras de dados podem ser moduladas utilizando BPSK (*Binary Phase-Shift Keying*), QPSK (*Quadrature Phase-Shift Keying*), 16QAM (*Quadrature Amplitude Modulation*) ou 64QAM. Além das opções de modulação apresentadas, existem diferentes opções de taxas de codificação para correção de erros (*Forward Error Correction - FEC*). As taxas de codificação possíveis são 1/2, 2/3 e 3/4, conforme ilustrado na Tabela 2.1.

**Tabela 2.1:** Taxas de Transmissão do Padrão IEEE 802.11a.

Modulação	Taxa de Codificação FEC	Taxa de Transmissão
BPSK	1/2	6 Mbit/s
BPSK	3/4	9 Mbit/s
QPSK	1/2	12 Mbit/s
QPSK	3/4	18 Mbit/s
16QAM	1/2	24 Mbit/s
16QAM	3/4	36 Mbit/s
64QAM	2/3	48 Mbit/s
64QAM	3/4	54 Mbit/s

<sup>2</sup>Os termos padrão, emenda ou adendo são utilizados para descrever variações do Padrão IEEE 802.11.

### 2.2.2 IEEE 802.11b

O Padrão IEEE 802.11b [IEEE 1999c] estende a especificação da camada física original do IEEE 802.11 baseada em Espalhamento Espectral por Sequência Direta (*DSSS*) para o *High Rate Direct Sequence Spread Spectrum (HR/DSSS)* e aumentou o *throughput* de 1 e 2 Mbps para 5,5 e 11 Mbps. O aumento do *throughput* é obtido pela utilização da modulação *CCK (Complementary Code Keying)*. As redes IEEE 802.11b usam para operação 14 canais na banda *ISM* de 2,4 GHz, cada qual ocupando 22 MHz. Os canais adjacentes sobrepõem-se parcialmente (Tabela 2.2), mas na prática apenas 03 canais podem ser usados simultaneamente. Para reduzir o *overhead* associado à camada física, o Padrão IEEE 802.11b define um formato curto (opcional) para o preâmbulo e o cabeçalho da *PLCP*, em alternativa ao formato padrão. Somente podendo ser utilizado entre estações compatíveis.

**Tabela 2.2:** Canal e Frequência para o Padrão IEEE 802.11b/g.

Canal	Frequência Central	Canais Sobrepostos
1	2.412 GHz	2,3,4,5
2	2.417 GHz	1,3,4,5,6
3	2.422 GHz	1,2,4,5,6,7
4	2.427 GHz	1,2,3,5,6,7,8
5	2.432 GHz	1,2,3,4,6,7,8,9
6	2.437 GHz	2,3,4,5,7,8,9,10
7	2.442 GHz	3,4,5,6,8,9,10,11
8	2.447 GHz	4,5,6,7,9,10,11,12
9	2.452 GHz	5,6,7,8,10,11,12,13
10	2.457 GHz	6,7,8,9,11,12,13,14
11	2.462 GHz	7,8,9,10,12,13,14
12	2.467 GHz	8,9,10,11,13,14
13	2.472 GHz	9,10,11,12,14
14	2.484 GHz	10,11,12,13



### 2.2.3 IEEE 802.11g

O Padrão IEEE 802.11g [IEEE 2003] é uma extensão da especificação IEEE 802.11b. Trabalha com modulação OFDM na banda *ISM* de 2,4 GHz e suporta o *throughput* máximo de 54Mbit/s, similar ao Padrão IEEE 802.11a. As estações IEEE 802.11g devem suportar todos os modos de transmissão do Padrão IEEE 802.11b para haver compatibilidade, o que torna necessário a implementação da modulação *CCK*. O desafio neste caso é a detecção das transmissões que utilizam a modulação *OFDM* por estações compatíveis IEEE 802.11b para evitar colisões. Uma solução consiste em transmitir antes dos dados os quadros de *RTS (Request to Send)* e *CTS (Clear to Send)*, usando modulação *CCK*, que é reconhecida pelo IEEE 802.11b. Opcionalmente, pode-se utilizar um esquema de modulação híbrida, que trabalha simultaneamente com as modulações *CCK* e *OFDM*. A modulação *CCK* é usada para transmitir o preâmbulo e cabeçalho dos dados, enquanto a modulação *OFDM* é usada para a transmitir dados. Devido a isso, as estações compatíveis IEEE 802.11b são avisadas do início e duração de uma transmissão. Depois, os dados são transmitidos com máximo *throughput* suportado pela modulação *OFDM*. O custo da compatibilidade proporcionada pelas soluções apresentadas é a queda do desempenho devido à introdução de *overheads* quando comparado ao uso exclusivo da modulação *OFDM*.

### 2.2.4 IEEE 802.11n

O IEEE criou no ano de 2004 um novo grupo de trabalho para desenvolver a emenda 802.11n. Esta emenda utiliza uma nova estratégia de transmissão chamada MIMO (*Multiple-Input Multiple-Output*) onde há um aumento das taxas de transmissão oriunda da utilização simultânea de vários canais de comunicação. Essa emenda opera na banda *ISM* em 2,4 e 5 GHz sendo compatível com os padrões 802.11a e 802.11b, obtendo uma taxa de transmissão máxima de 320Mbps utilizando modulação *MIMO-OFDM* [Ni 2005]. Atualmente está editada em versão rascunho (draft).<sup>3</sup> A Tabela 2.3 fornece uma comparação básica entre as tecnologias a, b, g e n compatível ao padrão IEEE 802.11.

---

<sup>3</sup>Obs: quando da escrita desta tese, o IEEE divulgou a comunidade científica que o padrão 802.11n estaria disponível até o final do ano de 2009.

**Tabela 2.3:** Comparativo entre os Padrões da Família IEEE 802.11.

Padrão	Publicação	Banda de Frequência	Modulação	Taxa de Transmissão
802.11	1997	2,4 GHz	FHSS ou DSSS	2 Mbps
802.11a	1999	5 GHz	OFDM	54 Mbps
802.11b	1999	2,4 GHz	HR/DSSS	11 Mbps
802.11g	2003	2,4 GHz	DSSS-OFDM	54 Mbps
802.11n	2009	2,4/5 GHz	MIMO-OFDM	320 Mbps

O IEEE atua desenvolvendo e publicando especificações de padrão e emendas, porém não certifica os equipamentos implementados pelos fabricantes; esta responsabilidade é dada a *Wi-Fi Alliance* onde há um programa que concede o selo *Wi-Fi* aos equipamentos em conformidade com os padrões 802.11a/b/g/n para garantir a interoperabilidade dos dispositivos de diferentes fabricantes [Demarch 2007].

## 2.3 Subcamada de Controle de Acesso ao Meio (MAC)

A subcamada de controle de acesso ao meio (MAC) do IEEE 802.11 implementa dois mecanismos distintos: a Função de Coordenação Distribuída (*Distributed Coordination Function - DCF*), obrigatória; e a Função de Coordenação Pontual (*Point Coordination Function - PCF*), opcional.

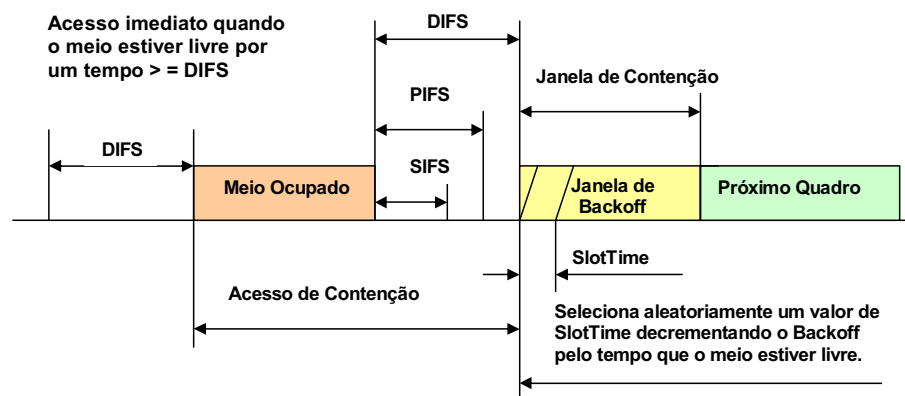
O *DCF* é o mecanismo básico de controle de acesso ao meio para IEEE 802.11 e utiliza um protocolo de acesso aleatório do tipo *CSMA/CA* (*Carrier Sense Multiple Access/Collision Avoidance*), que suporta tráfego assíncrono oriundo da disputa em igualdade de condições das estações ao meio de transmissão. Já o *PCF*, por outro lado, é um protocolo de interrogação (*Polling*) que permite o acesso ao meio livre de contenção pelas estações, o que torna sua utilização mais adequada para o transporte de tráfego síncrono (tempo real, por exemplo), entretanto a sua implementação não é obrigatória no padrão, ao contrário do *DCF*.

### 2.3.1 Função de Coordenação Distribuída (DCF)

De acordo com o Padrão 802.11 [IEEE 1999a] [IEEE 2007], o *DCF* é um mecanismo de controle de acesso ao meio baseado no *CSMA/CA*. Logo esse mecanismo está sujeito a

colisões, que diminuem o seu desempenho. Ele trabalha da seguinte maneira: no instante inicial, antes de começar a transmissão de um quadro de dados, a estação transmissora deve verificar se o meio está livre, e se o meio continuar livre após um período *DIFS* e seu contador de *backoff* for igual a zero, a estação estará apta para transmitir o seu quadro. Caso contrário, a estação adia momentaneamente a sua transmissão e entra no processo de *backoff* [Bianchi et al. 2005].

Assim, a distância entre quadros (*Interframe Spaces - IFS*) é usada como uma marcação temporal mostrando as possibilidades de inatividade do canal. Os espaçamentos temporais entre quadros são pré-definidos e podem assumir valores distintos, tais como: *SIFS*, *PIFS* e *DIFS*, que são os mais comuns e estão ilustrados na Figura 2.4, onde:



**Figura 2.4:** Espaçamento Entre Quadros do Padrão IEEE 802.11.

- Espaçamento curto entre quadros (*Short Inter Frame Spacing - SIFS*): é usado para separar os quadros de *RTS*, *CTS*, *DATA* e *ACK* em um processo de transmissão. Se a estação receptora não recebe o quadro após *SIFS*, geralmente significa que algum erro ocorreu ao quadro transmitido [Chittamuru et al. 2005].
- Espaçamento *PCF* entre quadros (*PCF Inter Frame Spacing - PIFS*): o *AP* utiliza este espaçamento para tomar o controle do canal sem interferências das estações. [Chittamuru et al. 2005].
- Espaçamento *DCF* (*DCF Inter Frame Spacing - DIFS*) entre quadros: utilizados por estações durante o período de contenção. Depois de qualquer transmissão de dados, se uma estação quiser acessar o canal, ela deve esperar por um período de inatividade do canal igual a um período de *DIFS* e iniciar a sua transmissão ou retomar o seu processo de *backoff* [Chittamuru et al. 2005].

No processo de *backoff*, a estação gera um número inteiro randômico, uniformemente distribuído entre o valor zero até o valor mínimo da janela de contenção ( $CW_{min}$ ), que é usado para iniciar o contador de *backoff* chamado de janela de contenção *Contention Window - CW* [Li et al. 2005]. Quando o meio fica livre novamente e permanece livre durante um período *DIFS*, a estação começa a decrementar o seu contador de *backoff* em intervalos fixos de tempo de (*aSlotTime*), até o contador chegar a zero. Se a estação detecta uma nova transmissão durante esse período, há a parada no decremento do contador de *backoff*, aguardando que o meio volte a ficar livre durante um novo período *DIFS* para recomençar o decremento do contador de *backoff* a partir do ponto em que houve a parada. Quando o contador de *backoff* alcança o valor zero, a estação inicia a sua nova tentativa de transmissão de dados. Caso duas ou mais estações iniciem suas transmissões simultaneamente ocorre o fenômeno chamado colisão [Xiao 2004].

No processo de colisão onde duas ou mais estações têm suas transmissões iniciadas simultaneamente não há recebimento do quadro de reconhecimento *Acknowledgment - ACK* em um período definido por *ACK-timeout*. Há também outra possibilidade da estação receptora não enviar um quadro de *ACK*: quando esta recebe um quadro de dados corrompido. Assim, do lado do emissor não há diferença entre este caso e o da ocorrência de uma colisão. Se o emissor não receber um quadro de *ACK* em resposta ao envio de um quadro de dados, ele assume que o quadro não foi entregue. Esgotado o período de *ACK-timeout* para a recepção do quadro *ACK*, o emissor repete novamente o processo de *backoff*, antes de retransmitir o quadro de dados. Entretanto, para minimizar a probabilidade de colisão, após cada tentativa sem êxito de transmissão de um quadro, as estações transmissoras dobram os seus valores de  $CW$  ( $CW = 2(CW + 1) - 1$ ) até um valor máximo de  $CW_{max}$ . Caso ocorra uma transmissão com sucesso (recepção do quadro de *ACK*), o valor de  $CW$  volta a ser igual à  $CW_{min}$  [Ni 2005]. Caso a estação tenha novos dados a transmitir, ela deve executar novamente o processo de *backoff* antes de iniciar a nova transmissão. Este exemplo é ilustrado na Figura 2.5.

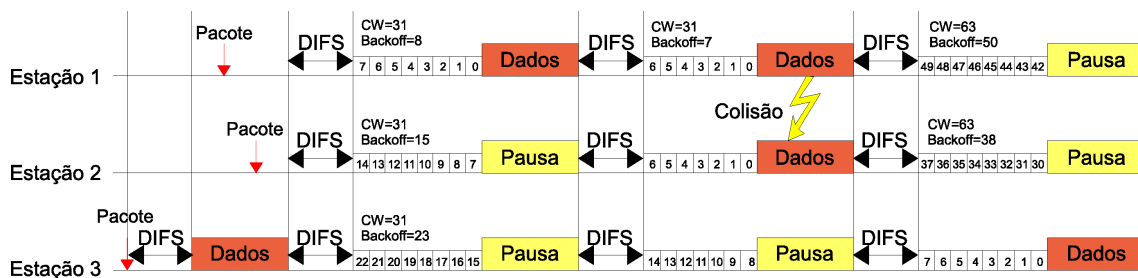


Figura 2.5: Exemplo de Operação DCF.

De acordo com o Padrão IEEE 802.11 [IEEE 1999a], como o período de tempo *SIFS* é menor do que o *DIFS*, a estação receptora pode transmitir o quadro *ACK* sem verificar o meio anteriormente e sem risco de colisão de quadros, já que as outras estações estão inibidas de iniciar as suas transmissões durante um período de tempo maior. O destinatário de um quadro de dados, que foi recebido corretamente responde à transmissão com um quadro de *ACK*, depois de esperar por um período de tempo igual a *SIFS*, havendo a seguinte relação:

- $PIFS = SIFS + aSlotTime$
- $DIFS = SIFS + 2 * aSlotTime$ .

A probabilidade de recepção de um quadro corrompido nas redes sem fio é maior do que em redes cabeadas, já que num meio sem fio não é possível a detecção de colisões devido à utilização do *CSMA/CA* que é incapaz de transmitir e escutar o meio simultaneamente, além do que as transmissões estão muito mais expostas a ruídos que degradam o desempenho do canal de comunicação.

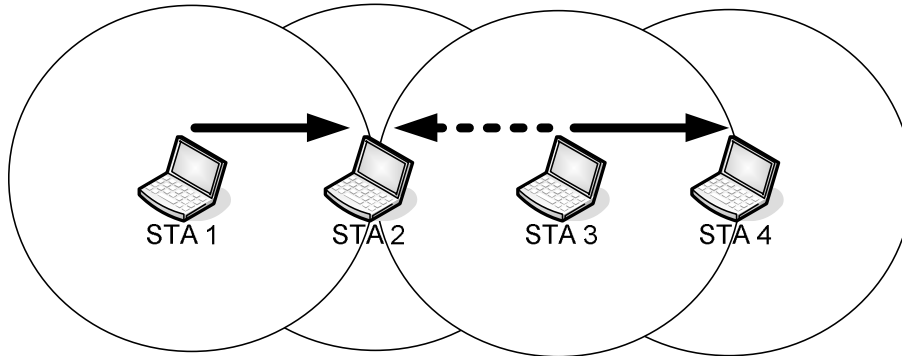
### O Problema da Estação Oculta e Exposta

O problema mais comum que ocorre com o protocolo *CSMA/CA* é o fenômeno da estação oculta e exposta, pois em uma rede sem fio as estações são sensíveis às suas respectivas localizações para acessar o meio e transmitir seus quadros de dados. Dependendo de como estão distribuídas na rede podem haver problemas na comunicação.

Considere uma rede sem fio composta por quatro Estações: STA1, STA2, STA3 e STA4, como mostrado na Figura 2.6. Vamos primeiro considerar o seguinte cenário: ambas as estações STA1 e STA3 têm quadros de dados para enviar as estações STA2 e STA4, respectivamente. Suponha que as estações STA1 e STA3 estão fora do alcance de comunicação uma da outra. Agora, suponha que estação STA1 começa a transmitir primeiro. Neste momento, a estação STA3 verifica o meio, mas não poderá escutar o pacote de STA1, pois está fora do alcance de STA1. Então, STA3 acha que o meio está livre e decide começar também a sua transmissão do pacote a estação STA4. Infelizmente, os quadros de dados das estações STA1 e STA3, colidem próximos a estação STA2, logo a estação STA2 não pode receber o quadro de dados com êxito.

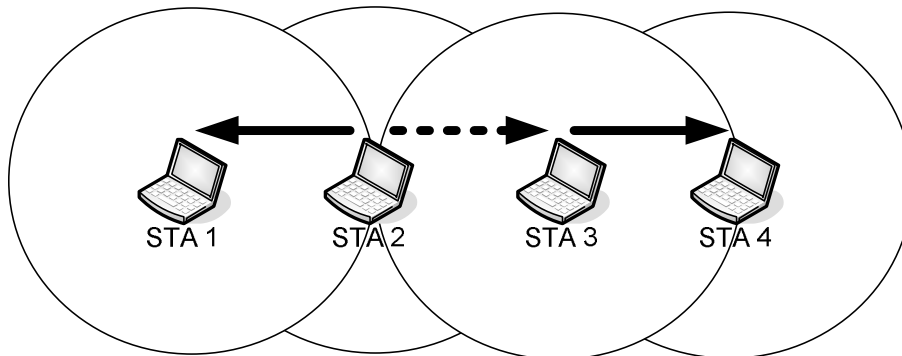
Este problema no processo de detecção da portadora em redes sem fio é chamado fenômeno da estação oculta ou *hidden station* [Ni 2005] [Willig et al. 2005], o que significa que este fenômeno acontece quando uma estação é capaz de detectar o sinal de

transmissão de duas estações próximas, mas as referidas estações não conseguem detectar o sinal uma da outra, seja por estarem muito distantes ou por terem o sinal atenuado por obstáculos [Ni 2005]. A Figura 2.6 ilustra o fenômeno da estação oculta.



**Figura 2.6:** Problema da Estação Oculta.

Agora, vamos considerar a mesma rede, mas inverter o sentido da transmissão de STA1 para STA2, ou seja, as estações STA2 e STA3 têm um quadro de dados a ser transmitidos para STA1 e STA4, respectivamente. Quando a estação STA2 começa a transmitir, a estação STA3 verifica o canal e encontra o meio ocupado. Assim a estação STA3 decide adiar a transmissão, embora o meio em torno da estação STA4 esteja livre e os quadros de dados possam ser recebidos com êxito. Este adiamento da transmissão é desnecessário e introduz atrasos adicionais. Esta situação é chamado de problema da estação exposta, o que significa que um emissor (estação STA3) é exposta ao alcance de um outro emissor (estação STA2) [Mangold et al. 2003]. A Figura 2.7 ilustra o fenômeno da estação exposta.



**Figura 2.7:** Problema da Estação Exposta.

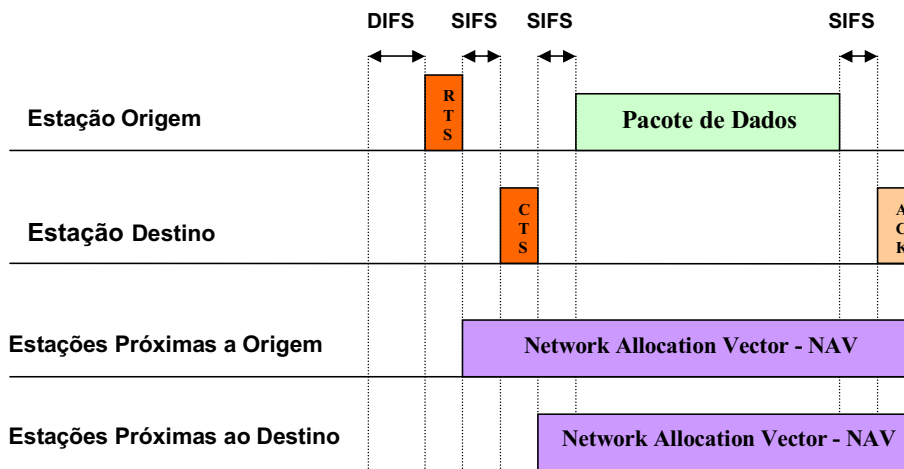
Baseado nos exemplos apresentados, pode-se observar que uma estação transmissora pode verificar o meio somente dentro do raio de alcance de seu sinal. No entanto, a recepção bem sucedida depende do estado em que se encontra o meio no lado do receptor, também. O estado que se encontra o meio nos dois lados podem ser diferentes e, portanto, a estimativa do estado do meio (que pode estar livre ou ocupado no lado do receptor) pelo emissor é uma questão complexa.

### **Mecanismo de Controle de Transmissão (RTS/CTS)**

Para tratar o fenômeno da estação oculta e exposta, o protocolo *MAC* do IEEE 802.11 inclui um mecanismo opcional, baseado na troca de dois quadros de controle antes do envio do quadro de dados, chamados de quadro *RTS* (*Request To Send*) e *CTS* (*Clear To Send*) que são transmitidos em *broadcast*. O *RTS* é enviado pela estação transmissora do quadro de dados (estação origem). A estação origem recebe em resposta à solicitação, o quadro *CTS*, enviado pela estação receptora (estação destino), após um período *SIFS*. Se o quadro *CTS* não chegar com sucesso, a estação origem inicia o processo de *backoff* para retransmitir o quadro *RTS*. De outra forma, se a troca dos quadros de *RTS* e *CTS* tiver sucesso, a estação origem pode transmitir o quadro de dados, após um período *SIFS* de inatividade do meio. O quadro *RTS* possui um campo chamado *Duration* que indica o tempo previsto para a transmissão do quadro *CTS*, do quadro de dados (*DATA*) e do quadro *ACK*, incluindo os períodos *SIFS* entre os quadros. Esse tempo é calculado pela estação origem levando em consideração o tamanho dos quadros e a velocidade de transmissão utilizada.

O campo *Duration* (tempo previsto para a transmissão) também está presente no quadro *CTS*, sendo atribuído pela estação receptora com base no valor definido no quadro *RTS*, descontando-se o tempo necessário para a transmissão do quadro *CTS*. Todas as estações que fazem parte da rede analisam o conteúdo dos quadros *RTS* e *CTS*, independentemente do destinatário, e utilizam a informação do campo *duration* para atualizar os seus respectivos temporizadores (*Network Allocation Vector* - *NAV*). Este temporizador bloqueia a transmissão da estação até o tempo programado expirar, mesmo que o meio esteja inativo. Este mecanismo é chamado de detecção de portadora virtual e é ilustrado na Figura 2.8





**Figura 2.8:** Comunicação entre Estações Utilizando o Mecanismo *RTS/CTS*.

Como podemos ver na Figura 2.8, a transmissão do quadro *RTS* inibe a transmissão das estações próximas da estação origem, enquanto que a transmissão do quadro *CTS* inibe a transmissão das estações próximas da estação destino. Desta forma, a estação origem está apta para a transmissão do quadro de dados sem o risco de colisão.

Com o uso do mecanismo de *RTS/CTS* há probabilidade de haver colisões entre quadros *RTS* e é equivalente a ocorrência de colisões entre quadros de dados sem a utilização do mecanismo. Entretanto, como os quadros *RTS* são geralmente menores do que os quadros de dados, o tempo gasto numa colisão é menor, resultando em um aumento da eficiência do protocolo *MAC*, mesmo não considerando o fenômeno da estação oculta. A desvantagem principal da utilização do mecanismo de *RTS/CTS* está no aumento do *overhead* da rede, principalmente na transmissão de quadros de dados pequenos, como é o caso de aplicações em redes industriais. A opção do parâmetro chamado *RTS Threshold* permite justamente definir a partir de que tamanho do quadro de dados o mecanismo *RTS/CTS* pode ser utilizado na rede.

### Estrutura do Quadro do IEEE 802.11/11e

O Padrão IEEE 802.11/11e [IEEE 1999a] [IEEE 2005] especifica três tipos de quadros:

- Quadros de Dados (*Data Frames*) são utilizados para transmissão de dados;
- Quadros de Controle (*Control Frames*) são utilizados para o controle do acesso ao meio, tais como: *RTS*, *CTS* e *ACK*;



- Quadros de Gerenciamento (*Management Frames*) são utilizados para troca de informação de supervisão e são transmitidos do mesmo modo que os quadros de dados, apenas não são repassados às camadas dos níveis superiores.

Cada um desses tipos é subdividido em subtipos diferentes, de acordo com as suas características.

Figura 2.9 ilustra os campos que compõem o quadro básico do IEEE 802.11/11e, onde:



**Figura 2.9:** Formato do Quadro Padrão IEEE 802.11/11e.

- O preâmbulo é um campo utilizado pela camada física e é formado por duas partes: o campo de sincronismo e o campo de início de quadro (*Start Frame Delimiter - SFD*). O campo de sincronismo é uma sequência alternada de valores “0” e “1”s que é usada pelo *hardware* que implementa a camada física para sincronizar as estações.
- O campo PLCP contém informações lógicas que são utilizadas pela camada física para decodificar o quadro e é formado por três partes: *PLCP-PDU Length Word*, *PLCP Signaling Field* e *Header Error Check Field*. O subcampo *PLCP-PDU Length Word* informa o número de bytes que o pacote possui. Esta informação será usada pela camada física para detectar corretamente o final do quadro. O subcampo *PLCP Signaling Field* reconhece a taxa de transmissão utilizada. finalmente, o subcampo *Header Error Check Field* é um campo de 16 bits que contém um CRC (*Cyclic Redundancy Code*) usado para detecção de erro no cabeçalho PLCP.

**Quadro de Dados MAC** - O formato geral do quadro de dados *MAC* no nível de enlace do Padrão IEEE 802.11/11e dispõe de um conjunto de campos que tem uma ordem fixa em todos os quadros. A Figura 2.10 ilustra o formato do quadro geral do *MAC*. Os três primeiros campos (Controle de Quadro, Duração/ID e Endereço 1) e o último campo (FCS) constituem o formato do quadro mínimo e estão presentes em todos os quadros, incluindo tipos e subtipos reservados. Os campos de Endereço 2, Endereço 3, Controle de Sequência, Endereço 4, Controle de *QoS* (somente para quadros 802.11e) e área de dados (Frame Body) estão presentes apenas em certos tipos e subtipos de quadros. O campo área de dados é de tamanho variável

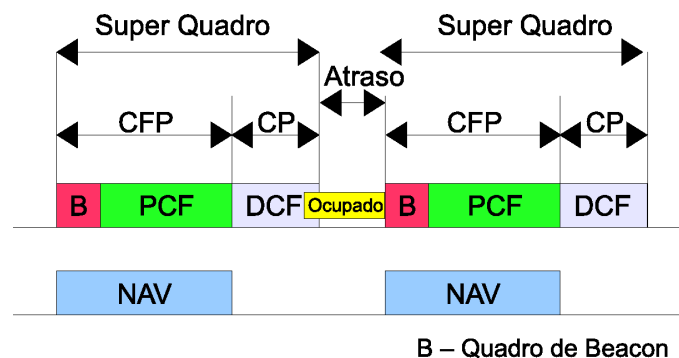


- O campo controle de *QoS* (*QoS Control*) é um campo de comprimento fixo de 2 bytes que identifica o *Traffic Category - TC* ou *Traffic Stream - TS* a qual o quadro pertence e várias outras informações relacionadas ao *QoS* que varia conforme o tipo e subtipo. O campo de controle de *QoS* está presente em todos os quadros de dados do IEEE 802.11e;
- O campo área de dados (*Body Frame*) é um campo de comprimento variável que transporta informação relacionada com o tipo e subtipo do quadro. No caso específico do quadro de dados, este campo transporta os dados úteis ou *payload*, limitado a 2312 bytes para IEEE 802.11 e 2304 para IEEE 802.11e;
- O campo FCS (*Frame Check Sequence*) carrega um código CRC de 04 bytes usado para a detecção de erros no quadro. O FCS é calculado sobre todos os campos do cabeçalho e da área de dados.

### 2.3.2 Função de Coordenação Pontual (*PCF*)

O padrão IEEE 802.11 [IEEE 1999a] [IEEE 2007] incluiu a Função de Coordenação Pontual (*PCF*) com o objetivo de proporcionar suporte a serviços síncronos (por exemplo, tempo real). Este *MAC* utiliza um protocolo de *Polling* controlado pelo Coordenador Pontual (*Point Coordinator - PC*) geralmente embutido no *AP*, que controla as estações que devem transmitir seus dados na rede.

A função *PCF* é usada para o controle de acesso ao meio durante o período livre de contenção (*Contention Free Period - CFP*), que se alterna com o período de contenção (*Contention Period - CP*) em um superquadro. No período de contenção, usa-se a função *DCF* para transmissão assíncrona, permitindo-se também a inscrição de novas estações na lista de *Polling* utilizada pelo escalonador. O superquadro é ilustrado na Figura 2.11.



**Figura 2.11:** Coexistência da Função *PCF* e *DCF* no Superquadro.

O *PCF* trabalha da seguinte forma: em um novo *CFP*, o coordenador pontual (*PC*) aguarda que o canal de comunicação permaneça livre durante um período *PIFS* para tomar o controle do meio. O *CFP* inicia com a transmissão do quadro de *Beacon* pelo *PC*. Logo em seguida, é feita a interrogação das estações pertencentes à lista de *Polling*. A transmissão do quadro *CF-End*, pelo *PC*, encerra o *CFP*. Neste período, os quadros são espaçados de um período *SIFS* [Afonso 2004].

O *Beacon* é um quadro de gerenciamento utilizado para transmitir diversas informações relacionadas com a área básica de serviço (*BSA*). Entre as informações ligadas à função *PCF*, encontra-se o intervalo de repetição do período *CFP*, ou seja, a duração nominal do superquadro ou intervalo de serviço (*Service Interval - SI*). O *Beacon* inclui também o parâmetro *CFPMaxDuration*, que indica a duração máxima do período *CFP* atual, sendo usado pelas estações para atualizar os seus temporizadores (*NAV*). O máximo valor que este parâmetro pode assumir corresponde à duração nominal do superquadro menos o tempo necessário para a transmissão de um *MPDU* (*MAC Package Data Unit*) de tamanho máximo durante o *CP*, incluindo os quadros de controle.

Durante o *CFP*, os seguintes subtipos de quadros de dados podem ser transmitidos: *DATA*, *DATA+CF-ACK*, *DATA+CF-Poll*, *DATA+CF-ACK+CF-Poll*, *Null Function* (sem dados), *CF-ACK* (sem dados), *CF-Poll* (sem dados) e *CF-ACK+CFPoll*(sem dados) [Cloran 2004].

O *PC* tem a função de transmitir os quadros de *CF-Poll*, enquanto os outros quadros podem ser transmitidos por qualquer estação que esteja usando a função *PCF*. O envio do quadro *CF-Poll* autoriza a transmissão de dados pela estação destino. O quadro *Null Function* é utilizado pela estação receptora quando esta é interrogada e não tem dados nem informação de reconhecimento *CF-ACK* para transmitir.

A partir do momento que o *PC* recebe os dados de uma estação, este pode enviar dados (*DATA*) e interrogar (*CF-Poll*) uma outra estação simultaneamente em que reconhece (*CF-ACK*) os dados recebidos da primeira. A capacidade do *PC* de mesclar a interrogação, o reconhecimento e os dados num mesmo quadro foi concebida para aumentar a eficiência do protocolo. Quando o *PC* envia um quadro de *Polling* a uma estação e esta não inicia a sua transmissão no momento esperado (decorrido um período *SIFS*), o *PC* retoma novamente o controle do meio após um período *PIFS*, dando sequência à interrogação das outras estações presentes na lista de *polling*. Como podemos ver na Figura 2.12.

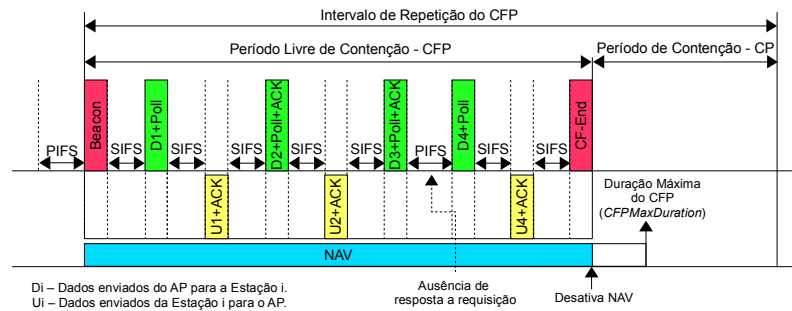


Figura 2.12: Transmissões Realizadas Durante o CFP.

Quando a estação é interrogada pelo PC, esta pode enviar um quadro de dados para outra estação. A outra estação responde com um quadro ACK da função DCF. Depois, o PC retoma o controle do meio, esperando durante um período PIFS. Da mesma forma, o PC pode transmitir um quadro de dados para outra estação que não reconhece a função PCF durante o CFP, e esta estação responde com um quadro ACK da função DCF. Se, no tempo previsto para o início do CFP, o meio estiver ocupado com uma transmissão em curso no CP (tráfego DCF), o PC tem que esperar pelo fim desta transmissão para adquirir o controle do meio. Porém, o CFP não pode durar para além do limite especificado pelo parâmetro *CFPMaxDuration*, que é o tempo de duração máxima do CP. Então, este superquadro sofre uma redução temporal. Como uma transmissão pode se iniciar imediatamente antes do término previsto para o CP, a diminuição do CFP pode atingir o tempo necessário para a transmissão de um MPDU de comprimento máximo, juntamente com os quadros de controle. A diminuição do CFP provocado pelo aumento do CP diminui a largura de banda disponível para o transporte do tráfego PCF, aumentando o jitter das conexões. Este evento é mostrado na Figura 2.13. Devido a este problema do início aleatório do CFP, não houve o interesse dos fabricantes em implementar o PCF em equipamentos reais [Cena et al. 2008].

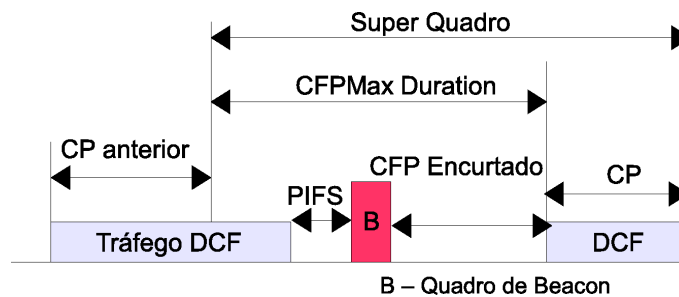


Figura 2.13: Exemplo de Diminuição do CFP.

### Restrições à Qualidade de Serviço (*QoS*)

Atualmente o suporte a serviços que necessitam de *QoS* é um dos maiores desafios para a subcamada *MAC* nas redes IEEE 802.11. Estas redes possuem características intrínsecas com razoável perda de pacotes, atraso (*delay*) e variação do atraso (*jitter*) altos. Logo, os mecanismos *DCF* e *PCF* possuem algumas limitações para prover *QoS*. Um serviço de *QoS* pode ser classificado em parametrizado e priorizado. O primeiro é uma exigência fundamental de *QoS* que se expressa em termos de valores quantitativos, como atraso, *jitter* e taxas de transmissão. O segundo é expresso em termos de entrega de pacotes com classes de prioridade sem requisitos quantitativos.

### Restrições do Mecanismo *DCF*

O mecanismo *DCF* provê somente o serviço de melhor esforço (*best effort*), onde em uma rede sobrecarregada os pacotes de dados são simplesmente descartados da fila de transmissão, quando esta estiver cheia. Além disso, no *DCF*, todas as estações competem pelo acesso ao meio simultâneo com a mesma prioridade ocasionando razoável nível de colisões de pacotes. Não existe um esquema que diferencie os tipos de tráfego para prover, por exemplo, um melhor serviço para as aplicações de voz e vídeo em tempo real ou para as aplicações de dados originando problemas para estas aplicações que exigem qualidade de serviço na rede [Ni 2005].

### Restrições do Mecanismo *PCF*

A princípio, o mecanismo *PCF* foi desenvolvido para suportar aplicações que demandassem requisitos de tempo real, mas mesmo assim possui problemas que prejudicam o serviço de *QoS*. Primeiramente, o *PCF* não é capaz de gerenciar simultaneamente vários tipos de tráfego de *QoS*, pois trabalha geralmente com o algoritmo de escalonamento *Round-Robin*, onde as transmissões são realizadas na ordem sequencial da lista de *polling*. Além disso, o *AP* ainda precisa competir com as outras estações para acessar o meio a fim de poder transmitir o quadro de *Beacon*, geralmente gerando atrasos na transmissão do mesmo devido ao tráfego das estações no *CP*. Este atraso adia a transmissão de dados durante o próximo *CFP*, prejudicando o desempenho da rede. Isto ocorre em função do *PCF* não conseguir controlar o tempo de transmissão de uma estação após o *polling*, uma vez que as estações podem enviar quadros de tamanhos variados (entre 0 e 2312 *bytes*), o que gera um tempo de transmissão diferente para cada tamanho de quadro [Afonso 2004]. Além disso, a taxa de transmissão de uma estação pode variar de acordo com as condições do canal, de maneira que o *AP* não é capaz de prever de forma precisa

o tempo de transmissão, fazendo com que o mesmo falhe em prover garantias de atraso e *jitter* para as estações presentes na lista de *polling* durante o próximo *CFP*. Pode-se citar que um dos principais problemas comuns a ambos os mecanismos é a inexistência de uma especificação de um controle de admissão para o IEEE 802.11, causando uma perda de desempenho quando há muito tráfego na rede [Ni 2005].

## 2.4 Emenda IEEE 802.11e

O grupo de trabalho **E** especificou a emenda denominada IEEE 802.11e [IEEE 2005], cujo objetivo foi aperfeiçoar o protocolo de controle de acesso ao meio pertencente ao Padrão IEEE 802.11 [IEEE 1999a] [IEEE 2007], de forma a aumentar a eficiência e o suporte de qualidade de serviço para aplicações em redes sem fio tal como a transmissão de voz e vídeo (tráfego multimídia). A norma IEEE 802.11e também define uma nova Função de Coordenação para o controle de acesso ao meio, denominado Função de Coordenação Híbrida (*Hybrid Coordination Function - HCF*), que tem esta denominação híbrida porque consegue combinar dois mecanismo de acesso ao meio, um baseado em contenção e outro baseado em não-contenção [Pavón & Shankar 2004].

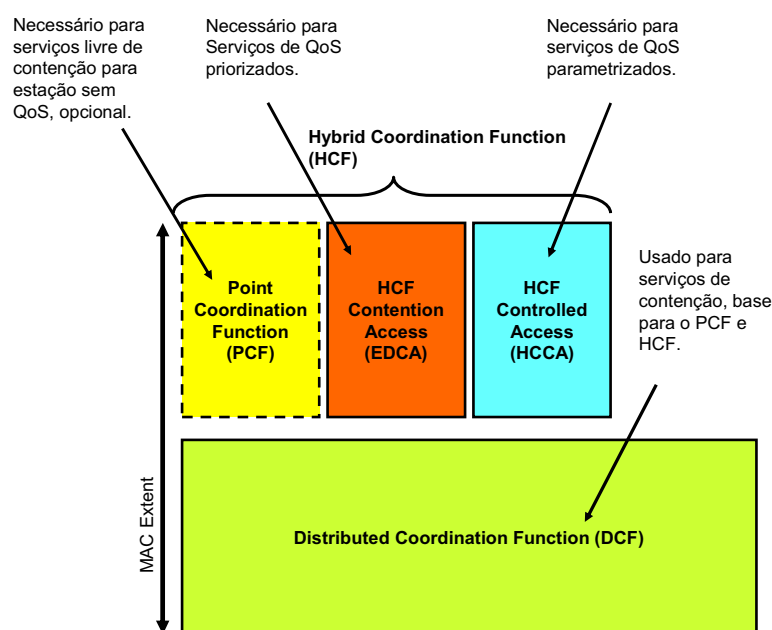
### 2.4.1 Função de Coordenação Híbrida (*HCF*)

O *HCF* utiliza-se de mecanismos de controle de acesso ao meio similar ao *DCF* e o *PCF*. Para prover a qualidade de serviço, combina acesso baseado em contenção denominado *EDCA* (*Enhanced Distributed Channel Access*), que é uma melhoria do *DCF* e o baseado em não-contenção, chamado acesso controlado ao canal, denominado *HCCA* (*HCF Controlled Channel Access*), que por sua vez é uma melhoria do *PCF*. O controle destes mecanismos é realizado pelo coordenador híbrido (*Hybrid Coordination-HC*) que geralmente está embutido no *AP*. No padrão IEEE 802.11e, o *AP* que suporta *QoS* e as estações que provêm serviços de *QoS* são denominados *QAP* (*QoS Access Point*) e *QSTA* (*QoS Station*), respectivamente, e o *BSS* em que operam é denominado *QBSS* (*QoS Basic Service Set*) [IEEE 2005].

O *HCF* cria o conceito de oportunidade de transmissão (*Transmission Opportunity - TXOP*), definido como um intervalo de tempo exclusivo no qual uma estação tem direito de transmitir seus pacotes [Casetti et al. 2005]. Com este parâmetro, o *HC* restringe a duração das transmissões das *QSTAs*, conseguindo um controle eficiente sobre o acesso ao meio, com isso, administrando melhor a qualidade de serviço oferecida aos diferentes fluxos de dados. Este intervalo de tempo possui um limite máximo (*TXOPLimit*) que é



definido de acordo com o tipo de mecanismo de transmissão em uso. A utilização deste conceito objetiva resolver os problemas de tempos de transmissão imprevisíveis e atrasos intrínsecos ao padrão IEEE 802.11, além de proporcionar justiça na alocação de tempo para utilização do meio por cada estação [Demarch 2007]. Este intervalo é controlado pelo *HC* e pode ser definido como *EDCA-TXOP* (referente ao *EDCA*) e *HCCA-TXOP* (referente ao *HCCA*). A duração do *TXOP* para o *EDCA* é definida por um parâmetro incluído nos quadros de *Beacon*, enquanto sua duração no caso do *HCCA* é especificada no quadro de *Polling*. O mecanismo HCF é ilustrado pela Figura 2.14.



**Figura 2.14:** Arquitetura da Camada MAC do IEEE 802.11e.

Pode-se destacar que uma *QSTA* não deve transmitir um pacote de dados se o período total necessário (incluindo os quadros de controle) extrapolar a duração da sua oportunidade de transmissão. De forma análoga, a *QSTA* não deve começar a transmissão dos dados se não puder terminá-la antes do instante previsto para o início de transmissão do próximo quadro *Beacon*, que é definido pelo tempo de transmissão do *Beacon* (*Target Beacon Transmission Time - TBTT*). Com isso, consegue-se evitar o problema da diminuição do *CFP*, isto é certo desde que não existam estações na rede que operem exclusivamente com o mecanismo *DCF*. A emenda IEEE 802.11e também permite que uma estação transmita múltiplos pacotes de dados em sequência (*Contention Free Burst - CFB*), contanto que a duração da referida oportunidade de transmissão não seja ultrapassada, visando o aumento da eficiência do protocolo MAC através da diminuição do *overhead* na rede.



### 2.4.2 Acesso Baseado em Contenção (EDCA)

O EDCA é um melhoramento da função DCF [IEEE 2005] e define quatro categorias de acesso (*Access Category - AC*) para os diferentes tipos de tráfego de dados, com suporte a 08 níveis de prioridade (*User Priority - UP*), cada conforme definido no Padrão 802.1D [IEEE 2004]. Cada nível de prioridade é mapeado numa categoria de acesso. As quatro categorias de tráfego são definidas como AC-BK, AC-BE, AC-VI e AC-VO, para segundo plano (*background*), melhor esforço (*Best Effort*), vídeo e voz, respectivamente, onde AC-BK tem o menor e o AC-VO tem a maior prioridade [Mangold et al. 2003]. Cada quadro da camada superior chega à camada MAC juntamente com um valor de prioridade. Este valor de prioridade é atribuído de acordo com o tipo de aplicação ou tráfego ao qual o quadro pertence. Há oito tipos diferentes de prioridades, onde os valores variam de 0 a 7 [Gao et al. 2005], esta definição é ilustrada conforme a Tabela 2.4.

**Tabela 2.4:** Mapeamento das Prioridades de Usuário para as Categorias de Acesso.

Prioridade	UP	AC	Tipo
Baixa	1	AC_BK	Segundo Plano
	2	AC_BK	Segundo Plano
	0	AC_BE	Melhor Esforço
	3	AC_BE	Melhor Esforço
	4	AC_VI	Vídeo
	5	AC_VI	Vídeo
	6	AC_VO	Voz
	Alta	7	AC_VO

Cada categoria de acesso é equivalente a uma função DCF independente que compete pelo acesso ao meio utilizando um conjunto de parâmetros diferentes, que incluem AIFS[AC] (*Arbitration Interframe Space - AIFS*), janela de contenção mínima (CW<sub>min</sub>[AC]) e janela de contenção máxima (CW<sub>max</sub>[AC]) e são difundidos no quadro de Beacon [Garg et al. 2003]. Estes parâmetros são referidos como parâmetros do EDCA. Quanto menor o valor dos parâmetros, maior será a probabilidade de acesso ao meio para a categoria de acesso correspondente. O HC pode otimizar esses parâmetros de forma dinâmica em função do tráfego na rede para melhor atender à qualidade de serviço das

aplicações. Para que o *HC* tenha prioridade no acesso ao meio em relação às demais estações, o período  $AIFS[AC]$  mínimo deve ser maior que o *PIFS* e igual a *DIFS*, ou seja:  $AIFS[AC] = SIFS + AIFSN[AC] * aSlotTime$ , onde  $AIFSN[AC]$  (*AIFS Number*) é um inteiro maior ou igual a dois para as estações [Ni 2005]. O diagrama temporal do *EDCA* é ilustrado na Figura 2.15.

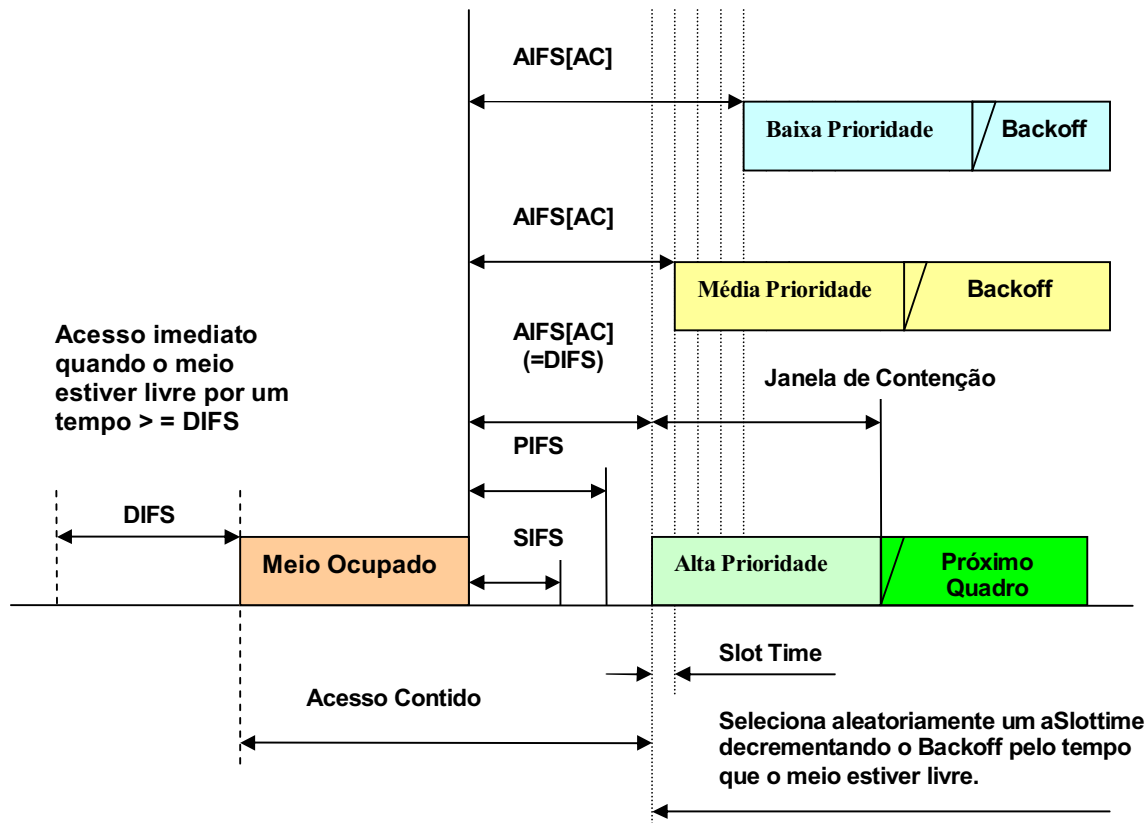


Figura 2.15: Esquema Básico do *EDCA*.

Cada categoria de acesso implementada na estação comporta-se como uma estação virtual individual, com a sua própria fila, competindo pelo acesso ao meio e executando o processo de *backoff* de forma independente [Bianchi et al. 2005]. A principal diferença é que não ocorrem colisões internas quando duas ou mais categorias de acesso da mesma estação terminam o processo de *backoff* simultaneamente. Neste caso, a estação atribui a oportunidade de transmissão (*TXOP*) à categoria de acesso de maior prioridade, enquanto as outras categorias de acesso comportam-se como se tivesse havido uma colisão externa, como podemos ver na Figura 2.16 [Pavón & Shankar 2004].

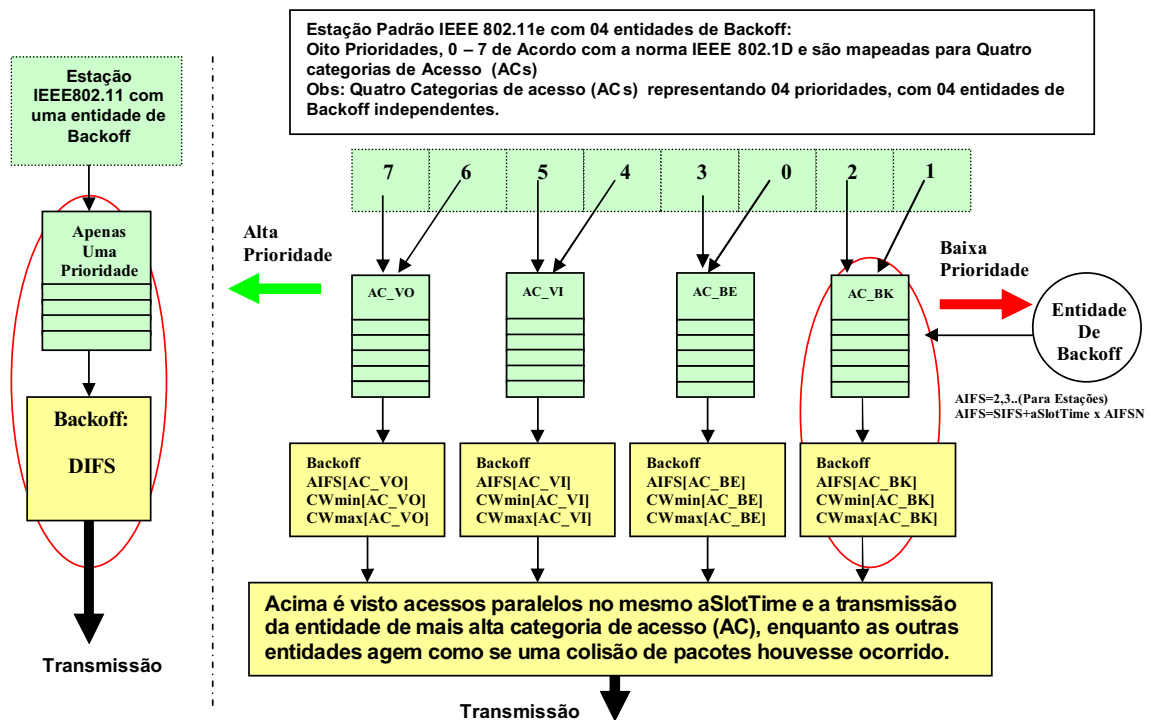


Figura 2.16: Comparação entre DCF e EDCA.

### 2.4.3 Acesso Controlado ao Canal (HCCA)

O mecanismo *HCCA* foi proposto para melhorar o *PCF* do Padrão IEEE 802.11 e se destina a garantir limites temporais (*delays*) com base no funcionamento do escalonador padrão *Round Robin*. O *HCCA* atua principalmente durante o período livre de contenção (*CFP*), mas também pode operar durante o período de contenção (*CP*). Isso é chamado de fase de acesso controlado (*CAP*). Um *CAP* termina quando o *HC* não recupera o canal após a duração de um período *PIFS* no final de um *TXOP* no intervalo do período de contenção. A transição entre o intervalo de repetição de dois *SI's* consecutivos é sinalizado com a transmissão de um quadro de *Beacon* [Grilo et al. 2003].

O *HC* é o responsável pelo controle de admissão e pelo escalonamento de tráfego, capaz também de obter o controle de acesso ao meio no *CFP* ou *CP*, esperando para isso um intervalo de *PIFS* entre as transmissões de estações usando os mecanismos *EDCA* e *DCF*, proporcionando, assim, qualidade de serviço às estações. O *HC* pode incluir o parâmetro *CFP* no quadro de *Beacon*, que se destina a informar todas as estações da rede para que definam os seus valores de *NAV's* para o tempo de duração da fase de acesso controlado [Yang 2005].

Para cada estação que recebe o *Polling*, é concedido um *TXOP* para transmitir o seu próprio tráfego ou quando o *HC* interroga uma estação que não tem pacotes a transmitir, esta deve enviar um quadro nulo (*QoS Null*), para só depois de todo o procedimento de transmissão o *HC* enviar um quadro de *CF-End* avisando todas as estações o final do *CFP* [Ye 2006].

O *HC* gera quadros de *Beacon* periodicamente a cada intervalo de serviço, onde cada estação já sabe antecipadamente quando o próximo quadro de *Beacon* vai ser transmitido [Mangold et al. 2003]. Isto resolve o problema da imprevisibilidade da transmissão das estações admitidas na lista de *Polling* do mecanismo *PCF* legado, no entanto as estações devem enviar suas requisições de *QoS* através de um quadro especial de gerenciamento chamado *Traffic Specification - TSPEC*, cabendo ao *HC* verificar se há recursos disponíveis suficientes para atender o *TSPEC* requerido pela estação. Se for o caso, pode-se propor um *TSPEC* alternativo, talvez com um nível de qualidade de serviço menor ou mesmo rejeitar a requisição de *QoS*. Na Figura 2.17, é ilustrado um exemplo de operação do *HCCA*.

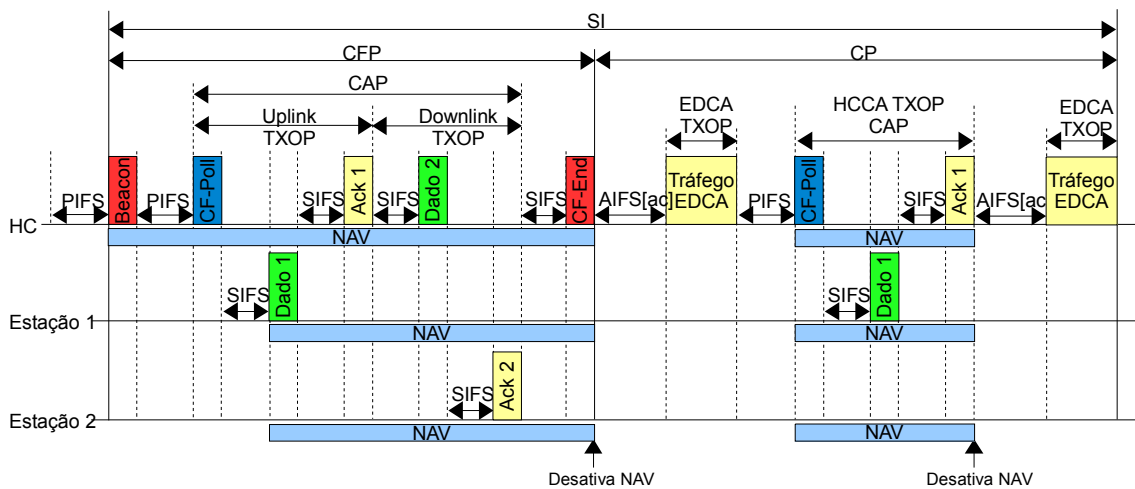


Figura 2.17: Superquadro Padronizado pela Emenda IEEE 802.11e.

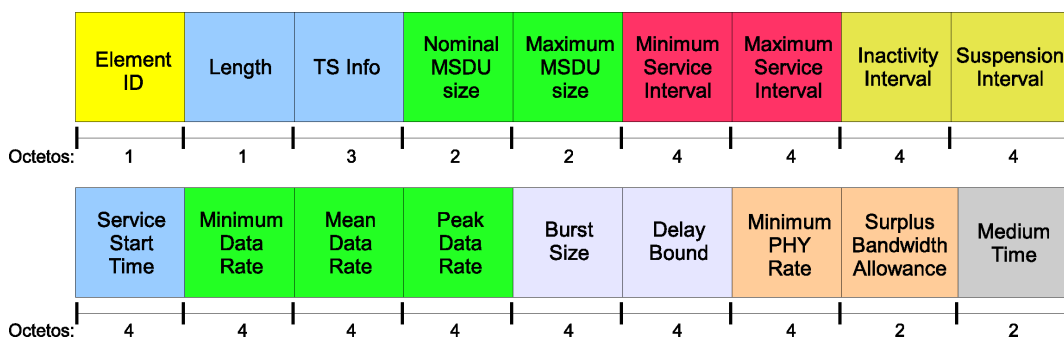
### Descrição do Quadro *TSPEC*

O quadro *TSPEC* contém o conjunto de parâmetros que definem as características de *QoS* do tráfego de rede que podem ser necessários ou estar disponíveis para qualquer instância específica de tráfego *QoS* parametrizado associadas a um dado fluxo unidirecional [Lim et al. 2004].

O mecanismo *HCCA* segue o seguinte processo para transmissão de dados: todas

as estações na lista de *Polling* devem enviar ao *HC* uma requisição de admissão para cada fluxo de dados que desejam transmitir, juntamente com as especificações do tráfego referido (*TSPEC*). As estações envolvidas na transmissão suportam oito fluxos de tráfego (*Traffic Stream- TS*) no sentido *uplink* e oito no sentido *downlink*. Desta forma, se um determinado fluxo for admitido pelo *HC*, um *TXOP* será atribuído à estação requisitante através do envio do quadro de *Polling*, e esta estará apta a transmitir seus pacotes de dados dentro de um intervalo de tempo especificado.

O valor do *TXOP* para o *HCCA* varia de  $32 \mu s$  a um limite máximo de  $8160 \mu s$  (*TXO-PLimit*) [IEEE 2005]. Caso o valor do *TXOP* seja  $0 \mu s$ , isto indica que um quadro nulo (*Qos Null*) ou apenas um quadro de dados deve ser transmitido em resposta ao quadro de *Polling* (*CF\_Poll*). As especificações de tráfego são definidas como um conjunto de parâmetros que definem os requisitos de *QoS* de um fluxo de dados para uma estação. A Figura 2.18 apresenta o formato do *TSPEC* com os parâmetros utilizados na especificação de tráfego [IEEE 2005, IEEE 2007].



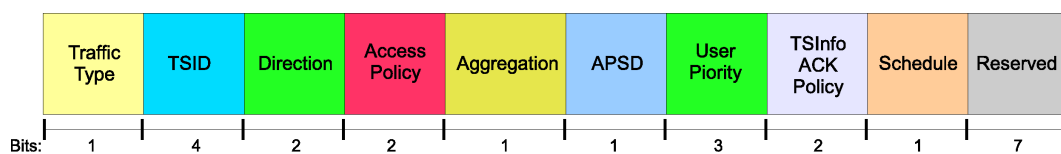
**Figura 2.18:** Formato do Quadro *TSPEC* segundo o Padrão IEEE 802.11e.

A seguir descreveremos resumidamente o formato dos campos do quadro *TSPEC*:

- Campo *Element ID* é a identificação do quadro *TSPEC*;
- *Length* é o tamanho em bytes do quadro *TSPEC*;
- *TS Info* é a informação do fluxo de tráfego, contém as informações relativas à identificação e configuração do fluxo de tráfego do *TSPEC*;
- *Nominal/Maximum MSDU size* determina o tamanho nominal e tamanho máximo do pacote de dados em bytes;
- *Minimum/Maximum Service Interval* determina os tempos mínimo e máximo de duração do intervalo de serviço (*SI*);

- *Inactivity/Suspension Interval* determina o intervalo mínimo de tempo que pode ocorrer sem a chegada de nenhum pacote antes que o fluxo seja descartado ou seja interrompido pelo HC;
- *Service Start Time* determina o tempo em que o período de serviço (*Service Period - SP*) deve começar;
- *Minimum/Mean/Peak Data Rate* definem, respectivamente: taxa mínima, média e máxima de transmissão dos dados em bytes por segundo;
- *Burst Size* determina o máximo tamanho em rajadas (*Burst*) de um pacote no *Peak Data Rate*;
- *Delay Bound* determina o atraso permitido para a transmissão de um pacote de dados;
- *Minimum PHY Rate* determina a taxa básica de transmissão de acordo com a camada física utilizada;
- *Surplus Bandwidth Allowance* especifica o acréscimo de tempo que pode ser alocado na transmissão para permitir que um fluxo de dados possa ser concluído e
- *Medium Time* determina o tempo médio para o meio ser acessado, porém não é utilizado pelo HCCA.

A Figura 2.19 apresenta o formato do quadro TS Info, onde o significado dos campos são os seguintes:



**Figura 2.19:** Formato do Campo TS Info Segundo o Padrão IEEE 802.11e.

- *Traffic Type* determina se o tipo de tráfego tem um padrão síncrono (taxa de transmissão constante) ou assíncrono (taxa de transmissão variável);
- TSID identifica o fluxo de dados;
- *Direction* determina o sentido do fluxo de dados como: *uplink*, *downlink* ou bidirecional;

- *Access Policy* determina qual o tipo de mecanismo de acesso ao meio que deve ser utilizado (*HCCA*, *EDCA* ou *HEMM*<sup>4</sup>);
- *Aggregation*, válido somente para o mecanismo *HCCA*, é usado no escalonamento agregado, que consiste no envio de quadros de dados e de *Polling* a uma estação durante um intervalo de tempo;
- *APSD (Automatic Power Save Delivery)*, usado para gerenciar o consumo de energia das estações; faz com que as estações fiquem em “*standby*” consumindo o mínimo de energia quando não houver transmissão;
- *User Priority* determina os valores de prioridades relativas aos pacotes pertencentes a um determinado fluxo de dados;
- *TS Info Ack Policy* determina quais são as políticas de confirmações utilizadas (com ou sem *ACK*) no recebimento dos pacotes e
- *Schedule* determina se deve ser utilizado, ou não, o escalonamento de tráfego.

As especificações de tráfego são definidas de acordo com os requisitos da aplicação. Porém, não está definido na emenda IEEE 802.11e [IEEE 2005] como devem ser geradas as especificações de tráfego, mas os parâmetros que geralmente são fornecidos pelas aplicações são o *Nominal/Maximum MSDU Size*, *Minimum/Maximum Service Interval*, *Inactivity Interval*, *Minimum/Mean/Peak Data Rate*, *Burst Size* e *Delay Bound*, e os parâmetros fornecidos pela subcamada *MAC* são o *Minimum PHY Rate* e *Surplus Bandwidth Allowance*. Usualmente o *HC* define os limites do *TXOP* como os menores possíveis com o objetivo de maximizar o *throughput*, seguindo as restrições impostas pelo *overhead* existente nos quadros de *Polling*, *ACK* e de *QoS DATA* [Demarch 2007].

#### 2.4.4 Controle de Admissão

O *HCCA* realiza o controle de admissão através de quadros de gerenciamento para determinar as especificações de tráfego. Para que uma estação possa transmitir um pacote com *QoS* pretendido, esta deve enviar uma requisição ao *HC* para solicitar admissão dos seus fluxos de dados. Desta forma, o *HC* pode ou não admitir o tráfego requerido de acordo com o algoritmo de escalonamento utilizado, verificando as condições do canal e os requisitos de *QoS*. O controle de admissão deve garantir que o tempo alocado para

---

<sup>4</sup>Conhecido como *HCCA*, *EDCA mixed mode*

o *CAP* não exceda o limite imposto pelo *CAPLimit* e também deve alocar um tempo adicional para possíveis retransmissões devido a ruídos no canal de comunicação.

A emenda IEEE 802.11e define dois controles de admissão para os mecanismos *EDCA* e *HCCA*.

**i) Controle de admissão para *EDCA*** - A estação solicita a admissão do seu tráfego de acordo com a categoria de acesso e, desta forma, o *AP* associa à prioridade do usuário (*UP*) recebida na requisição com a respectiva categoria de acesso (*AC*). No caso de uma *AC* não precisar de controle de admissão, a estação está autorizada a transmitir o seu tráfego sem negociar o *TSPEC* previamente. Entretanto, se em uma *AC* o controle de admissão for necessário e uma estação tem de transmitir sem o seu uso, a estação transmitirá os seus fluxos de dados usando os parâmetros de contenção de uma categoria de menor prioridade. Isto independe da exigência ou não do controle de admissão por uma categoria de acesso específica e o *AP* sempre deverá responder às requisições oriundas destas estações [Xiao & Li 2004].

**ii) Controle de admissão para *HCCA*** - O *HC* é responsável por admitir, ou não, um novo fluxo de dados de acordo com a *TSPEC* negociada com a estação. Para garantir *QoS*, o escalonador de tráfego deve satisfazer alguns requisitos. Neste caso, o escalonador deve atribuir *TXOP* a todos os fluxos de tráfego admitidos de acordo com a *TSPEC*; a duração mínima de uma *TXOP* deve garantir que pelo menos um pacote de tamanho máximo seja transmitido à taxa de transmissão mínima (Taxa Básica). Durante a negociação de *TSPEC*, parâmetros devem ser especificados, tais como *Mean Data Rate*, *Nominal MSDU Size*, *Minimum PHY Rate*, *Surplus Bandwidth Allowance* e, pelo menos, *Maximum Service Interval* ou *Delay Bound* para o quadro de requisição para admissão de tráfego, e todos estes, com exceção do *Delay Bound* para o quadro de resposta à requisição.[Grilo et al. 2003]

A emenda IEEE 802.11e não define nenhum algoritmo específico de escalonamento dos fluxos de dados para utilização no controle de admissão, apenas apresenta um escalonador de referência padrão para ser tomado como base [IEEE 2005].

## 2.4.5 Escalonador de Referência

De acordo com a emenda IEEE 802.11e, para o modo *HCF* existem dois tipos de escalonadores:



- Escalonador do *HC* - Controla o tráfego das estações, definindo o tamanho dos respectivos *TXOP*'s para cada estação e organiza a transmissão dos dados de todas as estações admitidas através do procedimento de *Polling*.
- Escalonador da *QSTA* - Responsável por escalonar os próprios fluxos interno de dados dentro do *TXOP* atribuído pelo *HC*, decidindo quanto tempo cada determinado fluxo de dados terá de disponibilidade para uso do canal.

O processo de escalonamento é iniciado quando um novo fluxo de dados é gerado na *QSTA*. Então, esta envia a especificação do tráfego (*TSPEC*), que pode admitir, ou não, estes novos fluxos. Após a negociação das *TSPECs*, o escalonador do *HC* realiza o escalonamento de tráfego com base nos parâmetros *Mean Data Rate*, *Nominal MSDU Size* e *Maximum Service Interval* ou *Delay Bound*.

Para a realização do escalonamento do IEEE 802.11e *HCF*, inicialmente são calculados dois parâmetros: Intervalo de Serviço e Oportunidade de Transmissão.

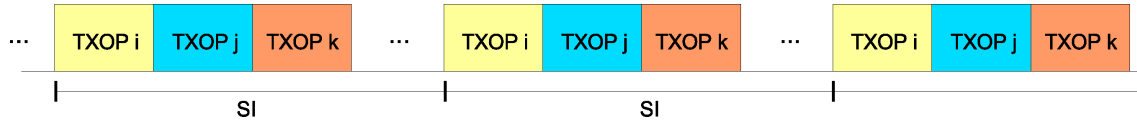
**a) Intervalo de Serviço (*SI*)** - Este parâmetro define o período do procedimento de *Polling* para cada estação com base nos valores do intervalo de *Beacon* e do Intervalo de Serviço Máximo (*Maximum Service Interval* - *MSI*).

No exemplo mostrado a seguir, o escalonador trabalha com o valor de *SI* fixo e igual para todas as estações, sendo recalculado quando novas estações são admitidas ou quando são retiradas da lista de *Polling*. Um valor mínimo é escolhido entre todos os *MSIs* de todos os fluxos de tráfego e o *SI* é determinado por um valor menor que este mínimo que deve ser submúltiplo do intervalo de *Beacon*.

**b) Oportunidade de Transmissão (*TXOP*)** - Logo após a determinação do *SI* é realizado o cálculo do *TXOP*, que dependem dos parâmetros *TSPEC* negociados, podendo ser diferente para cada estação.

**Exemplo de Cálculo de Escalonamento** - No exemplo proposto pela emenda IEEE 802.11e *HCF* [IEEE 2005], considera-se que o *MSI* mínimo é igual 60ms e que o intervalo de repetição do *Beacon* é igual a 100ms, um valor numérico imediatamente menor que 60ms e submúltiplo de 100ms é 50ms, que é atribuído ao *SI*.

Na Figura 2.20 foram consideradas três estações *i*, *j* e *k*, representando a operação básica do escalonador de referência, onde são atribuídos *TXOPs* às estações, dentro do intervalo de serviço calculado pelo *HC*.



**Figura 2.20:** Operação do Escalonador de Referência.

O *HC* determina a duração dos *TXOP* para cada estação em particular, e cada estação é responsável em dividir o *TXOP* atribuído entre os seus vários fluxos de dados. O tempo restante que não foi atribuído a nenhuma estação em particular será utilizado por todas no período de contenção através do mecanismo *EDCA*.

Para definir o *TXOP*, o escalonador utiliza os parâmetros negociados no quadro de *TSPEC*: *Mean Data Rate* ( $\rho$ ), *Nominal MSDU Size* ( $L$ ) e, também, os parâmetros *Service Interval* ( $SI$ ) calculado anteriormente, *Minimum PHY Rate* ( $R$ ), *Maximum Allowable Size of MSDU* ( $M$ ) e *overheads* (incluindo o espaçamento entre quadros (*IFS*) e dos quadros de confirmação (*ACK*) e de *Polling*) em unidades de tempo ( $O$ ). Com base nestes parâmetros, o escalonador efetua os seguintes cálculos:

1. Baseado nos valores de  $SI$  e do *Mean Data Rate*, o escalonador calcula a quantidade de pacotes ( $N_i$ ) que são gerados durante o intervalo de serviço, conforme a Equação 2.1.

$$N_i = \left\lceil \frac{SI \times \rho_i}{L_i} \right\rceil \quad (2.1)$$

2. O escalonador atribui à *TXOP* um valor máximo entre o tempo para a transmissão de  $N_i$  pacotes de tamanho  $L_i$  à taxa  $R_i$  e o tempo para a transmissão de um pacote de tamanho máximo à taxa  $R_i$ , com os respectivos *overheads*, conforme a Equação 2.2.

$$TXOP_i = \max \left( \frac{N_i \times L_i}{R_i} + O_i, \frac{M}{R_i} + O_i \right) \quad (2.2)$$

3. Para efetuar o controle de admissão de um novo fluxo, o escalonador determina inicialmente a duração do  $SI$  e dos *TXOPs*, e só admite o fluxo caso o mesmo satisfaça a Equação 2.3.

$$\frac{TXOP_{k+1}}{SI} + \sum_{i=1}^k \frac{TXOP_i}{SI} \leq \frac{T - T_{CP}}{T} \quad (2.3)$$

Onde  $k$  é a quantidade de fluxos existentes,  $k + 1$  é utilizado para indexar o novo fluxo admitido,  $T$  é o intervalo de tempo de repetição do *Beacon* e  $T_{CP}$  é o tempo reservado para tráfego das estações que disputam o meio utilizando o mecanismo *EDCA*.

## 2.4.6 Melhorias Obtidas pela Emenda IEEE 802.11e

Com o objetivo de melhorar o desempenho da rede, o IEEE 802.11e também fornece outros mecanismos para otimização dos requisitos de *QoS*:

1. **Reconhecimento em Bloco (*Block Acknowledgment - BlockACK*):** é uma extensão relacionada com o aumento de desempenho, pois eleva o *throughput* durante a comunicação de dados. Este mecanismo permite verificar uma rajada de pacotes (até 64 quadros) enviadas de uma *QSTA* sem que seja necessário que o receptor reconheça individualmente cada uma [IEEE 2005].

O padrão oferece dois tipos de *BlockACK*, um imediato e outro com atraso variável, que pode ser ativado através de um processo de negociação entre *QSTA* e *QAP*. Quando o mecanismo de *BlockACK* é estabelecido, vários pacotes de dados (*QoS DATA*) são transmitidos em rajadas pelo canal, com um intervalo *SIFS* entre quadros adjacentes. No final do bloco, todos os quadros de dados são reconhecidos por único quadro de *BlockACK* que contém um padrão de bits transmitidos no campo de dados do quadro [Mangold et al. 2003].

O *BlockACK* deve ser enviado antes que expire o *time out*, caso contrário, o remetente retransmite o pedido de confirmação de bloco. Depois de certa quantidade de tentativas de reenviar o pedido *BlockACK*, o conjunto de quadros de dados é descartado. Isso reduz o *overhead* da rede imposta pelo mecanismo *ACK* convencional, que requer a transmissão de um quadro *ACK* individual para cada quadro de dados transmitidos [Gao et al. 2005].

O tipo *BlockACK* imediato é útil para aplicações com alta demanda de largura de banda e baixa latência, o tipo com atraso variável é adequado para estações com um baixo desempenho, que são incapazes de processar o *BlockACK* imediatamente ou para aplicações que suportam latência moderada. A possibilidade de omitir quadros *ACKs* reduz naturalmente a confiabilidade do tráfego de dados, mas melhora por outro lado, a eficiência global para as transmissões de tempo real, onde os dados tem restrições de *deadlines*.

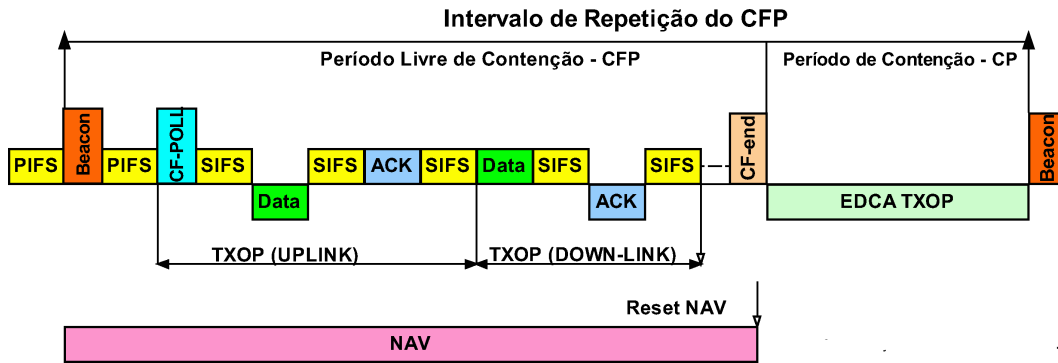
2. **Protocolo de Enlace Direto (*Direct Link Protocol - DLP*):** é mais uma inovação da extensão IEEE 802.11e. Este protocolo permite que duas *QSTAs* associadas ao mesmo *QAP* possam transmitir seus quadros de dados diretamente entre si sem necessidade de ser retransmitida pela *AP* formando uma pequena rede *ad-hoc*. Antes de iniciar a comunicação direta, as *QSTAs* envolvidas devem negociar com o *QAP* os parâmetros de transmissão utilizando o mecanismo de sinalização definido no referido protocolo. Assim que os dados não são mais enviados a comunicação direta é desestabelecida após um período de inatividade do meio de transmissão.
3. **Quadro sem Confirmação (*No Acknowledgment frame - No-ACK*):** geralmente são utilizados em transmissões de quadros em (*Broadcast/Multicast*), pois esses não precisam de confirmação ou em aplicações que têm requisitos *soft real-time*, que podem tolerar algumas perdas de pacotes.
4. **Atrelamento de Quadros (*Piggyback*):** é um mecanismo que permite o atrelamento ou *Piggyback* dos dados a serem enviados em quadros de *Polling* e *ACKs* para reduzir o *overhead* de transmissão de dados na rede [Gao et al. 2005].

#### 2.4.7 Motivação para Mudanças no Mecanismo *HCCA*

O mecanismo *HCCA* foi projetado para dar suporte de *QoS* a transmissão de rajadas de dados onde os impactos do *overheads* são minimizados, mas quando as estações necessitam enviar apenas um fluxo de dados (*traffic streams*) contendo apenas um pequeno pacote de dados (caso típico de automação industrial que exige requisito de tempo real) para outra estação utilizando um *AP*, há um aumento do *overhead* da rede. Como mencionado anteriormente, o *HCCA* define um Intervalo de Serviço (*SI*) com dois períodos.

A primeira parte é utilizada para o uso do período livre de contenção (*CFP*) por estações que são interrogadas pelo coordenador híbrido (geralmente embutido no *AP*) e têm suas oportunidades de transmissão (*TXOPs*) de *uplink* e *downlink* reservadas (período mais adequado a transmissão de dados em tempo real).

A segunda parte é utilizada como um período de contenção onde as outras estações (*DCF* e *EDCA*) que fazem parte da rede competem pelo acesso ao meio para transmitirem seus pacotes de dados (período mais adequado para transmissão de mensagens assíncronas, no caso de automação industrial, à transmissão de alarmes). Neste caso, pode-se ilustrar o diagrama temporal do *HCCA* como o da Figura 2.21.



**Figura 2.21:** Mecanismo de *Polling* Utilizado Pelo *HCCA* para um Fluxo de Dados.

A Tabela 2.5 apresenta os valores dos tempos de transmissão utilizando como base o valor de 40 bytes de dados e camada física do Padrão IEEE 802.11b.

**Tabela 2.5:** Tempo de Transmissão do Pacote de Dados de 40 bytes.

HCCA Parameters	Time of Transmission
SIFS	10 $\mu$ s
aSlotTime	20 $\mu$ s
PIFS	30 $\mu$ s
DIFS	50 $\mu$ s
PHY Header	192 $\mu$ s
MAC Header	26.18 $\mu$ s
Delay Transmission	2 $\mu$ s
Beacon Frame	826 $\mu$ s
CF-Poll Frame	482 $\mu$ s
Data Payload	249,27 $\mu$ s
ACK packet	306 $\mu$ s
CF-End Frame	354 $\mu$ s
TXOP UP Time	1061,27 $\mu$ s
TXOP DOWN Time	571,27 $\mu$ s
CAP Time	1642,54 $\mu$ s
CFP Time	2848,54 $\mu$ s

Agora, considerando que se deseja transmitir um pacote de dados de 40 bytes (*Payload*) por estação de forma síncrona utilizando o padrão IEEE 802.11b de acordo com a emenda IEEE 802.11e, temos que o *HCCA TXOP* pode ser definido como o tempo de transmissão de um pacote de dados utilizando *uplink* e *downlink*, incluindo os cabeçalhos da camada física (*PHY*), cabeçalhos da camada *MAC* a taxas básicas e dados a taxas nominais de transmissão. Onde a sobrecarga da interrogação (*CF-Poll Overheads*) é definida por:

$$CFPoll_{overhead} = CFPoll + SIFS \quad (2.4)$$

Os valores das *TXOP uplink* e *downlink overheads* são descritos, respectivamente, como:

$$Data = PHY_{overhead} + MAC_{overhead} + Payload \quad (2.5)$$

$$TXOP_{up} = CFPoll_{overhead} + Data + 2SIFS + ACK \quad (2.6)$$

$$TXOP_{down} = Data + 2SIFS + ACK \quad (2.7)$$

Nesta parte do desenvolvimento das equações é definido que a fase de acesso controlado (*CAP*) é a somatória de todos as *TXOPs* e é descrita como:

$$CAP = TXOP_{up} + TXOP_{down} \quad (2.8)$$

Finalmente considerando que o percentual de *overhead* médio do *HCCA* é o custo total de tempo de transmissão do pacote de dados (*Data*) descrito no *CAP*, teremos:

$$HCCA_{overhead}\% = 100 * [(CAP - Data)/CAP] \quad (2.9)$$

Agora, pode-se mostrar através de um gráfico, o custo do *overhead* no desempenho do *HCCA* em função da variação do número e tamanho dos pacotes transmitidos na rede. Aqui se pode observar que o desempenho do *HCCA* aumenta em função destes parâmetros, isto significa que quanto maior o tamanho e o número de pacotes transmitidos menor é o impacto do *overhead* na rede, como ilustrado na Figura 2.22.

Um outro aspecto importante é que no Padrão IEEE 802.11, geralmente o *overhead* associado à transmissão dos quadros *ACK* após cada quadro de dados recebidos sem erros tem impacto sobre a eficiência do protocolo de controle de acesso ao meio, tanto mais porque o quadro *ACK*, por ser um quadro de controle, é transmitido utilizando um modo de transmissão inferior (taxa básica) ao utilizado pelo quadro de dados que geralmente tem o tamanho pequeno. Isto não é desejável principalmente em comunicação industrial.

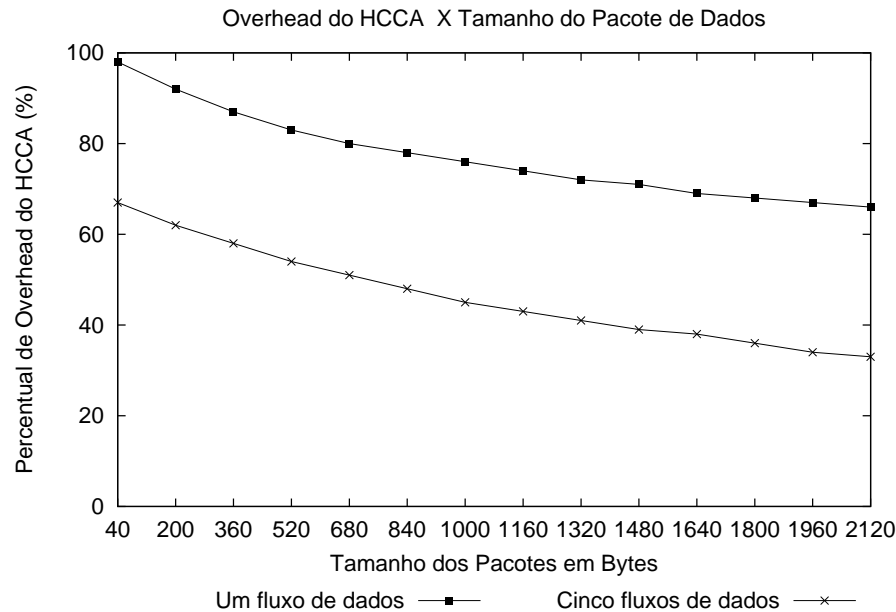


Figura 2.22: Desempenho do Mecanismo de *Polling* do *HCCA*.

## 2.4.8 Considerações sobre as Redes Padrão IEEE 802.11/11e

Neste capítulo foram apresentadas às tecnologias de comunicação sem fio baseadas no Padrão IEEE 802.11/11e; onde descrevemos as particularidades da camada física e de enlace, principalmente a subcamada de controle de acesso ao meio. O grande desafio é a utilização desta tecnologia em aplicações para automação industrial que demandam requisitos de tempo real.

Particularmente neste capítulo foram tratados os mecanismos de controle de acesso ao meio (*DCF* e *PCF*) e suas modificações para garantir qualidade de serviço utilizando a emenda IEEE 802.11e (*EDCA* e *HCCA*). Outro ponto importante que foi considerado referiu-se às possíveis modificações no mecanismo de transmissão do *HCCA*, que se mostra mais adequado para o tráfego de rede de automação industrial, onde os pacotes de dados transmitidos de forma periódica são pequenos e cumprem *deadline* rígidos admitindo-se poucas perdas no canal de comunicação. Neste caso o mecanismo *HCCA* tem um baixo desempenho. Desta forma é interessante notar que há espaço para novas propostas de mudanças no referido mecanismo. Baseado nesta assertiva, trataremos no próximo capítulo dos trabalhos relacionados com comunicação de tempo real para redes sem fio Padrão IEEE 802.11/11e.





---

## Capítulo 3

# Trabalhos Relacionados

---

O objetivo deste capítulo é descrever de forma sucinta os trabalhos relacionados à comunicação de tempo real em redes sem fio compatível com o Padrão IEEE 802.11 [IEEE 1999a] e com a emenda IEEE 802.11e [IEEE 2005]. Estas investigações foram norteadas pelos estudos obtidos na literatura científica atual, os quais revelam que mudanças no mecanismo de controle de acesso ao meio podem levar a uma maior eficiência em termos de desempenho da comunicação em redes sem fio, principalmente quando se trata da transmissão de pacotes pequenos com requisitos de tempo real.

### 3.1 Comunicação de Tempo Real para Redes Sem Fio

A demanda por alto desempenho em redes sem fio Padrão IEEE 802.11 já é um fato e tende a aumentar significativamente durante os próximos anos. Esta é uma consequência do recente desenvolvimento dessa tecnologia que já implementa acesso sem fio em ambientes de escritório, em áreas públicas e em ambientes domésticos [Moraes et al. 2007]. Então, é razoável esperar que no futuro próximo, as exigências impostas às redes sem fio com relação às restrições temporais para aplicações industriais, fiquem semelhantes aos impostos em redes cabeadas, onde o suporte à comunicação confiável é uma das principais exigências normalmente imposta para sistema de comunicação de tempo real para automação [Willig et al. 2005]. Em tais sistemas, a transmissão de dados (geralmente pacotes de tamanho pequeno) deve ser executada periodicamente entre sensores, controladores e atuadores de acordo com rígidos prazos de tempo (*deadline*). Adicionalmente, há uma recente tendência de se usar as mesmas infra-estruturas de comunicação industriais para transferir dados de controle em tempo real, tráfego multimídia genérico e tráfego relacionado aos dados de segundo plano (*background*). Desta forma, busca-se garantir a esses sistemas de tempo real que as exigências temporais destes pacotes de dados sejam respeitadas quando o meio de comunicação for compartilhado com diversos tipos de tráfegos

e sem restrições [Sauter & Vasques 2006]. Quando migramos parcialmente os sistemas de comunicação industrial (redes cabeadas) para o domínio de redes sem fios, deve ser considerado que o meio de comunicação sem fios é essencialmente um ambiente aberto de comunicação. Quer dizer, qualquer novo dispositivo pode a princípio tentar acessar o meio de comunicação a qualquer instante (de acordo com as regras de *MAC*) e estabelecer o seu próprio canal de comunicação [Moraes et al. 2007]. Como consequência, a carga de sistema não pode ser anteriormente prevista na configuração, nem pode ser controlada efetivamente durante o tempo de execução do sistema. Entretanto o suporte do protocolo de rede sem fio deve poder garantir as restrições temporais para tráfego de tempo real em um ambiente de comunicação industrial, caracterizado por uma carga previsível de tráfego oriundo de um número determinado de dispositivos de comunicação. Atualmente, o comportamento da comunicação de tempo real é garantido pelo controle rígido de todos os dispositivos de comunicação. A coexistência de estações controladas para tempo real junto com estações não controladas (genéricas) torna-se possível através de restrição de comportamento das estações não controladas.

A semelhança entre a rede cabeada Padrão IEEE 802.3 (*Ethernet*) e a rede sem fio Padrão IEEE 802.11 (*Wi-Fi*) é o uso do protocolo *CSMA* (*Carrier Sense Multiple Access*) para controle de acesso ao meio. Ao usar este protocolo, cada estação da rede verifica a ausência de tráfego antes de transmitir um pacote dentro do meio físico compartilhado, que pode ser um barramento elétrico ou uma faixa do espectro eletromagnético [Ye et al. 2000]. As redes *Ethernet* possuem um mecanismo de detecção de colisão chamado *CD* (*Collision Detection*) [Viégas Jr et al. 2006b]. Isto proporciona à estação escutar a rede enquanto transmite seus dados. Se uma estação detecta uma colisão, toda transmissão é interrompida e é emitido um sinal para anunciar que ocorreu uma colisão. Para evitar colisões sucessivas, a estação espera um período aleatório de tempo (*backoff time*) e volta a tentar transmitir [Viégas Jr et al. 2006a]. Diferente das redes *Ethernet*, as redes sem fio Padrão IEEE 802.11 não detectam colisões, e utilizam um outro mecanismo chamado *CA* (*Collision Avoidance*) para evitar colisões; este mecanismo possui parâmetros restritivos baseado em espaçamento temporal (*IFS*) que trabalha juntamente com o tempo de *backoff* entre a tentativa de transmissão de cada pacote, o que contribui para a redução da ocorrência de colisões em uma rede sem fio. De acordo com [Decotignie 2005], há várias propostas para "*Ethernet* Industrial" capazes de suportar comunicações de tempo real. Algumas destas propostas foram adaptadas para as redes IEEE 802.11. Porém, segundo [Cena et al. 2008] redes sem fio padrão IEEE 802.11 não são consideradas apropriadas para a conexão de dispositivos com requisitos de tempo real no chão de fábrica, como as aplicações de controle distribuído no nível de campo. Isto se deve principalmente pelo

mecanismo de controle de acesso ao meio que torna o comportamento das redes sem fio não determinístico [Willig et al. 2005]. Entretanto, várias técnicas existem para forçar o comportamento determinístico em uma rede sem fio, por exemplo, as especificações do mecanismo *PCF* pertencente ao Padrão IEEE 802.11 [Cloran 2004]. Porém, o *PCF* tem sérios problemas de sincronismo devido à existência de *delays* imprevisíveis e efetivamente este mecanismo não foi implementado pelos fabricantes para utilização prática no mundo real [Willig 2008] e não será tratado nesse capítulo. De outra forma, a emenda 802.11e [IEEE 2005] foi desenvolvida para satisfazer aos requisitos de qualidade de serviço para prover sistemas de tempo real definindo a função de coordenação híbrida, a qual opera com dois mecanismos distintos chamados *EDCA* e *HCCA* (já discutidos em detalhes no capítulo anterior). Desta feita, convém resumir as características desses mecanismos da seguinte forma:

- O mecanismo *EDCA* é uma melhoria do *DCF* pertencente o Padrão IEEE 802.11 e deve prover qualidade de serviço (*QoS*) através de quatro categorias de acesso (*AC*). Cada uma delas trabalha como uma entidade *DCF* virtual e independente para a priorização de tráfego que consiste no ajuste de parâmetros de contenção (*CWmin[AC]*, *CWmax[AC]*, *AIFS[AC]* e *TXOPLimit[AC]*) anunciados periodicamente nos quadros de *Beacon* gerado pelo coordenador híbrido (*HC*) [Demarch 2007]. Aumentando assim a probabilidade de acesso ao meio para uma determinada estação sem fio. Porém este esquema não está apto para assegurar um comportamento determinístico rígido do tráfego na rede. Logo, não é o mais indicado para operar em chão de fábrica [Cena et al. 2008]. Atualmente já existem disponíveis no mercado algumas implementações em placas de rede do mecanismo *EDCA* denominado *WMM* (*Wireless MultiMedia*) [Bellalta et al. 2006].
- O mecanismo *HCCA*<sup>5</sup> foi proposto para garantir exigências de atraso máximo (*delay*) dos pacotes de dados baseado por exemplo em um escalonador *Round Robin*, onde o Coordenador Híbrido (*HC*) interroga todas as estações presentes na lista de *Polling* de forma sequencial atribuindo uma oportunidade de transmissão (*TXOP*) que foi negociado anteriormente através do *TSPEC*, embora algumas estações possam não ter mensagens para transmitir, usando um ponto de acesso (*QAP*) [Yang 2005]. O *HCCA* tem como principal característica prover a parametrização de tráfego (requisitos definidos em valores quantitativos, como exemplo: *throughput*, *delay* e *jitter* da rede em períodos livres de contenção (*CFP*) e períodos de

---

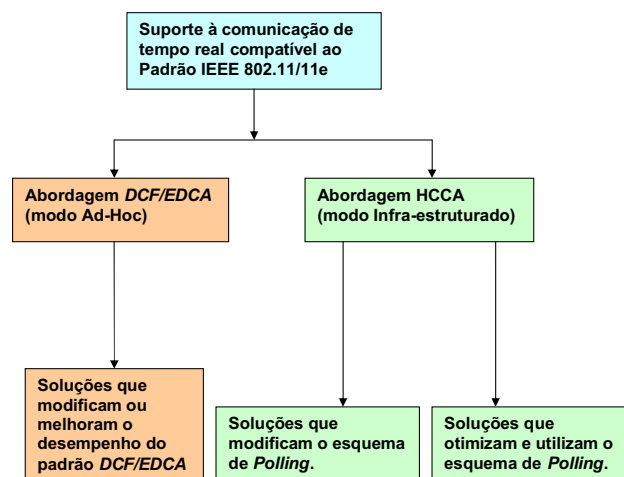
<sup>5</sup>Até o presente momento da elaboração desta tese não há equipamentos de redes disponíveis no mercado que suportem o mecanismo *HCCA*.

contenção (*CP*) para um meio compartilhado, isto acaba se tornando adequado e muito interessante para sistemas de comunicação aplicados à automação industrial [Cena et al. 2008].

Atualmente existem trabalhos inovadores sendo desenvolvidos para prover suporte à comunicação de tempo real para redes sem fio compatíveis com o padrão IEEE 802.11/11e que operem em modo *ad-hoc* (*DCF/EDCA*) e infra-estruturado (*HCCA*), ou seja, são soluções que não utilizam protocolos proprietários<sup>6</sup> para implementar o mecanismo de transmissão de dados. Neste capítulo, faz-se uma breve exposição das soluções que têm como base a abordagem *DCF/EDCA*, mas o foco é detalhar as soluções infra-estruturadas (*HCCA*), pois segundo [Cena et al. 2007] são mais adequadas para se implementar sistema comunicação para automação industrial. Basicamente os trabalhos relacionados podem ser classificados em dois grupos principais:

- 1) Soluções compatíveis com o Padrão IEEE 802.11/11e e operando em modo *DCF/EDCA*.
  - Soluções que modificam ou otimizam o mecanismo padrão do *DCF/EDCA*.
- 2) Soluções compatíveis com o Padrão IEEE 802.11/11e e operando em modo *HCCA*.
  - Soluções que otimizam e utilizam o esquema de *Polling* padrão do *HCCA*.
  - Soluções que modificam o esquema de *Polling* padrão do *HCCA*.

Na Figura 3.1 pode-se visualizar uma representação gráfica da classificação dos trabalhos relacionados envolvendo os dois grupos principais de soluções e seus subgrupos.



**Figura 3.1:** Abordagens Para Comunicação de Tempo Real em WLAN.

<sup>6</sup>Os grupos de soluções proprietárias apresentam ideias interessantes, mas até a presente data não são viáveis economicamente [Viéguas Jr et al. 2007] e estão fora do escopo desta tese.

### 3.1.1 Soluções Baseadas no DCF/EDCA

Várias propostas surgiram como alternativas em oferecer tempo real às redes sem fio, caracterizando-se por tentar impor diferentes prioridades às estações que disputam o acesso ao meio através de modificações no mecanismo *DCF/EDCA*. Dentro de cada mecanismo, diferentes abordagens foram usadas para realizar a diferenciação. O mecanismo *DCF* tem como objetivo fornecer um acesso ao meio com a mesma prioridade para grupos distintos. Para otimizar o desempenho, geralmente procura-se alterar o espaçamento entre quadros (*IFS*), utilizar o esquema de tamanho dos pacotes ou alterar o período de tempo aleatório (*Backoff Time*) para tentar prover um nível mínimo de *QoS*. Já o mecanismo *EDCA* pode prover melhorias nos resultados de *QoS* oferecidos ao serviço de comunicação através da introdução de categorias de acesso (*AC*). Cada categoria de acesso combina essencialmente dois parâmetros para fornecer *QoS*, que são as janelas de contenção (*CW<sub>min</sub>* e *CW<sub>max</sub>*) e diferentes *IFS* para diferentes classes de tráfego, juntamente com a sua arquitetura interna distribuída que abriga suas entidades virtuais de *Backoff*. Porém, em ambos os casos, os mecanismos *DCF* e *EDCA* trabalham de forma probabilística e, em parte, tentam garantir o determinismo no controle de acesso ao meio. A seguir, citaremos alguns dos trabalhos mais recentes e relevantes nesta área.

- Em [Sobrinho & Krishnakumar 1999], propõe-se uma solução baseada em contenção chamada *Black-Burst (BB)* que é um esquema de *MAC* distribuído aplicado para redes sem fio Padrão IEEE 802.11 operando em modo *ad-hoc*, que requer o desligamento do esquema de retransmissão randômica para minimizar o atraso para o tráfego de tempo real. Quando uma estação de tempo real deseja transmitir pacotes, esta acessa o canal em contenção e ordena seus direitos de acesso enviando um sinal de *BB* no canal. O comprimento do sinal de *BB* transmitido por uma estação de tempo real é uma função crescente do atraso de contenção experimentada pelo estação. As estações de baixa prioridade usam o mecanismo *DCF* para transmissão. Após transmitir o sinal *BB*, a estação de tempo real verifica o meio a procura de um outro sinal *BB* de maior duração, significando que outra estação esteve aguardando por mais tempo e portanto deve acessar o meio primeiro. Se não houver tráfego no meio, a estação envia o seu quadro. Caso contrário, espera que o meio esteja livre novamente para enviar outro sinal *BB*. Após o sucesso na transmissão do quadro, a estação determina o próximo instante em que tentará transmitir um novo quadro, sincronizando desta forma os seus fluxos de tempo real na rede. As principais limitações desta solução são prover apenas dois níveis distintos de *QoS* e ser sensível a distribuição física das estações na rede.

- Em [Vaidya et al. 2000], é utilizada a técnica que consiste na escolha de valores de *backoff* diferenciados para as janelas de contenção de cada estação pertencente a rede Padrão IEEE 802.11. Este valor de *backoff* é calculado proporcionalmente ao tamanho do pacote transmitido e inversamente proporcional ao peso atribuído ao fluxo que o pacote pertence. Esse esquema de diferenciação determina que estações que possuem maior peso escolham tempos de *backoff* menores e assim teriam maior prioridade de acesso ao meio. A utilização do tamanho do pacote (peso) tem um efeito de justiça para o compartilhamento da banda da rede. A técnica implementa o mecanismo de escalonamento *fair-queuing* de maneira distribuída. Garantindo que estações que transmitem pacotes maiores tenham prioridade no acesso ao meio. Esta estratégia pode levar à inanição *starvation* os pacotes pequenos.
- Em [Hwang & Cho 2005], é apresentado um esquema semelhante ao de [Sobrinho & Krishnakumar 1999], onde estações transmitem pacotes de voz (*Voice over Internet Protocol - VoIP*) usando *Energy-Burst (EB)* ou sinal de *Black Burst (BB)* para priorizar pacotes de tempo real em cima de pacotes de dados. O mecanismo proposto consiste em permitir a transmissão de pacotes de voz (prioridade mais alta) no primeiro *slot* de tempo vazio da primeira retransmissão. Quando a primeira retransmissão falhar, a segunda retransmissão executa o procedimento de *backoff* original.
- Em [Moraes et al. 2006], utilizou-se o mecanismo *EDCA* para prover comunicação de tempo real usando a categoria de acesso (*AC*) com prioridade mais alta (voz) para transferir mensagens com *deadlines* rígidos. No referido trabalho foi avaliado o comportamento desta categoria quando utilizada para transferir pacotes de tamanho pequenos (caso de automação industrial) em uma rede sem fio. Esta técnica não obteve o sucesso esperado, pois o número de perdas de mensagens e a tamanho médio das filas de transmissão apresentaram números inaceitáveis de perdas de *deadlines* para um sistema de tempo real, até mesmo para casos de tráfego intermediário na rede.
- Em [Hamidian & Korner 2006], propõe-se uma solução para otimizar o *EDCA*, permitindo estações com tráfego de prioridade mais alta reservar tempo para acesso ao meio livre de contenção. Basicamente, transfere-se o controle de admissão do *HCCA* e os algoritmos de escalonamento do *QAP* para as estações que disputam o meio. Esta solução permite a coexistência de dispositivos otimizados com ambos dispositivos padrão IEEE 802.11 [IEEE 1999a] e a emenda IEEE 802.11e [IEEE 2005] que operam nos modos *DCF* e *EDCA*.



- Em [Moraes et al. 2007], propõe-se uma arquitetura *ad-hoc* que permite a coexistência de estações padrão IEEE 802.11/11e com estações modificadas de tempo real no mesmo domínio de rede, impondo prioridade mais alta para o tráfego das estações de tempo real. A solução está baseada no protocolo de controle de acesso ao meio utilizando uma técnica de *VTP* para evitar colisões entre as estações de tempo real e as estações genéricas. Isto garante que sempre que uma estação de tempo real estiver disputando o canal, ela ganhará o acesso ao meio no período de contenção antes de qualquer estação genérica. O mecanismo de separação de tráfego que é usado na solução é semelhante ao protocolo proposto por [Work 2006] e [Carreiro et al. 2003]. Este protocolo especifica um valor de *backoff* igual à zero para estações com o nível de prioridade mais alta. O principal problema é que se múltiplas estações de tempo real tentam transmitir simultaneamente devido a uma falta de sincronismo do controle de acesso, colisões sucessivas acontecerão e uma falha de comunicação será informada as camadas superiores. Percebe-se que o problema acontece devido à perda de pacotes de tempo real, onde algumas estações não podem descobrir a transmissão de outras, por exemplo, para ambos o cenário de estação oculta e exposta.

### **Análise Comparativa das Soluções Baseadas no DCF/EDCA**

Na comparação entre as soluções apresentadas, as abordagens propostas em [Sobrinho & Krishnakumar 1999], [Hwang & Cho 2005] e [Vaidya et al. 2000] precisam modificar o protocolo de acesso ao meio para melhorar a previsibilidade das transferências de mensagem de tempo real. Logo, estas abordagens prejudicam a coexistência de outras estações IEEE 802.11 (melhor esforço) no mesmo domínio de rede. Em [Moraes et al. 2006] houve a intenção de aproveitar o mecanismo *EDCA* Padrão IEEE 802.11e para transmissão de mensagens de tempo real, apesar de totalmente compatível com o *DCF* não logrou êxito. A solução de [Hamidian & Korner 2006] tem limitações, pois se houver qualquer *QAP* que opere sob o modo de *HCCA*, a solução proposta não funcionará adequadamente, isto é uma dificuldade já que o ambiente de rede sem fio é aberto. Em [Moraes et al. 2007] foi desenvolvida uma solução para transmitir pacotes pequenos de tempo real, apesar da utilização de arquitetura *ad-hoc* ter contornado o problema do ponto de falha (*QAP*) da rede infra-estruturada, por outro lado, a referida solução não suporta muitas estações de tempo real introduzindo um complexo mecanismo de admissão e seu desempenho é bastante degradado quando o canal de comunicação estiver sujeito a ruídos, caso típico de utilização de redes sem fio em chão de fábrica.

### 3.1.2 Soluções Baseadas no HCCA

Segundo [Cena et al. 2008], o mecanismo *HCCA* definido no Padrão IEEE 802.11e é adequado para o tráfego de dados de tempo real, proporcionando um *QoS* parametrizado e utiliza-se do esquema de *Polling* para controle da comunicação. Porém as soluções que utilizam o mecanismo padrão de *polling* do *HCCA* sofrem as consequências de perda de desempenho devido ao *polling overhead* [Casetti et al. 2005]. Entretanto, trabalhos inovadores propõem novos algoritmos de escalonamento para melhorar o desempenho do *HCCA* para taxa de bits constante (*Constant Bit Rate - CBR*) e taxa de bits variável (*Variable Bit Rate-VBR*) tais como aplicações multimídias (Voz e Vídeo). Citaremos a seguir alguns trabalhos relevantes que utilizam esta abordagem:

- Em [Grilo et al. 2003], propõe-se um algoritmo de escalonamento para a Função de Coordenação Híbrida (*HCF*) do IEEE 802.11e. O *HCF* pode ser usado para fornecer ao endereço IP (*Internet Protocol*) garantias de qualidade de serviço em *WLANs* operando em modo infra-estruturado. O escalonador de pacotes é chamado de *Scheduling Based on Estimated Transmission Times - Earliest Due Date (SETT-EDD)*. O algoritmo proposto é compatível com os mecanismos de enlace implementadas em *WLANs* Padrão IEEE 802.11 comerciais, impondo limites à quantidade de tempo durante o qual cada estação pode controlar o acesso ao meio na rede sem fio. O desempenho do algoritmo é avaliado através de simulação computacional e comparado com o escalonador original do Padrão IEEE 802.11e (*Round-Robin*). Os cenários de simulação foram montados com a integração entre IEEE 802.11a/11e para uma *WLAN* infra-estruturada. Os resultados demonstram que *SETT-EDD* consegue melhor desempenho que o escalonador padrão, especialmente na transmissão de *streaming* de vídeo, que é um caso típico de *VBR*.
- Em [Lim et al. 2004] é descrita uma arquitetura *MAC* que, quando implementada em uma estação Padrão IEEE 802.11 (genérica), permite suportar as funcionalidades do padrão IEEE 802.11e. Existe nessa solução um algoritmo para garantir o acesso justo ao *TXOP* concedido a diferentes fluxos (*streams*) para uma determinada estação, combinando-os em uma única fila do tipo *FIFO (First In, First Out)*, permitindo assim o processamento mínimo de mensagens pelo *MAC* quando um quadro de (*Polling*) é recebido. Este trabalho também introduz um novo escalonador que ao ser utilizado em conjunto com o mecanismo *HCCA*, otimiza a entrega de mensagens de acordo com o *QoS* especificado para o tráfego na rede. O *MAC* do IEEE 802.11e é simulado usando *OPNET Software* e o desempenho dos mecanismos *EDCA* e *HCCA* são verificados sob vários cenários de tráfego multimídia



e dados. Os resultados mostram que o mecanismo *HCCA*, quando utilizado em conjunto com o escalonador proposto, pode simultaneamente satisfazer as especificações de *QoS* dos diferentes tipos de fluxos de dados, tais como, aplicações de *streaming* de vídeo (unidirecional) e aplicações de voz (bi-direcional com exigências rígidas de *deadline*) com melhor desempenho.

- Em [Cicconetti et al. 2005a], propõe-se um algoritmo de escalonamento para suporte à *QoS* em uma rede IEEE 802.11e utilizando o mecanismo *HCCA*. Este algoritmo utiliza um processo de programação *off-line*, que geram um agendamento de tarefas de forma periódica, e um processo de programação *on-line* em que se aplica posteriormente ao escalonamento dos *TXOPs* de forma adequada para as prioridades dos fluxos de mensagens *HCCA* negociados entre as estações e o coordenador híbrido. Esse escalonador é chamado de *Real-Time HCCA (RTH)* e visa minimizar o *polling overhead* mantendo uma complexidade computacional viável. Através de resultados preliminares de simulação computacional, foi mostrado que o *RTH* tem um melhor desempenho que o escalonador de referência *Round Robin*, principalmente otimizando o uso da largura de banda da rede.
- Em [Li et al. 2005], propõe-se uma análise da técnica de reconhecimento de bloco (*Block Acknowledgment - BA*) em função de uma possível melhoria no desempenho do controle de acesso ao meio. Foi apresentado um modelo teórico para avaliar o esquema proposto. O modelo levou em conta os efeitos das colisões e erros de transmissão ocorridos em um canal com ruídos. O modelo de simulação foi desenvolvido para o simulador *NS2 (Network Simulator 2)*. O esquema de reconhecimento de bloco (*BA*) foi simulado em uma rede com várias estações, onde as colisões eram o maior obstáculo. O esquema *BA* alcançou melhor desempenho utilizando a métrica *throughput* máximo, quando comparado com o tradicional reconhecimento utilizando o quadro de *ACK*. Os autores concluem que a técnica *BA* mostra-se promissora para uso na próxima geração de *WLANs* de alta velocidade, tal como as redes IEEE 802.11n.
- Em [Skyrianoglou et al. 2006], propõe-se um novo algoritmo para escalonamento de tráfego aplicado a redes infra-estruturadas IEEE 802.11e operando em modo *HCCA*. O algoritmo objetiva garantir *QoS* para tráfego multimídia sem quaisquer mudanças no Padrão. Este algoritmo foi denominado de *ARROW (Adaptive Resource Reservation Over WLANs)*. Os autores destacam como característica inovadora deste algoritmo a alocação do canal de comunicação com base no tráfego

de tempo real armazenado nas diversas estações móveis pertencente à rede em contraste com estimativas usadas pelas propostas anteriores. Esta característica segundo os autores torna o algoritmo proposto eficiente para o tráfego VBR. No entanto, foi também apresentada uma otimização do algoritmo para melhorar o desempenho para o tráfego CBR. O percentual maior de *overhead* de transmissão do escalonador proposto não mostrou-se ser significativo. O algoritmo proposto e seu aprimoramento são discutidos e avaliados em relação a dois outros escalonadores existentes, o escalonador padrão (*Round-Robin*) e o *SETT-EDD*. Os resultados obtidos através de simulação computacional mostram uma melhor utilização do canal de comunicação otimizando consideravelmente o desempenho em termos de atraso médio e perda de pacotes. Porém, observou-se a necessidade de se desenvolver um algoritmo de controle de admissão para evitar a degradação do canal quando a carga de entrada excede a capacidade máxima disponível.

### **Análise Comparativa das Soluções Baseadas no HCCA**

As abordagens apresentadas em [Skyrianoglou et al. 2006], [Grilo et al. 2003] e [Cicconetti et al. 2005a] propõem novos algoritmos de escalonamento para o uso no HCCA. Esses algoritmos são compatíveis com o padrão IEEE 802.11/11e e não requerem um ambiente de comunicação fechada para garantir comunicação de tempo real. Todos os escalonadores tentaram minimizar os efeitos do *polling overhead* e foram confrontados com o escalonador padrão *Round-Robin* obtendo melhor desempenho, especialmente para tráfego de taxa de bits variável (VBR), no caso de transmissão multimídia. A solução proposta por [Skyrianoglou et al. 2006] foi confrontada em termo de desempenho com a de [Grilo et al. 2003], obtendo melhor desempenho para VBR e para CBR. As métricas utilizadas foram o atraso médio e a perda de pacotes. Em [Lim et al. 2004] foi implementada uma nova arquitetura MAC para suportar o padrão IEEE 802.11e objetivando melhorar a previsibilidade das transferências de mensagens de tempo real em estações modificadas, introduzindo ainda um escalonador para trabalhar com o HCCA, que demonstrou ter desempenho superior para tráfego multimídia quando comparado com o escalonador padrão. Essa abordagem não prejudica a coexistência com estações IEEE 802.11 genéricas no mesmo domínio de rede.

Em contraste com estas soluções baseadas em escalonadores, foi descrita em [Li et al. 2005] uma análise da técnica de reconhecimento em Bloco (*Block ACK Scheme*). Nesse modelo foram levados em conta as colisões e erros de transmissão no canal quando da presença de ruído. A referida técnica obteve melhor desempenho que a tradicional que

usa o quadro *ACK* para confirmação dos pacotes. A métrica utilizada foi o *throughput*. Os autores consideraram viável a utilização da técnica nas redes IEEE 802.11n.

### 3.1.3 Soluções Baseadas em Modificações no *HCCA*

Algumas propostas buscam melhorar o desempenho do mecanismo *HCCA* através de modificações e ou otimizações no esquema de interrogação (*Polling Scheme*). A seguir, apresentamos alguns trabalhos mais relevantes que utilizam essa abordagem:

- Em [Lo et al. 2003], foi projetado um mecanismo de *multipolling* chamado de *Contention Period Multipoll (CP-Multipoll)*, que incorpora o esquema de acesso *DCF* dentro do esquema de *Polling*. Este esquema usa diferentes valores de tempo para inicialização do parâmetro de *backoff* para múltiplos fluxos de mensagens no grupo de *Polling*, onde cada estação executa o procedimento de *backoff* depois de receber o quadro de *CP-Multipoll*. A ordem de disputa ao meio destas estações é igual à ordem crescente de prioridade de acesso dado pelos valores de tempo de *backoff*. A estação na lista de *Polling* inicia sua transmissão imediatamente após o recebimento da *CP-frame Multipoll*. Esta ação evita a interferência de outras estações de executar os procedimentos de *backoff* no modo *DCF*. Além disso, a fim de evitar as colisões repetidas entre as estações que estão operando no mesmo canal em *BSSs* vizinhas, os valores atribuídos no *CP-Multipoll BSSs* devem ser diferentes.
- Em [Son et al. 2005], propõe-se a implementação de um sistema simples e eficaz de *Polling* para reduzir o número de interrogação para uma determinada estação que não tem pacotes para transmitir, onde o *HC* pode punir as estações que não têm pacotes para transmitir. Quando uma estação transmite um quadro nulo, esta estação não será interrogada novamente durante um período de tempo no decorrer de vários intervalos de serviço (*SI*). Também, neste esquema quando uma estação é interrogada e transmite um pacote de dados, esta pode obter *Polling* com mais frequência, reduzindo assim, o seu período de transmissão de dados. Por exemplo, os resultados das simulações mostram que há aumentos de desempenho de 35,8 % em quinze das trinta estações da lista de *Polling* que têm pacotes para transmitir utilizando a tecnologia IEEE 802.11a, mas em condições normais de transmissão de dados, não existem vantagens em relação ao *HCCA* original.

- Em [Lee & Kim 2006], propõe-se um algoritmo para lidar com um problema do *CF-Poll piggyback*. O fenômeno tem origem quando um quadro de *CF-Poll* é usado para garantir o canal para uma determinada QSTA e pode ter atrelado (piggyback) os seus dados de *QoS* para aumentar o desempenho. No entanto, o esquema de *CF-Poll piggyback* pode diminuir a eficiência da comunicação quando qualquer uma das QSTA associadas no QBSS usar uma baixa taxa de transmissão física para os seus dados. Para verificar este problema, foi realizada uma simulação computacional utilizando tráfego de voz e vídeo para o *HCCA* utilizando-se uma taxa mínima de transmissão física. Usando os resultados da simulação para carga de tráfego normalizada, observou-se que o atraso do quadro de dados, incluindo a *CF-Poll* aumentou cerca de 4,75 msec no pior caso. Foi encontrado um ponto de intersecção da eficiência da *CF-Poll piggyback* em relação ao atraso dos pacotes e está localizado dentro da faixa de 24 e 36 Mbps para a rede IEEE 802.11b/g.
- Em [Milhim & Chen 2007], propõem-se um esquema de *Polling* adaptativo, que trabalha dentro do *HC* para o mecanismo *HCCA*. O *HC* mantém duas listas de *Polling* dinâmicas para reduzir tanto o atraso da rede (*delay*) e o *polling overhead*. Os tráfegos VBR e CBR são levados em consideração nesta solução e os resultados das simulações mostraram que o *polling overhead* é reduzido significativamente, além do alto *throughput* e baixo atraso da rede (*Delay*) em comparação com o clássico escalonador *Round-Robin* de referência do IEEE 802.11e.
- Em [Chen et al. 2008], propõe-se um mecanismo *HCCA* modificado chamado *Non-Polling Based HCCA (NPHCCA)*. Esta solução evita a interrogação (*polling*) de todas as estações da rede, quando estas não têm pacotes pendentes a transmitir (evitando o desperdício de tempo) utilizando o quadro de *Beacon* gerado pelo *AP* de modo a melhorar o desempenho da transmissão. As estações que possuem quadros pendentes para transmitir serão organizadas em uma sequência adequada através da troca de mensagens através do *AP*. No cenário de simulação assume-se que existe um *AP* e várias estações na rede sem fio. A camada física adotada foi o IEEE 802.11b. Todas as estações inicialmente têm cem quadros de dados com três diferentes prioridades (alta, média e baixa), a ser transmitida, e o tamanho do quadro é de 512 Bytes. O tempo de simulação é de 20 segundos e utiliza-se o simulador de redes chamado *QualNet*. Os resultados da simulação computacional mostram que a melhoria do desempenho do mecanismo proposto é significativo para suporte a *QoS* em rede infra-estruturada, particularmente sob condição de carga leve.

### **Análise Comparativa das Modificações no HCCA**

A abordagem apresentadas em [Lo et al. 2003] propõe um novo mecanismo de *polling* para o uso no HCCA. Nessa proposta foi projetado um mecanismo de *multipolling* que executa o *backoff* depois de receber o quadro *CP-Multipoll*, este procedimento evita a interferência de outras estações que operam no modo *DCF* no mesmo domínio de rede. Este mecanismo de *polling* não é compatível com o padrão IEEE 802.11e, requerendo um ambiente de comunicação fechada para garantir comunicação de tempo real. Enquanto, a proposta de [Son et al. 2005] e [Chen et al. 2008] implementam um mecanismo de *polling* que tenta minimizar o efeito das estações que não têm mensagens a transmitir e em [Milhim & Chen 2007] é proposto um esquema de *polling* adaptativo visando reduzir o *Delay* e o *Polling overhead* em comparação com o tradicional esquema de *polling* utilizando o escalonador *Round-Robin*. Os resultados das simulações destas abordagens mostram que há uma melhora no desempenho da rede sob condição de carga leve, mas em condições normais de transmissão de dados, não há vantagens sobre o HCCA original. Em concordância com as abordagens baseadas em mecanismo de *polling*, foi descrita em [Lee & Kim 2006], um algoritmo para tratar o problema do atrelamento dos dados enviados (*piggyback*) no quadro de *polling* para uma determinada estação que utiliza uma baixa taxa de transmissão física para o seus dados. Nesse modelo foram levadas em conta as transmissões multimídias. A referida técnica encontrou um ponto de intercessão da eficiência do problema do *CF-Poll Piggyback* em relação ao atraso dos pacotes localizada dentro da faixa de 24 e 36 Mbps para a rede IEEE 802.11b/g.

### **3.1.4 Soluções para Aplicações Industriais**

Todas as soluções apresentadas anteriormente foram projetadas para otimizar o mecanismo de transmissão do HCCA para sistemas multimídias (Voz e Vídeo), porém não evitam o baixo desempenho para transmissão de pequenos pacotes de dados em presença de *deadlines* rígidos, como é o caso em cenários de automação industrial [Willig 2008] [Cena et al. 2008], apesar de algumas abordagens economizarem largura de banda quando as estações interrogadas pelo AP não têm pacotes de dados a transmitir [Casetti et al. 2005]. Para ilustrar, nós podemos citar os seguintes trabalhos iniciais na área de automação de processos:

- Em [Lee et al. 2005], é descrito uma solução utilizando redes IEEE 802.11b para indústria. Cita-se que essas redes apresentam um desempenho instável sob o tráfego pesado e os atrasos de transmissões não são constantes. Como tentativa de

contornar a natureza probabilística devido ao uso do protocolo *CSMA/CA*, é apresentada uma arquitetura de quatro camadas usando a especificação de interface de driver de rede (*Network Driver Interface Specification - NDIS*) trabalhando juntamente com um algoritmo de *polling* virtual que tem o objetivo de impedir colisões entre as mensagens e manter os *deadlines* dentro de valores admissíveis. Com base nos resultados experimentais, verificou-se que a arquitetura proposta para as redes IEEE 802.11b é uma alternativa válida para implementar redes sem fio industrial. Porém constata-se a necessidade de mais pesquisa, incluindo o desenvolvimento de aplicações para ambientes industriais reais, como sistema de monitoramento remoto. Além disso, é necessário implementar um controle de admissão e mecanismo de tolerância a falhas para o caso da perda de quadros de *polling*.

- Em [Trsek et al. 2006], é avaliado o desempenho dos mecanismos *HCCA* e *EDCA* para uma cenário de rede de automação industrial com requisitos rígidos de tempo real utilizando o simulador de rede OPNET. Os resultados são comparados em termos de latência em vários cenários. Descreve-se também o modelo utilizado para implementar o *HCCA* no simulador. Argumenta-se que o mecanismo *HCCA* é superior ao *EDCA* em cenários com um grande número de estações, por causa dos fluxos de tráfego pré-determinado aliado ao uso de mecanismo de *polling*. Como esperado, o mecanismo *EDCA* é penalizado quando um alto número de estações tenta transmitir quadros com a mesma prioridade levando ao um aumento de carga da rede. Em contraste com este comportamento, o *EDCA* pode ser o mais adequado para as configurações com um número muito pequeno de clientes. De acordo com os resultados obtidos, conclui-se que o *HCCA* é mais adequado para a prestação de garantias de tempo real para certas aplicações industriais, embora haja alguma restrição. No entanto, é necessário continuar a investigar os mecanismos *HCCA* e *EDCA* em cenários cujo fluxo de tráfego tem características de aplicações industriais. Especialmente, melhorias propostas para os algoritmos de escalonamento do *HCCA* e otimização dos conjuntos de parâmetros do *EDCA*. Além disso, desenvolver modelos nos simuladores de redes que se aproximem ao máximo possível de sistema de tempo real operando em canal com ruídos para se obter resultados mais realistas.
- Em [Cena et al. 2007], argumenta-se que a adoção de tecnologias de comunicação sem fio em ambientes industriais para suporte a aplicações em tempo real não críticos (*soft real-time*) depende fortemente da capacidade de garantir *deadlines* para as mensagens, pelo menos do ponto de vista probabilístico. Este aspecto é particular-



mente importante em sistemas de automação de fábrica, onde os tempos de resposta são considerados muito mais significativos do que outros parâmetros de desempenho, tais como *throughput*, que são normalmente considerados em diferentes áreas de aplicação. Nesse trabalho são descritas algumas medidas experimentais dentro de um contexto industrial real, utilizando-se tecnologias de redes IEEE 802.11g/e para melhor entendimento das respostas temporais das distribuições estatísticas dos pacotes de dados que podem ajudar a caracterizar estas soluções quando utilizado sistemas de tempo real não críticos.

- Em [Vitturi et al. 2007], é analisado o desempenho obtido por um protótipo de protocolo de camada de aplicação adequado para comunicações de tempo real aplicado à indústria. Esta proposta é derivada de protocolos empregados atualmente para rede *Fieldbus* cabeada, funcionando através de dispositivos baseados em dois padrões populares de comunicação sem fio, o IEEE 802.15.4 (*Low-Rate Personal Area Network - LR-WPAN*) e IEEE 802.11 WLAN. A metodologia apresentada foi baseada em um mapeamento de aplicações dentro dos serviços oferecidos pelos sistemas de redes sem fio e foram discutidas algumas possibilidades de escolha dos parâmetros do projeto da WLAN. Foram realizados vários testes em ambos os sistemas, sob diferentes cenários empregando várias configurações de rede. Em particular, foi simulado o comportamento de uma rede sem fio em um ambiente industrial utilizando um canal com ruídos. A análise dos resultados mostrou o desempenho em boa concordância com aqueles disponíveis na literatura científica. Em geral, no entanto, para ambas as implementações, foi observado comportamentos piores do que os indicados pela análise dos protocolos. Uma investigação revelou ainda que essa diferença é relacionada a diversos fatores, tais como: a implementação dos padrões de comunicação pelos fabricantes, a estrutura da camada de aplicação desenvolvida e os tempos de execução de software para os componentes específicos do sistema de tempo real.
- Em [Cena et al. 2008], é analisado o desempenho de redes IEEE 802.11e utilizando o mecanismo *EDCA* modificado para suportar a comunicação em ambientes industriais. Os tipos de dados que podem ser normalmente encontrados nesses ambientes foram caracterizados com base nos seus requisitos típicos e foi introduzido um esquema para mapear essas classes de tráfego para categorias de acesso *EDCA*. Em seguida, um conjunto de simulações foi realizado para avaliar o desempenho levando em conta o efeito das interferências (ruídos) entre as diferentes classes de tráfego. Os resultados foram obtidos através de percentual de latência que, na opi-

ção dos autores é a métrica mais interessante para sistemas de controle de tempo real que não exijam requisitos rígidos de tempo (*Soft Real Time*). Os experimentos mostraram que o *EDCA* (melhorado através de técnicas de TDMA) poderia oferecer uma solução satisfatória, quando a segurança e/ou requisitos críticos de tempo não são fundamentais. A partir da análise das simulações, as seguintes regras foram obtidas: a taxa de alarmes geralmente deve ser mantida baixa (menor que 3%). Devido ao processo de coordenação do esquema de transmissão, as transmissões de dados podem crescer até 15%, ao passo que a perda de quadros de alarme continua a ser aceitável somente se forem mantidos abaixo de 2%. Em qualquer caso, o tráfego total de tempo real não deve exceder cerca de 20% da carga máxima da rede. O desempenho resultante, em termos de tempos de intervalo de serviço (SI), latências e jitters, que foram obtidas assemelham-se com redes *Fieldbus* atualmente existentes. Porém, ao contrário das redes *Fieldbus*, as redes IEEE 802.11e que utilizam o mecanismo *EDCA* apresentam um nível visivelmente menor de confiabilidade e determinismo.

### 3.1.5 Considerações sobre Tempo Real para Redes Sem Fio

Existem várias proposições alternativas para utilização de redes IEEE 802.11/11e para sistema de tempo real em rede sem fio. No passado, a maioria das pesquisas concentrou-se em melhorias para o mecanismo *DCF* pertencente ao padrão IEEE 802.11, uma vez que esse mecanismo é obrigatório para a implementação de WLANs. Porém, desde que a emenda IEEE 802.11e foi publicada, juntamente com a crescente disponibilidade no mercado de produtos baseados no padrão, esta tecnologia tornou-se bem interessante para uso em sistemas de controle distribuídos.

Atualmente a maioria das pesquisas encontram-se centradas sobre o desempenho oferecido pelas camadas inferiores da pilha de comunicação e tem como foco a otimização dos mecanismos de transmissão: *HCCA* para *QoS* parametrizado e *EDCA* para *QoS* priorizado.

As diferentes soluções baseadas em *EDCA* e *HCCA* foram apresentadas neste capítulo e abordaram o problema da garantia de *QoS* de tal forma, que procurou-se mostrar várias soluções propostas na literatura técnica, onde foram rapidamente descritas e comparadas. Apesar de todas as soluções terem mais ou menos o mesmo objetivo de melhorar o *QoS* em uma rede sem fio, não são suficientes para concentrar todos os tipos de aplicações. Algumas suportam a transmissão de dados para tráfego multimídia, outras são mais eficientes para aplicações de tempo real com prazos rígidos.



Em particular para o *EDCA*, quando utilizados poucos dispositivos, é demonstrado que pode ser adequado em determinadas situações para operar no chão de fábrica, desde que não sejam exigidas restrições de tempo muito rígidas. Já em relação ao *HCCA*, as soluções em geral têm um melhor desempenho (cumprem *deadlines* rígidos) para um grande número de dispositivos do que as baseadas em *EDCA*.

Apesar de algumas melhorias no escalonador original *HCCA* terem sido recentemente propostas, porém, uma desvantagem do *HCCA* é que todos os quadros precisam atravessar o *QAP*, desperdiçando largura da banda e fazendo a comunicação pouco eficiente devido ao *overhead* de transmissão para pequenos pacotes. Outro aspecto é que o *QAP* é um único ponto de falha que pode fazer a rede inteira avariar [Casetti et al. 2005]. Além disso, outros estudos preliminares de [Casetti et al. 2005] e [Garg et al. 2003] mostraram que o mecanismo *HCCA* pode não ser satisfatório para garantir as exigências para aplicações industriais sob certas condições. O cumprimento dos *deadlines* e as oportunidades de transmissões iguais para cada um dos dispositivos, são especialmente importantes para o bom desempenho de uma rede industrial.

Baseado nestas assertivas apresentaremos no próximo capítulo a proposta intitulada “*GSC - Group Sequential Communication*”. A referida solução propõe uma nova abordagem para trabalhar em redes sem fio como alternativa ao mecanismo de *Polling* do *HCCA*, focado em suprir a lacuna de transmissão de pequenos pacotes de tempo real rígido obtendo maior desempenho quando aplicado à redes de automação industrial.



---

# Capítulo 4

## O Mecanismo *GSC*

---

Este capítulo descreve uma nova proposta para tratar o problema de controle de acesso ao meio com requisitos de tempo real para aplicações em redes industriais sem fio. A proposta foi chamada de *Group Sequential Communication (GSC)* e foi concebida para manter os requisitos de separação e parametrização de tráfego síncrono de tempo real utilizando o conceito Produtor/Consumidor em uma rede infra-estruturada compartilhada com estações genéricas (IEEE 802.11/11e), contornando o problema do baixo desempenho para transmissão de pequenos pacotes do mecanismo *HCCA*.

### 4.1 Visão Geral

Atualmente há uma grande disseminação da tecnologia de rede sem fio Padrão IEEE 802.11 e podemos notá-la no cenário cotidiano com cada vez mais frequência. Estudos recentes têm abordado a adequação das atuais tecnologias sem fio para comunicação industrial que geralmente operam com severas restrições de tempo. No entanto, o mecanismo de controle de acesso ao meio implementado não permite que as *WLANs* sejam determinísticas e portanto tornam-as inadequadas para tráfego de tempo real.

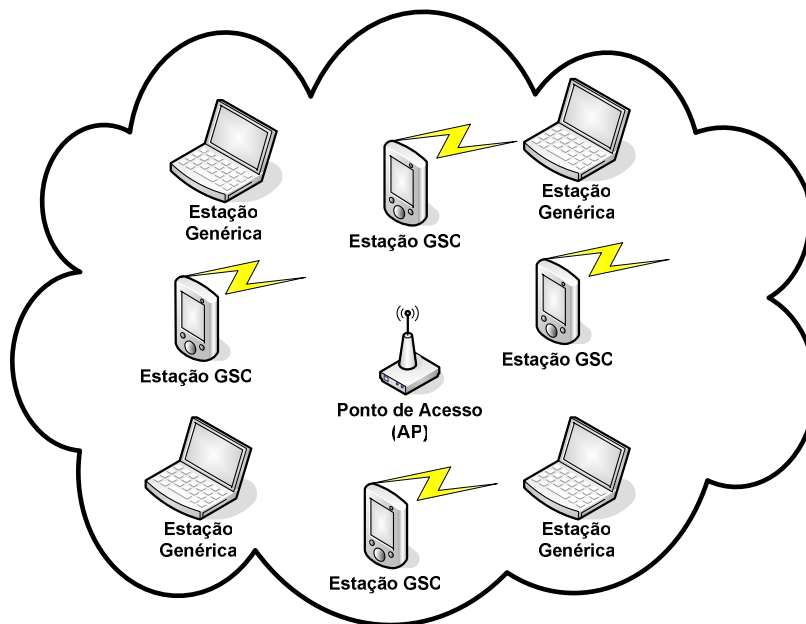
Aplicações em redes de automação industrial caracterizam-se pela transmissão de pequenos pacotes de dados que são enviados ao destino geralmente de forma periódica, admitindo pequenas taxas de perdas [Willig 2008]. Estes fatores são de fundamental importância para a eficiência da comunicação. Desta forma a habilidade da rede para prover qualidade de serviço vai depender diretamente do mecanismo de controle de acesso ao meio e do escalonamento das mensagens de tempo real.

A proposta *GSC* vem aumentar a eficiência do mecanismo de controle de acesso ao meio, quando comparado ao mecanismo *HCCA* para a transmissão de pequenos pacotes utilizando sistema de tempo real rígido, através da redução dos *overheads* da rede compostos pelos quadros de *Polling*, *Ack* e *QoS Null* para aplicações de automação industrial,

endereçando as seguintes questões:

1. Substituir o mecanismo *HCCA* pelo *GSC* para suportar com melhor desempenho a comunicação de tempo real em redes sem fio aplicada à automação industrial.
2. Ser compatível com o padrão IEEE 802.11e existente (tanto para as estações, como para o ponto de acesso), implicando portanto em pequenas modificações no hardware/software do *AP* e das estações de tempo real (*GSC Stations*).
3. Indicar a implementação da solução em plataforma aberta para *hardware/software* de baixo custo.

A proposta *GSC* foi concebida para uma rede sem fio do tipo infra-estruturada compatível com o Padrão IEEE 802.11e. Para isso utiliza-se o Coordenador Híbrido (*HC*) que geralmente está embutido no *AP* e tem a função de controlador de acesso à rede industrial. Na Figura 4.1, visualiza-se o cenário básico de operação da proposta, onde estações genéricas e de tempo real compartilham o mesmo meio de transmissão.



**Figura 4.1:** Cenário Proposto para o *GSC*.

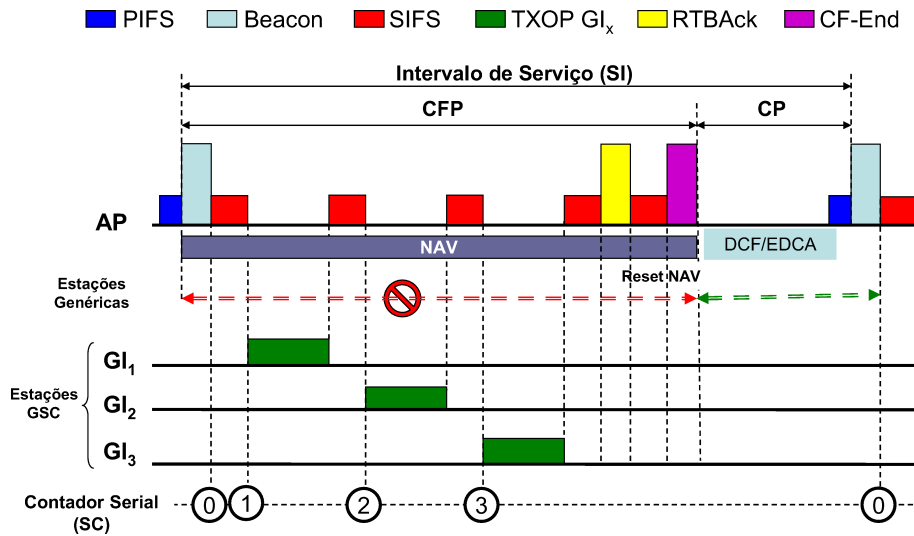
Do ponto de vista de conceito de projeto, o *GSC* funciona de forma análoga ao mecanismo *HCCA* descrito no padrão IEEE 802.11e. Como no *HCCA*, existe o conceito de superquadro, que é composto pelo período livre de contenção (*CFP*) e pelo período

de contenção (*CP*). Porém, as estações que utilizam o *GSC* podem transmitir mensagens de tempo real somente no *CFP*, liberando o meio para as estações genéricas (IEEE 802.11/11e) transmitirem suas mensagens no *CP* via mecanismo *DCF/EDCA*.

O intervalo de tempo entre *CFPs* consecutivos é chamado de Intervalo de Serviço (*Service Interval - SI*). Um *CFP* começa quando o controlador híbrido (*HC*) obtém o controle do meio de transmissão através de um período *PIFS* de inatividade do canal. De posse do controle do meio, o *HC* envia um quadro de *Beacon*. Este quadro de gerenciamento atribui um valor de tempo ao temporizador de rede (*Network Allocation Vector - NAV*) para todas as estações genéricas, mantendo-as proibidas de transmitir até o final do *CFP*. Durante este período, as estações *GSC* terão acesso exclusivo ao meio de transmissão.

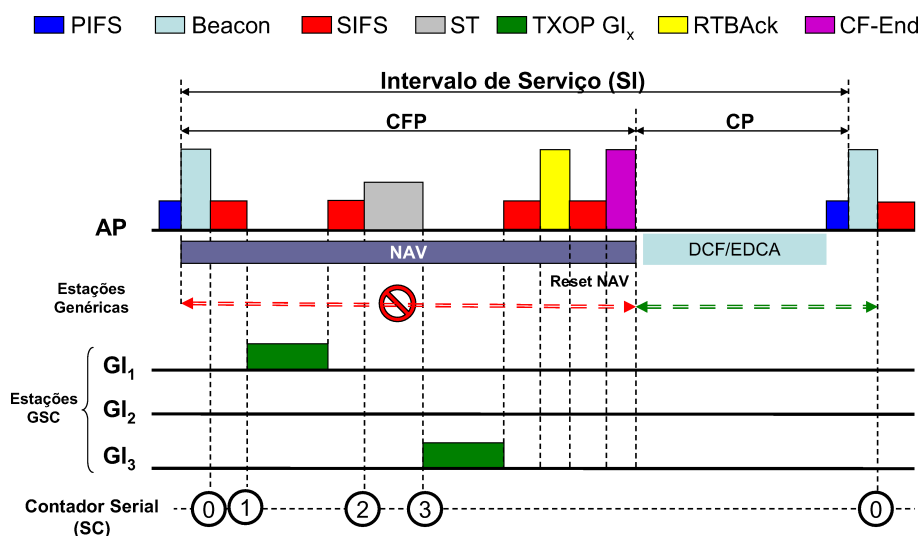
O mecanismo *GSC* utiliza o conceito de grupo de tempo real, onde os membros deste grupo têm acesso exclusivo ao meio no *CFP*. Ao contrário do *HCCA*, o *GSC* não utiliza o procedimento de interrogação (*Polling*) para atribuir (*TXOPs*) a cada estação *GSC* admitida no grupo. Para coordenar o acesso ao meio, o *GSC* utiliza o conceito de passagem de ficha virtual (*Virtual Token Passing - VTP*), onde cada estação *GSC* pode acessar o meio de transmissão sequencialmente (livre de colisões) e transmitir sua mensagem de tempo real em seu respectivo *TXOP* que foi negociado anteriormente no *CP* através do quadro de *TSPEC*. Desta forma, quando o escalonador do mecanismo *GSC* é ativado, o processo do *VTP* é iniciado para todas as estações de tempo real. Este procedimento executa a transferência de mensagens do grupo *GSC*, sem a necessidade de qualquer troca de quadros de controle entre o Coordenador Híbrido (*HC*) e as estações *GSC*.

O esquema *VTP* utiliza um conjunto de contadores locais, chamado de Contador Serial (*Serial Counter - SC*) que são implementados nas estações *GSC* e são incrementados basicamente em um período específico de *SIFS*. Cada Contador Serial (*SC*) implementa individualmente uma variável distribuída que define qual estação *GSC* com identificação do grupo (*Group Identification - GId*) terá garantida o seu acesso ao meio de transmissão, pois toda vez que o valor do *GId* for igual a *SC* ( $GId = SC$ ), a referida estação *GSC* estará apta a transmitir seu fluxo de dados contendo apenas uma mensagem de tempo real. A Figura 4.2 ilustra o comportamento básico do mecanismo de comunicação em tempo real *GSC* em função do tempo de transmissão.



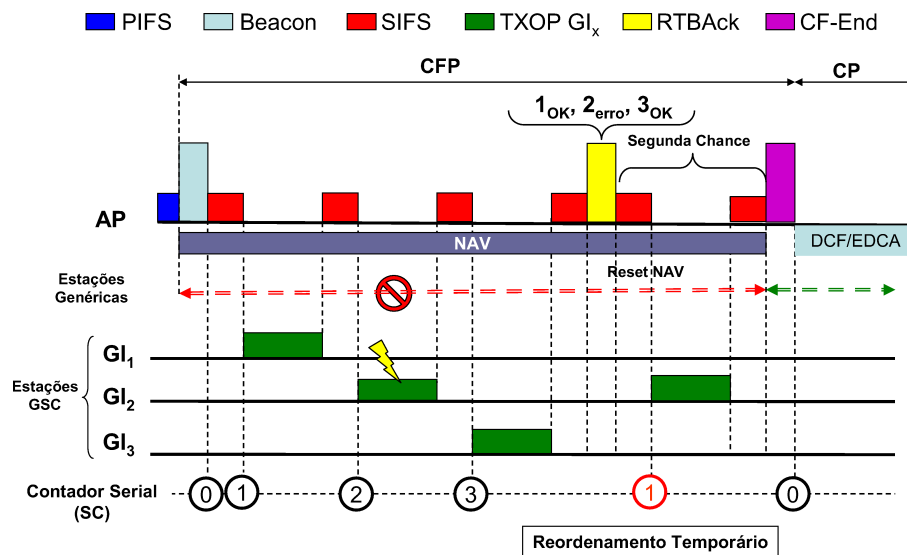
**Figura 4.2:** Funcionamento Básico do Mecanismo GSC

Contudo, sempre que uma estação GSC não tiver pacote de dados na fila de transmissão, essa estação vai transferir o direito de acesso ao meio à próxima estação do grupo de tempo real passado o período de  $aSlotTime$  (ST), minimizando assim a perda de largura de banda da rede; diferentemente do mecanismo HCCA que transmite uma mensagem nula (*QoS Null*) sempre que recebe um quadro de *Polling* do Coordenador Híbrido (HC) e não tem pacote de dados na fila de transmissão. A Figura 4.3 ilustra o comportamento temporal do mecanismo GSC quando uma estação não tem pacote a transmitir.



**Figura 4.3:** Economia de Banda do Mecanismo GSC

Ao final do processo de transmissão do grupo de estações de tempo real (estações GSC), todas as mensagens que foram transmitidas são confirmadas pelo HC através de um quadro de reconhecimento de bloco de tempo real (equivalente ao quadro *BlockAck* descrito na emenda IEEE 802.11e), chamado de *Real-Time Block Acknowledgment (RT-BlockAck)*. Se após este evento houver alguma mensagem que não foi recebida com sucesso, o processo de retransmissão dos pacotes perdidos recomeça imediatamente somente com as estações GSC que falharam. Este esquema de recuperação de erro é chamado de algoritmo de segunda chance (*Second Chance Algorithm*) e utiliza o mesmo procedimento VTP. A Figura 4.4 ilustra o comportamento temporal do mecanismo GSC quando um ruído destrói um pacote transmitido pela segunda estação GSC ( $GI_2$ ).



**Figura 4.4:** Esquema de Recuperação de Erro do Mecanismo GSC

O procedimento de inserção e remoção de estações GSC do grupo de tempo real é executado no período de contenção (CP) através de troca de mensagens padronizadas. Sua implementação é acionada através de dados adicionais transmitidos dentro do quadro de *Beacon*. Os dados são utilizados pelas estações GSC para obter as suas respectivas identificações ( $GI_d$ ) no grupo de tempo real. Entretanto para o mecanismo GSC começar a operar no CFP de um Intervalo de Serviço (SI), deve haver algumas condições iniciais no ambiente industrial, tais como:

- Não haver estações escondidas (GSC, Genéricas e AP) e
- Conceder às estações GSC uma oportunidade de transmissão (TXOP) destinada somente a um fluxo de mensagem de tempo real.

### 4.1.1 Especificação

O mecanismo *GSC* considera um Grupo Sequencial (*Sequence Group - SG*) com  $np$  membros onde circula uma ficha virtual de acesso ao meio (*Virtual Token Passing - VTP*) em  $L$ . As associações dos membros deste grupo são representadas como  $L = \{GI_1, GI_2, \dots, GI_{np}\}$ , onde  $GI_i$  é usado como identificação da estação (ID), e  $np$  é o número máximo de estações *GSC* admitidas no grupo de tempo real. Especificamente, todos os membros do *SG* devem manter um contador serial (*SC*) que é uma imagem da variável distribuída *SC*. O mecanismo *GSC* funciona da seguinte forma. No início do Intervalo de Serviço (*SI*), o *HC* transmite um quadro de *Beacon*. Este quadro cumpre basicamente as funções de:

- i) Atribuir valor de *NAV* para todas as estações genéricas, proibindo as suas transmissões até o fim do *CFP*;
- ii) Atribuir valor inicial para a variável distribuída *SC* igual a “0” ( $SC = 0$ ), tornando assim todas as estações *GSC* aptas ao processo de *VTP* e
- iii) Transmitir informações sobre os membros do grupo de tempo real (*SG*).

A primeira função impede todas as estações genéricas de iniciarem suas transmissões de mensagens durante o *CFP*. Isso garante que o mecanismo *GSC* funcione de forma a não ser perturbado. A segunda garante que no início do *CFP* todas as estações *GSC* têm o mesmo valor sincronizado de *SC*. A terceira será discutida em detalhes mais adiante na Subseção 4.1.3 sobre o quadro de *Beacon*.

Sempre que uma estação *GSC* de identificação  $GI_i$  tiver um pacote pronto para ser transmitido e sua imagem *SC* for igual ao seu ID ( $SC = GI_i$ ), então a estação estará autorizada para transmitir sua mensagem de tempo real imediatamente depois do meio tornar-se livre durante um intervalo igual a *SIFS*. De outra forma, se a estação de *GSC* estiver de posse da ficha virtual ( $SC = GI_i$ ) e não tiver nenhuma mensagem de tempo real a transmitir, o valor de *SC* será incrementado de uma unidade após *aSlotTime* (*ST*) em todas as estações *GSC*, permitindo que a próxima estação do grupo de tempo real ( $GI_{i+1}$ ) tenha a oportunidade de transmissão. Esta operação será repetida sucessivamente até que todas as estações *GSC* possam transmitir suas mensagens.

### Esquema de Recuperação de Erros (Algoritmo de Segunda Chance)

A proposta *GSC* implementa o paradigma Produtor/Consumidor, onde todas as mensagens de tempo real são transmitidas em *broadcast* e todas as estações interessadas nos



dados transmitidos podem consumi-los. Isto resulta numa redução de *overhead* na rede quando comparado com o mecanismo *HCCA*. Porém, o uso de abordagem Produtor/Consumidor para transmissão em redes sem fio pode conduzir a problemas, particularmente em ambientes ruidosos, pois se os quadros de reconhecimento (*ACK*) não são usados, não é possível confirmar se uma mensagem foi recebida corretamente, ou não. Se uma mensagem é corrompida devido a ruídos no canal de comunicação, então esta mensagem será rejeitada pelas estações receptoras. Porém, a estação transmissora não terá conhecimento desta situação.

Em termos práticos, somente no *CFP* seguinte seria possível transmitir outra vez a mesma mensagem, que é equivalente a um atraso indesejável (*delay*) na transmissão. Isto pode conduzir à perda de *deadline* e conseqüente degradação no desempenho do sistema. Assim, é necessário ter um esquema de recuperação de erro para assegurar-se de que uma mensagem de tempo real possa ser retransmitida com um atraso mínimo para não perder o *deadline*.

O *GSC* implementa o algoritmo de segunda chance por meio de um novo ciclo de *VTP* destinado à retransmissão de mensagens de tempo real que não obtiveram sucesso no ciclo inicial. Esse algoritmo funciona da seguinte forma. O *HC* monitora todas as mensagens de tempo real transmitidas e verifica se houve erros até o final do ciclo. Em seguida, após um intervalo *SIFS*, o *HC* transmite a taxa máxima da rede um quadro *broadcast* de reconhecimento (*RTBlockAck*) para confirmar todos os pacotes transmitidos. Este quadro é baseado no quadro de reconhecimento de bloco (*Block Acknowledgment*) definido no padrão IEEE 802.11e.

O quadro *RTBlockAck* tem um campo que contém um vetor de máscara de bits ( $mask(k)$ ) que indica para cada estação se as transmissões obtiveram sucesso (“1”) ou não (“0”), isto da perspectiva do *HC*. Ao receberem este quadro, as estações *GSC* podem analisar (verificando o bit correspondente da máscara) se a sua transmissão foi correta ou não.

De posse da informação do quadro *RTBlockAck*, o algoritmo de Segunda Chance funciona alterando temporariamente a identificação de cada estação de tempo real ( $GI_i$ ) para um novo ciclo (*VTP*) da seguinte forma: se uma estação transmitir uma mensagem e se o *HC* não detectar erros, então a referida estação é excluída temporariamente do grupo de tempo real. Isto pode ser feito pela seguinte configuração  $GI_{temp} = GI_i$ ,  $GI_i = 0$ , onde  $GI_{temp}$  é temporariamente igual a  $GI_i$ . Se uma estação teve sua transmissão afetada por erros, então a estação define o seu  $GI_i$  de acordo com a regra a seguir; a retransmissão dos quadros errados vai decorrer com um novo ciclo de *VTP*, da seguinte forma: a estação ao analisar a máscara, vai contar quantos bits “1” existem (ou seja transmissões corretas) entre o 1º bit da máscara e o bit correspondente à estação. Vai então, temporariamente,

atualizar o seu  $GI_i$  para:  $GI_{temp} = GI_i - \{\text{números de bits "1" contados à esquerda do vetor de máscara de bits}\}$ . Com esta operação cria-se, assim, uma segunda oportunidade de transmissão apenas para as estações que tiveram quadros corrompidos. Então, pode-se descrever o algoritmo desta forma:

- $GI_{temp} = GI_i$

$$GI_i = GI_i - \sum_{k=1}^i mask(k) \quad (4.1)$$

onde o vetor  $mask(k)$  representa o bit  $i$  no campo "*BlockAck Bitmap*" pertencente ao quadro *RTBlockAck*.

Ao aplicar esta regra, um novo grupo é temporariamente criado através de um reordenamento das estações. O grupo é composto apenas pelas estações que tiveram erros nas suas transmissões. Além disso, todas estas estações terão identificações consecutivas, o que minimiza o tempo necessário para completar o processo *VTP* de segunda chance. Portanto, este funciona como um esquema de recuperação de erro, o que minimiza as chances de perda ou atraso das mensagens de tempo real.

### Exemplo do Funcionamento do Algoritmo de Segunda Chance

Agora para ilustrar, utilizaremos o seguinte exemplo. Considere um ciclo de operação do mecanismo *GSC* que tem as seguintes estações:  $GI_1$ ,  $GI_2$ ,  $GI_3$ ,  $GI_4$  e  $GI_5$ . Nesse ciclo as estações  $GI_2$  e  $GI_4$  tiveram os quadros corrompidos por ruídos. Logo, o *HC* através do quadro *RTBlockAck* obtém a seguinte informação:  $mask(k) = (10101)$ . Ao receber o quadro as estações atualizam os seus  $GI_i$  da seguinte forma:

- A 1ª estação lê o 1º bit ( $mask(k)$ ) de valor "1" e confirma que o seu quadro teve sucesso na transmissão.
- A 2ª estação lê o 2º bit ( $mask(k)$ ) de valor "0" e confirma que o seu quadro foi corrompido na transmissão. Existe uma vez o bit "1" à esquerda, logo esta estação coloca o seu  $GI_{1temp} = 1$ .
- A 3ª estação lê o 3º bit ( $mask(k)$ ) de valor "1" e confirma que o seu quadro teve sucesso na transmissão.
- A 4ª estação lê o 4º bit ( $mask(k)$ ) de valor "0" e confirma que o seu quadro foi corrompido na transmissão. Existe duas vezes o bit "1" à esquerda, logo esta estação coloca o seu  $GI_{2temp} = 2$ .

- A 5ª estação lê o 5º bit ( $mask(k)$ ) de valor '1' e confirma que o seu quadro teve sucesso na transmissão.

No novo ciclo VTP as estações  $GSC$  ( $GI_2$  e  $GI_4$ ) vão passar a ter as seguintes identificações temporárias:  $GI_1$  {2},  $GI_2$  {4} (dentro de {chaves} estão os valores dos  $GI_i$  originais).

Após o procedimento de segunda chance, passado um período  $SIFS$ , o  $HC$  envia em *broadcast* um quadro de  $CF-End$  que assinala o fim do  $CFP$  e as estações voltam a utilizar as suas identificações  $GI_i$  originais.

O Quadro  $CF-End$  cumpre várias funções. Primeiramente, o quadro reinicia (*reset*) todos os valores de  $NAV$  das estações padrão  $DCF/EDCA$ . Assim essas estações podem começar a disputa ao meio para a transmissão de suas mensagens no  $CP$ . Logo em seguida, depois da transmissão de um novo quadro de *Beacon* todas as estações  $GSC$  recuperam suas identificações originais ( $GI_i$ ) do grupo de tempo real. Além dessas propriedades, há uma hipótese que especifica que o grupo lógico  $SG$  já foi anteriormente inicializado com  $np$  estações.

#### Descrição do Quadro $RTBlockAck$

O quadro  $RTBlockAck$  faz parte do esquema de recuperação de erro e tem formato similar ao  $BlockAck$  do IEEE 802.11e e é descrito da seguinte forma: 02 bytes para o quadro de controle (*Frame Control*), 02 bytes para a duração da transmissão (*Duration/ID*), 06 bytes para o endereço do receptor em *Broadcast* (*Receiver Address*), 06 bytes para o endereço do transmissor (*Sender Address*) que é o endereço  $MAC$  do  $HC$  (*MAC Address*), intervalo de 01 a 128 bytes para o mapa de bit (*BlockAck bitmap*), limite mínimo de 01 e máximo de 1.024 estações e 04 bytes para controle de erro (*Cyclic Redundancy Character - CRC*). O formato do quadro  $RTBlockAck$  é mostrado na Figura 4.5.

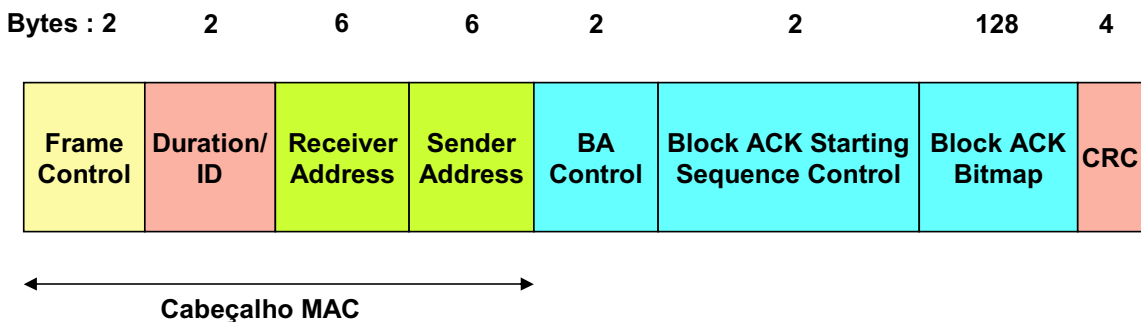
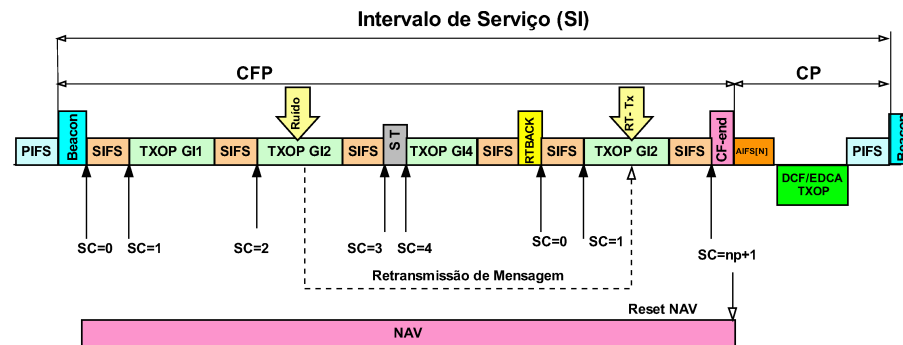


Figura 4.5: Formato do Quadro de *Block Acknowledgment*.

Obs: No campo que contém a máscara de bit (*BlockAck Bitmap*), o n-ésimo bit é usado para informar se a estação  $GI_{np}$  tem sua transmissão afetada por erros (atribui o valor “0”), ou não (atribui o valor “1”) e os campos formados por *BA Control* e *Block Ack Starting Sequence Control* não são utilizados.

### Exemplo de Funcionamento do Mecanismo GSC

A Figura 4.6 mostra um exemplo do funcionamento do mecanismo GSC. Neste cenário, o HC transmite um quadro de *Beacon* sincronizando os contadores seriais ( $SC=0$ ), iniciando o CFP. Logo após o meio ficar ocioso durante um intervalo *SIFS*, o contador serial da estação 1 tem o valor incrementado para  $SC = 1$ . Então, a estação  $GI_1$  começa a transmissão de sua mensagem de tempo real ( $SC = GI_1$ ).



**Figura 4.6:** Intervalo de Repetição do Mecanismo GSC.

Considerando que todas as estações receberam o quadro de *Beacon* e a estação  $GI_1$  começou sua transmissão com sucesso, será permitido a esta estação o controle de acesso ao meio de no máximo *TXOP*. No final do *TXOP*, após um período de *SIFS*, todas as estações GSC incrementarão seus valores dos contadores seriais ( $SC = SC + 1$ ) passando a ficha virtual ( $SC = 2$ ) para a próxima estação ( $GI_2$ )<sup>7</sup> que iniciará a sua transmissão, porém nesse momento a mensagem é corrompida por ruído no canal, por exemplo.

No entanto, o HC tomará uma providência para esta situação, e irá efetuar uma sinalização<sup>8</sup> através do uso do quadro *RTBlockAck*. Depois de um período *SIFS* ( $SC = 3$ ), a estação de  $GI_3$  deve iniciar sua transmissão. No entanto, considerando que a estação  $GI_3$  não tem nenhuma mensagem a ser transferida, o contador *SC* é incrementado depois de um *aSlotTime* (*ST*) ( $SC = 4$ ), e a estação  $GI_4$  estará autorizada a iniciar a sua transmissão.

<sup>7</sup>O fim da *TXOP* pode ser detectada por todas as estações, sempre que o meio estiver ocioso por um período de *SIFS*.

<sup>8</sup>Atribuir o valor “0” para bit referente à estação no campo *Block ACK Bitmap*.

Ao final do *TXOP* da estação  $GI_4$ , e depois de um período *SIFS*, o *HC* envia um pacote *broadcast* de confirmação (*RTBlockAck*) para todas as estações de tempo real. A estação de tempo real  $GI_2$  verifica que não obteve sucesso em sua transmissão e mudará o seu  $GI_2$  para  $GI_1$  ( $GI_{1temp} = 1$ ) e aguardará uma nova oportunidade para transmitir seus pacotes utilizando o algoritmo de segunda chance.

Após o término do novo ciclo de transmissão e passado um período de *SIFS*. O mecanismo *GSC* enviará um quadro de *CF-End* para liberar (*reset*) os valores dos *NAV's* de todas as estações. Isto permite a inicialização do período de contenção (*CP*), onde as estações genéricas utilizando o mecanismo *DCF/EDCA* devem começar a disputar o meio de transmissão. Nesse período novas estações *GSC* podem também se inscrever/desinscrever para um novo intervalo de serviço (*SI*).

A seguir, descrevem-se os algoritmos do Coordenador Híbrido embutido no *AP* (Listing 4.1) e da estação *GSC* (Listing 4.2), que foram implementados no simulador chamado *GSCSimu*:

---

**Listing 4.1:** Algoritmo Principal GSC

---

```

Input : SC, SI, NP, GIi ;
CP = SI - CFP;
function main()
{
  start wait_timer;
  while (wait_timer <= PIFS)
  {
    if (medium idle)
      restart wait_timer;
  }
  stop wait_timer;
  send beacon frame; // CFP Start, Set SC=0 and Set NAV
  create RTBlockAck[NP] = gsc(NP,CFP);
  counter = 0;
  for i=1 to NP
  {
    if (RTBlockAck[i] == 0)
      counter = counter + 1;
  }
  Wait SIFS;
  broadcast_blockack(RTBlockAck);
  gsc(counter, counter*(packet transmission time + SIFS));
  Wait SIFS;
  CF_End; // CFP End and Reset NAV
  start wait_timer; // CP Start

```

```

while (wait_timer <= CP - PIFS)
{
    listen medium; // join/leave GSC and EDCA/DCF stations
}
stop wait_timer; // CP End
}

function gsc(nstas ,duration) {
    create RTBlockAck[nstas];
    SC = 1;
    start cfp_timer;

    while (cfp_timer <= duration) {
        if (new event receveid) {
            if (successfull) {
                RTBlockAck[SC] = 1;
                SC = SC + 1;
            } else {
                RTBlockAck[SC] = 0;
                SC = SC + 1;
            }
        } else {
            Wait aSlotTime;
            RTBlockAck[SC] = 1;
            SC = SC + 1;
        }
    }
    return (RTBlockAck);
}

```

---

#### Listing 4.2: Algoritmo da Estação GSC

---

```

Input: Beacon ,RTBlockAck;
       CF_End,NP;
function main()
{
    Gli = join network()
    while (true) {
        Wait Beacon frame;
        success = false;
        TempRTBlockAck = cfploop(Gli ,NP, false);

        if (success == true) {

```

```

        Wait CF_End;
    } else {
        counter == 0;
        for i=1 to length(TempRTBlockAck) {
            if (TempRTBlockAck[i] == 0) {
                counter = counter + 1;
                if (GIi == i) {
                    new_GIi = counter;
                }
            }
        }
        cfploop(new_GIi , counter , true);
        Wait CF_End;
    }

    join or leave medium; // CP
    Wait PIFS;
}

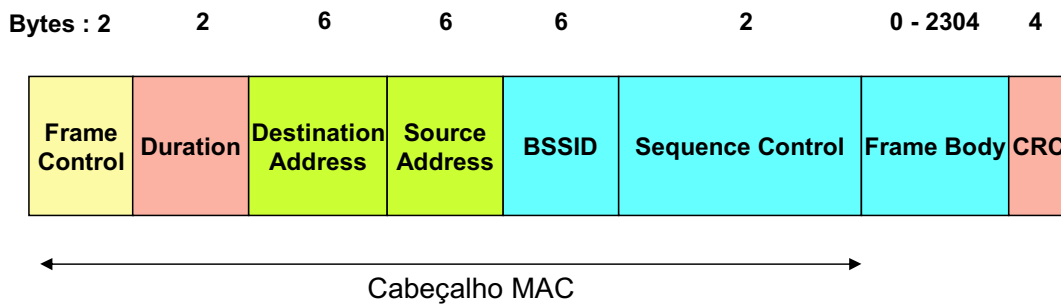
function cfploop(idx , npx , second_chance) {
    SC = 0;
    while (true) {
        Wait SIFS;
        SC = SC + 1;
        if (idx == SC) {
            Transmit RT Data Packet;
        } else if (SC == npx + 1) {
            if (second_chance == false) {
                Receive RTBlockAck;
                if (RTBlockAck[idx] == 1) {
                    success = true;
                }
            }
        }
        Break;
    } else if (medium busy) {
        Wait RT Transmission End;
    } else {
        Wait aSlotTime;
    }
}
return (RTBlockAck);
}

```

---

### 4.1.2 Quadro de *Beacon*

O *HC* transmite periodicamente quadros de *Beacon* às estações para a identificação da rede. Esses quadros têm um papel fundamental no funcionamento do mecanismo *GSC*, pois contêm o valor do relógio mantido pelo *HC* no momento da transmissão; uma vez sincronizados os relógios de todas as estações, pode-se garantir o bom funcionamento da rede. Na Figura 4.7 é apresentado o formato do quadro de *Beacon*.



**Figura 4.7:** Formato do Quadro de *Beacon*.

O formato desse quadro de gerenciamento é independente do subtipo do quadro. De acordo com o padrão IEEE 802.11e, é obrigatório apenas que o corpo do quadro de gerenciamento do subtipo *Beacon* contenha as informações apresentadas na Tabela 4.1.

**Tabela 4.1:** Corpo do Quadro de *Beacon*.

Order	Information
1	Timestamp
2	Beacon interval
3	Capability information
4	SSID
5	Supported rates
6	FH Parameter Set
7	DS Parameter Set
8	CF Parameter Set
9	IBSS Parameter Set
10	TIM
11	QBSS Load
12	EDCA Parameter
13	QoS Capability



O *GSC* utiliza dois campos no quadro de *Beacon* para transmitir suas informações: Informação de Capacidade (*Capability Information*) e o Mapa de Informação de Tráfego (*Traffic Information Map - TIM*). É importante notar que este é um quadro padrão (IEEE 802.11) e não uma modificação do mesmo.

O campo Capacidade de Informação é composto por vários sub-campos que são usados para anunciar as capacidades do *HC*. Os sub-campos *CF-Pollable* e *CF-Poll Request* são definidos a fim de informar às estações que o *HC* não suporta o mecanismo de *Polling HCCA*. Conseqüentemente, as estações genéricas (*DCF/EDCA*) não vão solicitar suporte para operar com este mecanismo.

Campo *TIM (Traffic Information Map)* também tem vários sub-campos. O sub-campo Mapa de Bit Virtual Parcial (*Partial Virtual Bitmap - PVB*) é usado pelo *HC* para enviar informações sobre associações ativas com as estações. Este sub-campo é uma máscara de bit, onde o bit  $i$  corresponde à identificação da associação (*Association ID<sub>i</sub>*). Se o bit  $i$  é definido com o valor igual a "1", então o *HC* tem pacotes no *buffer* armazenados para a estação *ID<sub>i</sub>*. Caso contrário, não há pacotes e a estação não está associada a rede.

O *GSC* usa o *PVB* da seguinte maneira. A identificação de associação *ID* de 1 a 64 está reservada para as estações do *GSC* (este valor limite pode ser ajustado se necessário), e os restantes (65 até 2008) são reservados para as estações genéricas. As estações genéricas interpretam o sub-campo *PVB* de acordo com as regras definidas pela norma IEEE 802.11. No entanto, para estações de *GSC* o sub-campo *PVB* tem uma semântica diferente: em cada período de contenção (*CP*) as estações *GSC* podem entrar ou sair do grupo de tempo real (discutido mais adiante). O *HC* leva em conta estas situações e usa o sub-campo *PVB* para informar a todas estações *GSC* sobre isto. Se a estação *GI<sub>i</sub>* é removida do grupo de tempo real, então o bit  $i$  do *PVB* é redefinido para o valor "0", caso contrário, é definido como "1". A remoção ou inserção de estações para o grupo de tempo real exige que o *GI<sub>i</sub>* para cada estação deva ser recalculado. Isto pode ser feito se cada estação *GSC* aplicar a regra a seguir depois de receber um quadro de *Beacon*:

$$GI_i = GI_i - \sum_{k=1}^i NOT(PVB(k)) \quad (4.2)$$

Onde ( $PVB(k)$ ) representa o bit  $i$  do sub-campo *PVB*. Portanto, o quadro de *Beacon* é utilizado para transmitir informações atualizadas sobre os atuais membros do grupo de tempo real, o qual é usado pelas estações *GSC* para atualizar a sua identificação no grupo.

### Exemplo do Funcionamento do Quadro de *Beacon*

Agora para ilustrar o funcionamento do quadro de *Beacon*, utilizaremos o seguinte exemplo. Considere um ciclo de operação do mecanismo *GSC* que tem as seguintes estações:  $GI_1$ ,  $GI_2$ ,  $GI_3$ ,  $GI_4$  e  $GI_5$ . Nesse ciclo as estações  $GI_2$  e  $GI_4$  se desassociaram do grupo de tempo real. Todas as estações através do quadro de *Beacon* obtém a seguinte informação:  $PVB(k) = (10101)$ . Ao receber o quadro as estações atualizam os seus ID's ( $GI_i$ ) da seguinte forma:

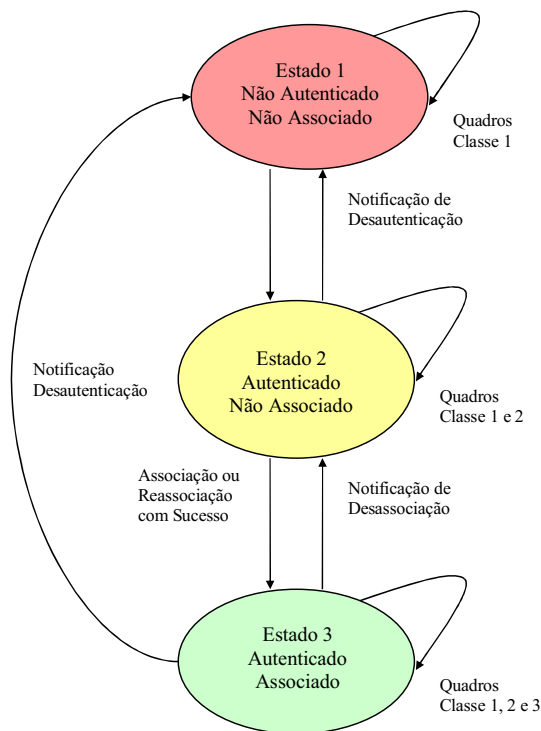
- A 1ª estação ( $GI_1$ ) lê o 1º bit de valor “1” em  $PVB(k)$  e confirma que o seu ID faz parte do grupo de tempo real. O próximo passo é recalcular o seu ID ( $GI_i$ ). A partir da posição do seu bit (1º bit) de identificação, subtrair o total de bits “1” a esquerda de  $NOT(PVB(k)) = (01010)$ , como não há nenhum, então o seu ID é  $GI_1$  (sem mudanças). A estação vai operar normalmente no próximo ciclo *GSC*.
- A 2ª estação ( $GI_2$ ) lê o 2º bit de valor “0” em  $PVB(k)$  e confirma que o seu ID não faz parte do grupo de tempo real.
- A 3ª estação ( $GI_3$ ) lê o 3º bit de valor “1” em  $PVB(k)$  e confirma que o seu ID faz parte da grupo de tempo real. O próximo passo é recalcular o seu ID ( $GI_i$ ). A partir da posição do seu bit (3º bit) de identificação subtrair o total de bits “1” a esquerda de  $NOT(PVB(k)) = (01010)$ , como existe apenas um bit “1”, logo é necessário atualizar o seu ID para  $GI_2$ . A estação vai operar normalmente no próximo ciclo *GSC* sem desperdício de banda da rede.
- A 4ª estação ( $GI_4$ ) lê o 4º bit de valor “0” em  $PVB(k)$  e confirma que o seu ID não faz parte do grupo de tempo real.
- A 5ª estação ( $GI_5$ ) lê o 5º bit de valor “1” em  $PVB(k)$  e confirma que o seu ID faz parte da grupo de tempo real. O próximo passo é recalcular o seu ID ( $GI_i$ ). A partir da posição do seu bit (5º bit) de identificação subtrair o total de números “1” a esquerda de  $NOT(PVB(k)) = (01010)$ , como existe duas vezes o bit “1”, logo é necessário atualizar o seu ID para  $GI_3$ . A estação vai operar normalmente no próximo ciclo *GSC* sem desperdício de banda da rede.

No novo ciclo VTP as estações *GSC* vão passar a ter seguintes atualização de identificação:  $GI_1 \{1\}$ ,  $GI_2 \{3\}$  e  $GI_3 \{5\}$  (dentro de {chaves} estão os valores dos  $GI_i$  anteriores).

Após o procedimento de atualização dos ID's, o *HC* envia em *broadcast* um novo quadro de *Beacon* que assinala o começo do *CFP* ( $SC=0$ ) e passado um período *SIFS* o contador serial (*SC*) é incrementado para  $SC=1$ . Desta forma, as estações utilizam as suas atuais identificações  $GI_d$ 's para iniciar um novo ciclo de transmissões de mensagens de tempo real.

### 4.1.3 Inserção de Estações

Uma estação pode entrar no grupo de tempo real, solicitando a sua admissão ao coordenador híbrido. Este pedido é realizado durante o período de contenção e, consequentemente, não perturba o funcionamento do *GSC*. Para obter conectividade com a rede, uma estação sem fio deve estabelecer uma relação com um ponto de acesso, chamada associação. Uma associação completa com um ponto de acesso envolve a transição por três estados: 1) Não autenticado e não associado; 2) Autenticado e não associado e; 3) Autenticado e associado. Os quadros do Padrão IEEE 802.11 podem ser de dois tipos: gerenciamento e dados, para transitar entre os estados 1 e 2. O *AP* e a estação sem fio trocam quadros de gerenciamento para autenticação. Na Figura 4.8 ilustra-se o esquema via máquina de estado:



**Figura 4.8:** Máquina de Estados do IEEE 802.11.

Uma estação de *GSC* interessada em se tornar um membro do grupo de tempo real deve enviar para o *HC* um quadro de *Association-Request frame* (Tabela 4.2). O campo *Capability Information* deste quadro (que tem características semelhantes ao campo do quadro de *Beacon* com o mesmo nome) é definido como interrogável (*Pollable*), afirmando que a estação quer ser membro do grupo de tempo real.

**Tabela 4.2:** Corpo do Quadro de Requisição de Associação.

Order	Information
1	Capability information
2	Listen interval
3	SSID
4	Supported rates
9	QoS Capability

As estações genéricas seguem as capacidades do *HC* (enviada no quadro de *Beacon*) que define este campo como não-interrogável (*No-Pollable*). Portanto, este artifício é usado para permitir a distinção entre as estações *GSC* e genéricas. Assim, apenas estações *GSC* podem entrar no grupo de tempo real. O *HC* responde para a estação com seus respectivos *Association-ID*. Essa identificação será correspondente à estação  $GI_i$ . A estação que não é admitida no grupo recebe um “0” como resposta. O gerenciamento das identificações (ID) das estações é, portanto, realizado pelo *HC*.

Depois de receber sua identificação ( $GI_i$ ), a estação deve enviar para o *HC* uma especificação de tráfego através do quadro *TSPEC*. Este quadro contém um conjunto de parâmetros que define as características de *QoS* do tráfego da estação. O *HC* usa esses dados para uma análise de escalabilidade do estado que uma estação pode, ou não, ser admitido no grupo de tempo real. Esta resposta é enviada para a estação através de um *TSPEC-Response frame*. Em caso afirmativo, a estação é agora membro do grupo de tempo real e pode trocar dados no *CFP* seguinte. No próximo quadro de *Beacon* o bit correspondente do *Association-ID* para esta estação será definido com o valor “1”. Caso contrário, a estação está fora do grupo de tempo real. Neste caso, o *HC* abandona a *Association-ID* anteriormente atribuída a esta estação.

#### 4.1.4 Remoção de Estações

Há duas formas distintas de se remover uma estação do mecanismo *GSC*, são elas: forma controlada e não-controlada, respectivamente.

- De forma controlada, a estação *GSC* decide autonomamente deixar o grupo. Neste caso, a estação deve comunicar a sua decisão para o *HC* através de uma mensagem de Desassociação (*Disassociation Request*) para remover-se do grupo durante o período de contenção (*CP*). Se o pedido for aceite, o *HC* abandona a *Association-ID* correspondente e envia uma resposta positiva de confirmação para a estação. Depois de receber a resposta, a estação considera que não pertencem mais ao grupo de tempo real. Na próxima transmissão do quadro de *Beacon* o bit correspondente ao *Associação-ID* para esta estação será reiniciado com o valor "0". Se a resposta for negativa, a estação permanece dentro do grupo de tempo real.
- De forma não-controlada, acontece quando uma estação parar de operar de uma forma repentina (por exemplo por falha de *hardware/software*). Como efeito imediato, não haverá mais mensagens de tempo real no *TXOP* atribuído a esta estação. Para lidar com esta situação, o *HC* usa uma mensagem de monitoramento (*heartbeat*) para verificar se a estação está "viva" na rede, ou não. Esta mensagem é enviada, somente se após um número pré-definido de *CFP's* nenhuma mensagem desta estação for detectada pelo *HC*.

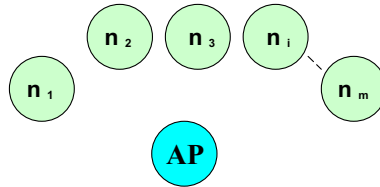
A mensagem de monitoramento é enviada no período de contenção como um pacote de dados, onde o campo *Frame Body* está vazio. Se a estação estiver "viva" e receber a mensagem, ela deve responder com um quadro de *ACK* e desta forma o *HC* considera que a referida estação ainda pertence ao grupo de tempo real. Se nenhuma resposta é recebida dentro de um período de tempo (*ACK Timeout*), o *HC* considera que a estação não pertence mais ao grupo de tempo real, pois a estação encontra-se fora de operação (*Off-Line*).

Assim, na próxima transmissão do quadro de *Beacon* o bit correspondente a identificação de associação *Association-ID* para esta estação será reiniciado com o valor "0". Nesta situação, o endereço utilizado  $GI_d$  será mais adiante recuperado pela última estação admitida para evitar o desperdício de largura de banda no *CFP*.

## 4.2 Modelagem do Sistema

Dada uma topologia de rede com  $m$  estações (nós) mais um ponto de acesso (*AP*) conectados através de um meio de comunicação sem fio de acordo com o mecanismo *GSC*. As estações  $n$  são identificadas pelo índice de  $i$ , do nó 1, nó 2,...nó  $i$ , até nó  $m$ . Considera-se também que na rede não há estações escondidas entre o *AP* e as estações. Ou seja, o *AP* é capaz de detectar o sinal de transmissão de todas as estações, e todas as

estações são capazes de detectar as mensagens uma das outras e as geradas pelo AP. Esta Topologia de rede é mostrada na Figura 4.9.



**Figura 4.9:** Topologia de Rede sem Fio.

Como descrito anteriormente, o mecanismo *GSC* tem um comportamento cíclico. Esse comportamento é reforçado pelo AP, que gera periodicamente um quadro de *Beacon*. A periodicidade do *Beacon* é indicada como o Intervalo de Serviço (*SI*). O *SI* é composto pelo período livre de contenção (*CFP*) para o tráfego de estações de tempo real mais o período de contenção (*CP*) para a transmissão de estações genéricas (*DCF/EDCA*) que tem baixa prioridade na rede.

### 4.2.1 Modelagem do Tráfego

Mensagens geradas no sistema em tempo de execução podem ser classificadas como *mensagens síncronas* ou *mensagens assíncronas*. Supomos que existem  $n$  fluxos de *mensagens síncronas*,  $S_1, \dots, S_2, S_n$  no sistema, que formam um conjunto de *mensagem síncronas*  $M$ , dado por:

$$M = \{S_1, S_2, \dots, S_n\} \quad (4.3)$$

As características dos fluxos de mensagens são definidas como:

$$S_i = \{C_i, P_i, d_i\} \quad (4.4)$$

Onde:

1.  $P_i$  descreve o tempo de chegada do fluxo (*inter-arrival time*) de *mensagem síncronas*  $S_i$  ( $i = 1, 2, \dots, n$ ). Mensagens síncronas são definidas como mensagens periódicas, isto é, são geradas com um tempo de chegada dos pacotes constante.
2.  $d_i$  representa o *deadline* de uma *mensagem síncrona* que é alcançado no final do seu período, ou seja, se uma mensagem do fluxo de mensagem  $S_i$  é gerado no instante  $t$ , então o seu *deadline* é dado por:  $(t + P_i)$ .

3.  $C_i$  é o *comprimento* de cada fluxo de mensagem  $S_i$ , que é a quantidade máxima de tempo necessária para transferir a mensagem, isto é,  $C_i$  inclui o pacote de dados, *overheads* da camada física e *MAC*, como também o intervalo de tempo durante o qual o meio não pode ainda ser usado para transferir a próxima mensagem, ou seja, o espaço mínimo entre quadros (*SIFS*).
4. As mensagens são *independentes*, logo as chegadas de mensagens não dependem do início ou a conclusão dos pedidos de transmissão de outras requisições.
5. As mensagens assíncronas não são periódicas e logo não possuem requisitos de tempo real.

Na seguinte discussão, assumimos que:

- Há somente um fluxo de mensagem síncrona por estação de tempo real  $1, 2, \dots, i, \dots, n$ , porém podem existir um ou mais fluxos de mensagens assíncronas por estações genérica nas estações  $n + 1, n + 2, \dots, m$ .
- As requisições de tempo real de um fluxo de mensagem síncrona  $S_i$  podem ser garantidas na rede.

No que diz respeito ao primeiro pressuposto, é fácil mostrar que uma rede *GSC* arbitrária, quando uma estação pode ter zero ou mais fluxos de mensagem síncrona, pode ser transformada em uma rede logicamente equivalente a um fluxo de mensagens por estação. Assim, a hipótese de um fluxo de mensagem síncrona por estação simplifica a análise, sem perda de generalidade.

Relativamente à cobertura das hipóteses acima, isto será avaliado na seção "Análise de Erros do Mecanismo *GSC*". Resumidamente, esta segunda hipótese será verdadeira, desde que:

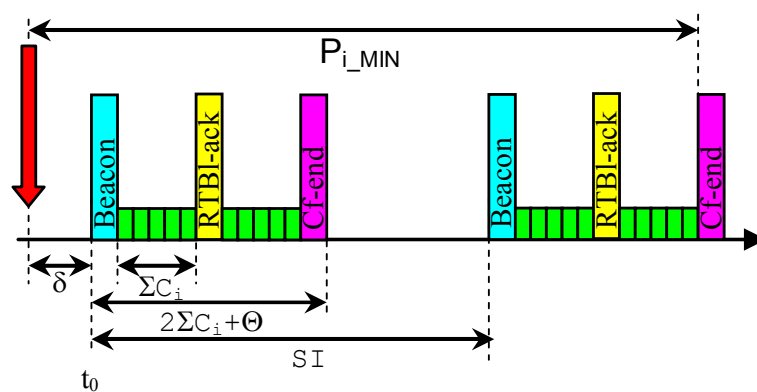
- Não haja perdas do quadro de *Beacon*;
- Não haja perdas do quadro *RTBlockAck* e,
- Ter a possibilidades de haver dois erros consecutivos (perda de pacote) utilizando o procedimento *VTP* e o algoritmo de segunda chance, após a transferência de um fluxo de mensagem síncrona oriundo da mesma estação *GSC*.

### 4.2.2 Controle de Admissão

O mecanismo *GSC* utiliza o *TSPEC* como definido no padrão IEEE 802.11e. O *TSPEC* é enviado através de um quadro de gerenciamento que contém informações sobre as características e expectativas de *QoS* de um fluxo de tráfego (*TS*), que inclui os seguintes parâmetros: *Mean Data Rate* ( $R_i$ ), *Nominal Service Data Unit (MSDU) size* ( $N_i$ ), *Nominal PHY Rate* ( $P_r$ ), *Service Interval* ( $SI$ ) and *Maximum Service Interval* ( $MSI$ ). Como mencionado anteriormente, à semelhança do *HCCA*, o *GSC* define um Intervalo de Serviço ( $SI$ ) como soma de dois períodos distintos descrito por:  $SI = CFP + CP$ .

Como a definição do valor de  $SI$  desempenha um papel importante na garantia dos *deadlines* das mensagens síncronas, nesta subseção discute-se uma metodologia para a atribuição de valor para  $SI$  que garanta o cumprimento dos prazos (*deadlines*), enquanto existir uma hipótese de erro no canal de comunicação. A orientação principal é que os parâmetros das mensagens síncronas ( $C_i$ 's e  $P_i$ 's) de todas estações (nós) são fatores que orientam a determinação do valor de  $SI$  para o *AP*.

O escalonamento e o controle de admissão de um novo fluxo de mensagem é executado pelo *HC*, onde é calculado pelo escalonador o valor do  $SI$  e  $C_i$  (duração do *TXOP* mais *SIFS*). Os valores dos parâmetros  $C_i$  e  $P_i$  são atribuídos pelas aplicações suportadas. Em vez disso, o valor de  $SI$  deve ser determinado no momento de inicialização da rede (que também pode ser atualizada *on-line*) e deve ser capaz de garantir os requisitos temporais do conjunto de mensagens síncronas  $M$ . Desta forma cumprindo os *deadlines* estabelecidos. Isto é ilustrado na Figura 4.10.



**Figura 4.10:** Avaliação do Limite Superior/Inferior para o Intervalo de Serviço.

Qualquer que seja o esquema de atribuição do valor de  $SI$ , este deve satisfazer duas restrições:



- a) A restrição do mecanismo de transmissão que garante o comportamento correto do *GSC* de forma cíclica, com a periodicidade adequada ao quadro de *Beacon* e ;
- b) A restrição de *deadline* a fim de garantir que as mensagens em tempo real, são transmitidas antes que esgotem os seus prazos temporais.

Teoricamente, o tempo total disponível para transmitir mensagens síncronas durante um ciclo *GSC* pode ser no máximo igual ao *SI*. No entanto, na prática fatores como a retransmissão de mensagens, transferência de mensagens de controle e etc., reduzem o tempo total disponível para transferência de mensagens síncrona.

Assim, descrevemos o *overhead* do *SI* chamado  $\Theta$  como a quantidade máxima de tempo necessária para transmissão do quadro de *Beacon*, *RTBlockAck* e quadro de *CF-End* incluindo os cabeçalhos da camadas *PHY* e *MAC* mais os intervalos de tempo durante o qual o meio não pode ser usado para transferir qualquer outra mensagem (*SIFS*).

Além disso, uma mensagem para ser transferida em um determinado ciclo (*SI*) deve ter sido gerada um pouco antes do início do ciclo do *GSC*. Denotamos o valor mínimo deste intervalo de tempo como  $\delta$ .

Portanto, uma restrição do mecanismo *GSC* é que: o tempo gasto para transferir duas vezes uma mensagem de cada fluxo de mensagens mais o *overhead* do *SI* não deve ser superior ao valor do parâmetro de *SI*, ou seja:

$$2 \sum_{i=1}^n C_i + \Theta < SI \quad (4.5)$$

Além disso, considerando que uma mensagem gerada após o instante de tempo ( $t_0 - \delta$ ) pode não ser capaz de ser transferida, no ciclo atual (a única que inicia no momento  $t_0$ ), uma restrição de prazo é que: o tempo mínimo de chegada dos pacotes de qualquer fluxo de mensagem síncrona não deve ser menor do que o ciclo *GSC* mais o tempo gasto para transferir duas vezes uma mensagem de cada fluxo de mensagens mais o *overheads* do *SI*, ou seja:

$$P_i \geq \delta + SI + 2 \sum_{i=1}^n C_i + \Theta, \forall i=1 \dots n \quad (4.6)$$

Assim, podemos dizer que um conjunto de mensagens é garantido para o *SI* pré-definido se ambos o mecanismo e as restrições de prazo (*deadline*) são satisfeitos, enquanto há uma hipótese de erro. Essas duas restrições podem ser combinadas em uma única equação como segue:

$$2 \sum_{i=1}^n C_i + \Theta < SI \leq P_{i\_min} - \delta - 2 \sum_{i=1}^n C_i - \Theta \quad (4.7)$$

onde  $P_{i\_min}$  é o tempo mínimo de chegada de todos os fluxos de mensagens síncronas para o canal de comunicação.

Ao considerar cenários de tempo real, é interessante avaliar o pior caso para análise temporal, que deve demonstrar que o  $SI$  cumpre os *deadlines*. De forma semelhante ao mecanismo *HCCA*, a proposta *GSC* também permite que estações genéricas (*DCF/EDCA*) disputem o acesso ao meio durante o intervalo de *CP*. O pior cenário ocorrerá quando o conjunto de estações genéricas não recebem o quadro de *Beacon*<sup>9</sup> e antes do final do período de contenção, em que uma estação padrão adquire o meio de transmissão e usa todo o tempo permitido ao seu *TXOP* causando um atraso no início da próximo *SI*. Portanto, o tempo de transmissão de pior caso (intervalo de serviço máximo - *MSI*) é dado por:

$$MSI = SI + TXOP_{max} \quad (4.8)$$

Onde  $TXOP_{max}$  é definido como o máximo valor atribuído à *TXOP* para um conjunto de estações padrão EDCA [IEEE 2005].

### Exemplo de um Esquema Prático de Controle de Admissão e Escalonamento

O escalonamento<sup>10</sup> e o controle de admissão de um novo fluxo de tráfego são realizados pelo *HC*, onde é calculado a duração dos *TXOP* e o intervalo de serviço (*SI*). O *TXOP* pode ser definido como o tempo de transmissão nominal da mensagem de tempo real, incluindo camada física (*T<sub>phy</sub>*), camada MAC (*T<sub>mac</sub>*) e o tempo de transmissão da mensagens (*T<sub>Ni</sub>*), como descrito abaixo:

$$TXOP = T_{phy} + T_{mac} + TNi \quad (4.9)$$

É interessante notar que os *TXOPs* são calculados para que cada estação *GSC* transmita apenas um fluxo de mensagens. Se necessário em caso de perda de todos os pacotes, todas as estações possam transmitir suas mensagens novamente utilizando o algoritmo de segunda chance. Considera-se agora que o *CFP* pode ser representado por um percentual do intervalo de serviço (*SI*). Então, pode-se definir *CFP* de duas formas:

$$CFP = SI - CP \quad (4.10)$$

<sup>9</sup>Em tal caso, para o conjunto de estações genéricas não são atribuídos os valores de *NAV*.

<sup>10</sup>Pode-se utilizar um escalonador padrão *Round-Robin* equivalente ao *HCCA*.

$$CFP \leq \alpha * SI \quad (4.11)$$

Onde ( $\alpha$ ) é uma constante que pode variar de zero (0) a um (1), equivalente a 0% a 100% do valor de  $SI$ , respectivamente.

Por definição o mecanismo  $GSC$  tem  $CFP$  máximo igual ao dobro da somatória de todas as  $TXOPs$  atribuídas aos membros do grupo  $np$  de tempo real  $SG$  separados por intervalos de  $SIFS$  utilizando um canal com ruídos. Então:

$$CFP_{max} = 2 \sum_{i=1}^{np} (TXOP(i) + SIFS) + \Theta \quad (4.12)$$

onde  $\Theta$  é o tempo de transferência do *overhead* composto pelo quadro de *Beacon*, *RTBlockAck* e *CF-End* incluindo  $SIFS$ , camadas físicas e MACs.

Entretanto, considera-se que pode ser admitida uma nova estação ( $np + 1$ ) com um novo fluxo de mensagem se satisfazer a seguinte inequação:

$$2 \sum_{i=1}^{np+1} (TXOP(i) + SIFS) + \Theta \leq CFP \quad (4.13)$$

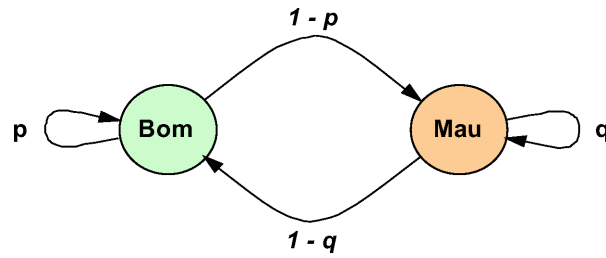
Da equação (4.11) e (4.12), é possível obter a seguinte relação:

$$SI \geq \frac{2 \sum_{i=1}^{np} (TXOP(i) + SIFS) + \Theta}{\alpha} \quad (4.14)$$

Utilizando-se, por exemplo, deste esquema, pode-se atribuir a determinado intervalo de serviço que apenas 10% do total seja usado pelas estações  $GSC$  ( $CFP$ ), bastando para isso atribuir o valor  $\alpha=0,1$ . Desta forma o período de contenção (CP) terá 90% do  $SI$  que será utilizado pelas estações genéricas ( $DCF/EDCA$ ).

### 4.2.3 Modelo de Erro do Canal

Considera-se que as mensagens geradas no sistema podem ser perdidas de acordo com o modelo de Gilbert-Elliot [Ebert et al. 1999], onde um canal de comunicação sem fio é representado por uma cadeia de *Markov* de dois estados: “Bom” chamado estado bom e “Mau” chamado de estado mau. Este comportamento é ilustrado na Figura 4.11. Onde,  $p$  é a probabilidade de permanecer no estado bom e  $q$  a de permanecer no estado mau.



**Figura 4.11:** Modelo do Canal de *Gilbert-Elliot* Representado por Cadeia de *Markov*.

Se o canal de comunicação estiver em “Bom” (estado bom), a taxa de erro de bit (*Bit Error Rate - BER*) na mensagem transmitida pode ocorrer com a probabilidade de  $e_B$ . Caso contrário, se o canal de comunicação está em “Mau” (estado mau), a probabilidade de erro na transmissão da mensagem é  $e_M$ . Destacando-se a seguinte relação  $e_B \ll e_M$ . Antes da transmissão de cada bit, o estado do canal de comunicação pode mudar para um novo estado ou permanecer no estado atual. Segundo Willig et al. (2001), o próximo estado do canal é determinado a cada bit transmitido através da matriz de transição descrita pela Equação 4.15.

$$\mathbf{P} = \begin{pmatrix} P_B & 1 - P_B \\ 1 - P_M & P_M \end{pmatrix} \quad (4.15)$$

Com base nessa assertiva, podemos obter as seguintes probabilidades para ambos os estados estacionários através da propriedade da cadeia de *Markov* [Ebert et al. 1999]:

$$P_B = \frac{1 - q}{2 - (p + q)} \quad (4.16)$$

$$P_M = \frac{1 - p}{2 - (p + q)} \quad (4.17)$$

Onde  $P_B$  e  $P_M$  são as probabilidades de estado estável para o estado bom “Bom” e o estado mau “Mau”.

A fim de simplificar usamos  $e_B = 0$  e  $e_M = 1$ . Então, para este modelo, a taxa de erro de pacote (*Packet Error Rate - PER*) de  $n$ -bytes do tamanho da mensagem é descrita por:

$$PER(n) = 1 - (P_B \cdot p^{8n} + P_M(1 - q) \cdot p^{8n-1}) \quad (4.18)$$

Pode-se observar que há duas maneiras em que não ocorre erro de bit na transmissão de um pacote:

- O primeiro caso, o canal de comunicação está em “Bom” (estado bom) e não há mudanças para todos os bits na transmissão do pacote; esta situação de probabilidade é descrita por  $P_B \cdot p^{8n}$  para todos os  $n$  bytes no pacote.
- O segundo caso, o canal de comunicação está em “Mau” (estado mau), mas há mudanças para o estado bom na transmissão do primeiro bit e permanece assim. Esta probabilidade é descrita por  $P_M(1 - q) \cdot p^{8n-1}$ .

Neste caso, um erro de pacote ocorre se somente se, nenhum dos dois eventos ocorrer. Desta forma o PER é obtida a partir do BER e do número de bits que forma o pacote. Inicialmente, o modelo de Gilbert-Elliot [Ebert et al. 1999] foi proposto para tratar a perda de bit em um canal de comunicação, mas atualmente o modelo é usado para tratar a perda de pacotes. Diferença agora é que no processo de transmissão que consiste em estados bom ou mau refere-se agora aos pacotes recebidos com sucesso ou perdidos, respectivamente.

#### 4.2.4 Análise de Erros

Segundo Willig et al. (2001), considera-se que a taxa de erro de pacotes (*Packet Error Rate - PER*) é independente entre pacotes sucessivos e pode-se obtê-lo através da adequação do modelo de Gilbert-Elliot [Ebert et al. 1999], como descrito na secção anterior. Com base nesta assertiva, calculamos o tempo médio de duração do *CFP* utilizado pelas estações *GSC* que têm um fluxo de mensagens de tempo real para transmitir nos seus respectivos *TXOPs*.

O *TXOP* é definido como o tempo total de transmissão para um fluxo de mensagem de tempo real, incluindo cabeçalhos da camada física ( $PhyHeader$ ) transmitidos à taxa básica ( $Pb$ ), cabeçalhos da camada *MAC* ( $MACHeader$ ) e mensagens de tempo real (*Data*) transmitidos à taxa nominal de transmissão ( $Pr$ ). Então, o *TXOP* pode ser descrito como:

$$TXOP = PhyHeader/Pb + (MACHeader + Data)/Pr \quad (4.19)$$

O  $C_i$  foi definido anteriormente como a quantidade máxima de tempo necessária que cada estação *GSC* tem para transferir um fluxo de mensagem de tempo real mais *overheads* da camada física e *MAC*, juntamente com o espaçamento mínimo entre quadros *SIFS* (Equação 4.20). Se necessário, todas as estações podem transmitir suas mensagens novamente usando o algoritmo de segunda chance.

$$C_i = TXOP + SIFS \quad (4.20)$$

Considera-se que o tempo de transmissão para o quadro de *Beacon*, *RTBlockAck* e *CF-End* são representados respectivamente por:

$$C_{beacon} = PhyHeader/Pb + Beacon/Pr + SIFS \quad (4.21)$$

$$C_{rtblockack} = PhyHeader/Pb + RTBlockAck/Pr + SIFS \quad (4.22)$$

$$C_{cfend} = PhyHeader/Pb + CFEnd/Pr \quad (4.23)$$

Onde o *overhead* do *SI* é descrito como o tempo máximo para a transferência dos quadros *Beacon*, *RTBlockAck* e *CF-End*, incluindo os seus cabeçalhos da camada física *PHY* e *MAC* mais os valores de *SIFS*, respectivamente. Isto é denotado por  $\Theta$ .

$$\Theta = C_{beacon} + C_{rtblockack} + C_{cfend} \quad (4.24)$$

O tempo de transmissão de passagem da ficha virtual quando uma estação *GSC* não tem pacote na fila a ser transmitido é descrito por *aSlotTime*:

$$C_{st} = aSlotTime \quad (4.25)$$

Agora, considera-se os seguintes parâmetros para obter o tempo médio do *CFP* dado um determinado valor de *SI*, onde:

$C_i$  = Tempo total de transmissão de uma estação *GSC* mais *SIFS*

$np$  = Numero total de estações *GSC* admitidas para o *SI*

$PER_{ci}$  = Taxa de Erro de Pacote de mensagem da estação *GSC*

$PER_{ba}$  = Taxa de Erro de Pacote para o quadro *RTBlockAck*

$PER_b$  = Taxa de Erro de Pacote para o quadro de *Beacon*

$M_t$  = Percentual das estações *GSC* aptas para transmitir mensagens

Então, o tempo médio de duração do *CFP* operando em um canal ruidoso é igual a:

$$\begin{aligned} CFP = & \Theta + (1 - PER_b) \cdot M_t \left( \sum_{i=1}^{np} C_i \right) + (1 - PER_b) \cdot (1 - M_t) np \cdot C_{st} \\ & + PER_b \cdot np \cdot C_{st} + (1 - PER_b) \cdot M_t \cdot (1 - PER_{ba}) \left( \sum_{i=1}^{np} PER_{ci} \cdot C_i \right) \\ & + (1 - PER_b) \cdot M_t \cdot (PER_{ba}) \cdot \left( \sum_{i=1}^{np} PER_{ci} \right) \cdot C_{st} \end{aligned} \quad (4.26)$$

Onde a probabilidade de perda de uma mensagem pela  $n$ -ésima estação  $GSC$  é descrito por:

$$P_{drop_{np}} = M_t \cdot PER_b + M_t(1 - PER_b) \cdot PER_{ci} [PER_{ba} + (1 - PER_{ba}) \cdot PER_{ci}] \quad (4.27)$$

Agora por motivos de simplificação para comparação com o  $HCCA$ , pode-se considerar as seguintes hipóteses que: não há perda dos quadros de *Beacon*, *RTBlockAck* e *CFEnd*, logo considera-se apenas as perdas das mensagens de tempo real tanto no ciclo  $VTP$  como na segunda chance. Então, tem-se o tempo médio do  $CFP$ , descrito como:

$$CFP = \Theta + M_t \left( \sum_{i=1}^{np} C_i \right) + (1 - M_t) \cdot np \cdot C_{st} + M_t \left( \sum_{i=1}^{np} PER_{ci} \cdot C_i \right) \quad (4.28)$$

#### 4.2.5 Aspectos Relacionados à Robustez

Nesta subseção descreve-se o comportamento do mecanismo  $GSC$  em situações de erro. Será dada ênfase somente ao período livre de contenção, pois no período de contenção o comportamento é padronizado pelo Padrão IEEE 802.11/11e. Admitiram-se primeiramente algumas condições iniciais:

1. O modo de operação da rede é infra-estruturado.
2. O funcionamento das estações de tempo real é controlado pelo mecanismo  $GSC$  e as estações genéricas ( $DCF/EDCA$ ) são suportadas totalmente, sem restrições de número e tipo.
3. Não existe a possibilidade de haver situação onde o  $AP$  e as estações  $GSC$  estejam ocultas.
4. Não há mobilidade das estações  $GSC$  ou esta mobilidade é pequena e limitada a área de cobertura do  $AP$ . Pois, tratando-se de um ambiente de chão de fábrica é razoável admitir que as estações de tempo real sejam alocadas de forma fixa seguindo um planejamento técnico prévio da localização que impede ou minimiza esta situação.
5. Há probabilidade de acontecer erros nas transmissões dos pacotes devido a ruído no canal. Estes erros podem afetar parte ou a totalidade das estações no chão de fábrica.

A seguir são citadas situações de erros comuns de ocorrer e como o mecanismo GSC pretende tolerar para operar de forma satisfatória.

- **Transmissão corrompida de uma determinada estação GSC** - Neste caso, a estação de tempo real não consegue detectar se houve erro na transmissão da sua mensagem gerando duas situações:
  - i) O AP detectou que o quadro foi enviado com erros. A recuperação faz-se através do quadro *RTBlockAck* utilizando o algoritmo de segunda chance;
  - ii) O AP não detectou o erro no quadro. Isto pode acontecer no caso da fonte de erro estar localizada junto do receptor (ou receptores) e não junto ao AP. Neste caso o quadro de *RTBlockAck* vai informar que a mensagem foi transmitida com sucesso, quando na verdade algumas estações não receberam a mensagem. Desta forma, essas estações vão descartar o quadro. Isto equivale à perda do quadro nesse ciclo de *SI* para as estações próximas à fonte de interferência.
  
- **Transmissão de Quadro de Beacon corrompida** - Neste caso, todas as estações recebem um quadro de *Beacon* corrompido. As estações GSC ignoram este quadro e ficam à espera do próximo *Beacon*. Como o AP não vai transmitir o *Beacon* no *SI* corrente, o meio não vai ficar reservado, logo o *CP* inicia-se imediatamente. Esta situação é análoga a uma extensão do *CP* vindo do ciclo anterior de *SI*. O AP detecta esta situação da seguinte forma:
  - i) Se detectar tráfego que corresponde a estações genéricas;
  - ii) Se não detectar tráfego de nenhuma estação GSC. Nesta situação, o AP não envia o quadro de *RTBlockAck*, pois admite-se que o ciclo do GSC não se iniciou. A consequência desta falha é o atraso do ciclo do GSC num valor superior ao *CP*.
  
- **Uma ou mais estações não recebem o quadro Beacon devido a interferências locais que afetam apenas essas estações.** Neste caso, surgem duas situações distintas:
  - i) Estações GSC - Neste caso, essas estações de tempo real não participam do ciclo VTP. Pois, o ciclo se inicia com o recepção correta do *Beacon*. No entanto, para as outras estações GSC o ciclo inicia-se normalmente (pois receberam o *Beacon*). Para as estações que não receberam o *Beacon* é equivalente a não ter mensagem a enviar e a ficha virtual passa para a próxima estação do grupo GSC ao fim de um período de *aSlotTime* (ST). Porém, as estações GSC que não receberam o *Beacon* têm uma nova oportunidade de transmissão quando recebem o quadro *RTBlockAck*. Verificam que não há confirmação da sua transmissão (porque a mensagem não



foi enviada) e entram no processo de retransmissão com o algoritmo de segunda chance.

ii) Estações Genéricas - Neste caso, temos uma situação mais complicada. Para a estação genérica o meio não está reservado (porque não recebeu o quadro de *Beacon*). Portanto, ela pode iniciar transmissões quando o meio ficar livre gerando colisões com os pacotes de tempo real. Contudo, isto é uma situação atípica (probabilidade mínima), pois, se o ciclo do *GSC* já tiver iniciado as estações genéricas só podem transmitir se o tempo de inatividade do canal for superior ou igual à *DIFS*. Isto só acontece se mais do que uma estação *GSC* consecutiva não transmitir seus quadros. A estação genérica pode ter uma improvável oportunidade (incluindo o procedimento de segunda chance). No entanto existe a chance de uma estação genérica ter acesso ao meio e transmitir seus pacotes antes do início do *CP*. Nesse caso, a consequência disto é o atraso do ciclo *GSC*, mas é interessante notar que o *NAV* mais o *SIFS* para o *GSC* é uma dupla garantia que as estações genéricas dificilmente conseguiriam transmitir durante o ciclo *GSC*.

- **O quadro do *RTBlockAck* é corrompido por ruídos** - Neste caso, nenhuma estação *GSC* recebe com sucesso o quadro de *RTBlockAck*. As estações *GSC* não fazem nada e esperam pelo quadro de *CF-End*. O *AP* ao detectar que não há tráfego de tempo real no procedimento de segunda chance envia o quadro de *CF-End*. Assim, se houver alguma transmissão corrompida, essa mensagem é descartada. Pode-se estimar o tempo máximo de espera do *AP* igual a  $(np.C_{st}) + 2 * SIFS$ , antes do envio do quadro de *CF-End*, caso não existam retransmissões.
- **Uma estação *GSC* descarta o quadro de *RTBlockAck*** - Similar ao caso anterior, a estação fica impossibilitada de confirmar se o seu quadro foi enviado corretamente ou não. Nesse caso, não faz nada e espera pelo quadro de *CF-End*.
- **Uma ou várias estações descartam o quadro *CF-End*** - Neste caso, não há *reset* do *NAV* enviado pelo *Beacon*. Logo, estas estações (*GSC* e Genéricas) esperaram pelo fim deste *NAV* havendo desperdício da largura de banda.
- **Erros gerados por ruídos durante o procedimento de segunda chance** - Neste caso, os quadros envolvidos pelo ruído vão ser descartados pelas estações receptoras.

### Considerações Finais Sobre o Mecanismo GSC

Neste capítulo foram apresentados os detalhes do projeto e implementação do mecanismo GSC para aplicações de tempo real rígido. A proposta foi concebida para manter os requisitos de separação e parametrização de tráfego síncrono utilizando o conceito Produtor/Consumidor para rede infra-estruturada compartilhada com estações genéricas (IEEE 802.11/11e). Observa-se que situações de erros podem ocorrer de formas isoladas ou combinadas entre si, mas o mecanismo proposto tende a tolerar estas falhas e operar satisfatoriamente. Nota-se também que é preciso ter especial atenção ao AP, dado que a sua avaria torna a rede não operacional. Logo, é necessário considerar as técnicas existentes de redundância. Outro ponto importante é o fato que na prática o quadro de Beacon pode não ser transmitido periodicamente. Isto se deve ao fato de utilizar estações legadas (IEEE 802.11). Isto pode implicar um jitter no Beacon que por sua vez vai ter consequências no mecanismo GSC, mas que é facilmente contornável através dos ajustes dos parâmetros de escalonamento, que é realizado on-line pelo HC. Já estações padrão IEEE 802.11e (EDCA) não devem causar jitter no tempo de transmissão do Beacon, pois calculam seus TXOPs antes de transmitir seus quadros. Desta forma, as estações EDCA não ultrapassam o valor estabelecido para SI.

No capítulo seguinte será realizada a análise de desempenho comparativo entre os mecanismos de transmissão com requisitos de tempo real GSC e HCCA para cenários industriais típicos de supervisão e controle, apresentando em gráfico o seus respectivos resultados.

---

## Capítulo 5

# Análise de Desempenho do GSC

---

O objetivo deste capítulo é apresentar a metodologia para a análise de desempenho comparativo entre os mecanismos *GSC* e *HCCA* em dois cenários industriais (supervisão e controle) dentro de um ambiente de rede sem fio, verificando a robustez dos mecanismos de transmissão quando o meio for susceptível a ruídos no canal de comunicação. Foram investigados os seguintes quesitos: duração máxima do período livre de contenção, latência média e taxa de sucesso na entrega dos pacotes. Todos os resultados foram obtidos utilizando ferramentas de simulação de redes desenvolvida em C/C++ denominadas *GSCSimu* e *HCCASimu*, respectivamente.

### O Ambiente de Simulação do Mecanismo GSC

O recurso da simulação é muito utilizado em análise de desempenho de redes de comunicação, permitindo emular cenários reais contendo parâmetros que podem ser facilmente modificados ou agregar novas propostas de protocolos de forma rápida sem necessidade da compra de equipamentos. Atualmente, existem vários simuladores disponíveis para uso, entretanto poucos são utilizados devido à complexidade, falta de documentação ou ter recursos limitados para a simulação [Fall 2001]. O mais difundido é o *Network Simulator 2* (NS-2) [Project Online - accessed 26 June 2008], que por ser um simulador gratuito e de código aberto, possui razoável documentação e boa aceitação pela comunidade científica.

Segundo [Hanley 2005] O NS-2 é um simulador de eventos discretos orientado a objetos desenvolvido em linguagem C++ em conjunto com a linguagem de *script* OTcl (*Object Tool Command Language*). O núcleo do simulador foi desenvolvido em C++ para garantir bom desempenho de execução e a linguagem OTcl é utilizada para interpretar os *scripts* criados pelo usuário sem que seja necessário alterar diretamente o código do simulador. O simulador NS-2 é amplamente utilizado na área de redes, porém exige certa dedicação para a sua aprendizagem devido a sua complexidade. Este simulador é consi-

derado uma ferramenta pouco amigável e não apresenta interface gráfica para o usuário; como alternativa, usa uma ferramenta de animação para a visualização dos dados simulados chamado *Nam*. Para se utilizar o NS-2 de forma satisfatória é necessário demandar muito tempo resolvendo problemas de implementação e as vezes isto resulta em modelos ineficientes para representar a rede [Müller 2006]. Além dessas questões, há ainda a ausência de suporte nativo ao padrão IEEE 802.11e.

Segundo [Viegas 2009] existem extensões do padrão IEEE 802.11e para o NS-2 que suportam o *HCCA*, criadas originalmente por [Ciconetti et al. 2005b] e depois melhorada por [Demarch 2007]. Estas implementações apresentam algumas falhas, tais como; sequência desordenada de troca de pacotes e erros de sincronismo, que leva em alguns casos a resultados em não conformidade com as especificações do Padrão IEEE 802.11e.

No início de desenvolvimento do simulador *GSCSimu* foram feitas modificações no código do NS-2 utilizando o *patch* para IEEE 802.11e. Devido à demanda por inúmeras adaptações e problemas em expandir o código fonte, optou-se por desenvolver um simulador próprio em C/C++. Desta forma, a principal razão que levou à criação do simulador *GSCSimu* foi a necessidade de flexibilidade para que todos os recursos especificados na proposta pudessem ser facilmente implementados e mais tarde comparados com as funcionalidades do mecanismo *HCCA* pertencente ao padrão IEEE 802.11e.

## 5.1 Descrição dos Cenários de Comunicação Industrial

A característica principal do tráfego industrial é a predominância de pequenos pacotes que devem ser transferidos periodicamente entre os sensores, controladores e atuadores de acordo com rígidos períodos de tempo (*deadline*) e a baixa taxa de perdas [Willig 2008]. Esta propriedade torna o tráfego industrial diferente do tráfego de melhor esforço (sem perdas e sem requisitos de tempo real) e do tráfego de multimídia (requisitos flexíveis de tempo, mas relativamente sensível a *jitter* e a perdas de dados). Aplicações para automação industrial não necessitam de altos *throughputs*, mas sim de uma largura de banda mínima, sendo então importante termos como referência cenários distintos e suas limitações baseado em valores reais.

Atualmente, existem dois tipos principais de paradigmas associados às comunicações industriais: mestre-escravo (*master-slave*) e produtor/consumidor (*publisher/subscriber*) [Cena et al. 2008].

- O paradigma mestre-escravo é o mecanismo mais simples de comunicação, onde o remetente só precisa saber o endereço do destinatário para enviar suas mensagens.

O modelo utiliza mecanismo de transmissão *unicast*, onde o mestre sempre inicia a comunicação para um escravo específico e este escravo apenas responde a essa requisição de dados. Este paradigma tem como principal exemplo o *HCCA*.

- O paradigma produtor/consumidor de comunicação utiliza mecanismos de *multicast* ou *broadcast*, onde o componente produtor (*publisher*) sempre inicia a comunicação transmitindo (publicando) a sua mensagem sobre a rede. Nesse caso, a mensagem não é endereçada a uma estação (*host*) específica. Em vez disso, cada mensagem tem um identificador sendo que um ou mais consumidores (*subscriber*) podem simultaneamente consumir estas mensagens. Este paradigma tem o *GSC* como exemplo.

As redes de supervisão e controle são importantes para a automação industrial e dividem-se em três funções básicas: supervisão, operação e controle. Um sistema supervísório monitora as informações de um processo produtivo em uma planta industrial. Estas informações são obtidas através de um equipamento de aquisição de dados onde são processados e posteriormente disponibilizados aos usuários. Estes sistemas são conhecidos como *SCADA* (*Supervisory Control and Data Acquisition*) e têm como objetivo promover uma interface de alto nível entre operador e processo produtivo, informando em tempo real todos os eventos importantes.

O processo de controle utiliza os *PLCs* (*Programmable Logic Controllers*) que são unidades computacionais específicas, usadas em instalações industriais com a função de obter os dados dos sensores, operar cálculos e enviar sinal de controle aos atuadores.

Rede de comunicação industrial é o meio onde trafegam as mensagens de tempo real oriunda dos sensores, controladores e atuadores. Dependendo dos requisitos da rede (área de abrangência do sistema), pode ser implementada através de cabos *Ethernet*, fibras ópticas, redes sem fio ou uma mistura destes meios.

Nesta seção consideraremos dois cenários industriais sem fio operando com dois ambientes distintos para realização dos testes, são eles:

- Canal livre de ruídos (*Noisy-Free Channel*)
- Canal com ruídos (*Noisy Channel*)

Cada ambiente é composto por dois cenários distintos que ilustram redes industriais reais, como ilustrado nas Figuras 5.1 e 5.3. A Figura 5.1 é chamada de cenário de supervisão e a Figura 5.3 é chamada de cenário de controle. Todos os cenários permitem o estabelecimento de comunicação entre todas as estações e o ponto de acesso sem a ocorrência do fenômeno de estação oculta.

- O primeiro cenário de simulação (Figura 5.1), chamado cenário de supervisão foi definido como um grupo de vinte (20) estações de tempo real (sensores sem fio), que transmitem um fluxo de mensagem contendo 40 bytes de dados em seus respectivos TXOP's. Para este cenário é considerado um conjunto de sensores ( $S_i$ ) que estão conectados por meio de dispositivos sem fio compatível com o padrão IEEE 802.11e. Os sensores medem uma grandeza física (por exemplo, temperatura e pressão) e convertem em pacotes de dados. Todas as mensagens transmitidas a partir dos sensores têm restrições temporais (*deadline*) e formam os dados que fazem parte da carga do sistema. Estas mensagens de tempo real dos sensores ( $S_i$ ) são endereçadas periodicamente para uma estação de supervisão SCADA conectada via rede Ethernet Industrial através de um Ponto de Acesso (AP) à rede sem fio, que utiliza os mecanismos de transmissão de mensagens usando o GSC e HCCA, respectivamente. Em seguida ilustra-se o diagrama temporal referente ao cenário de supervisão para o GSC e HCCA através da Figura 5.2.

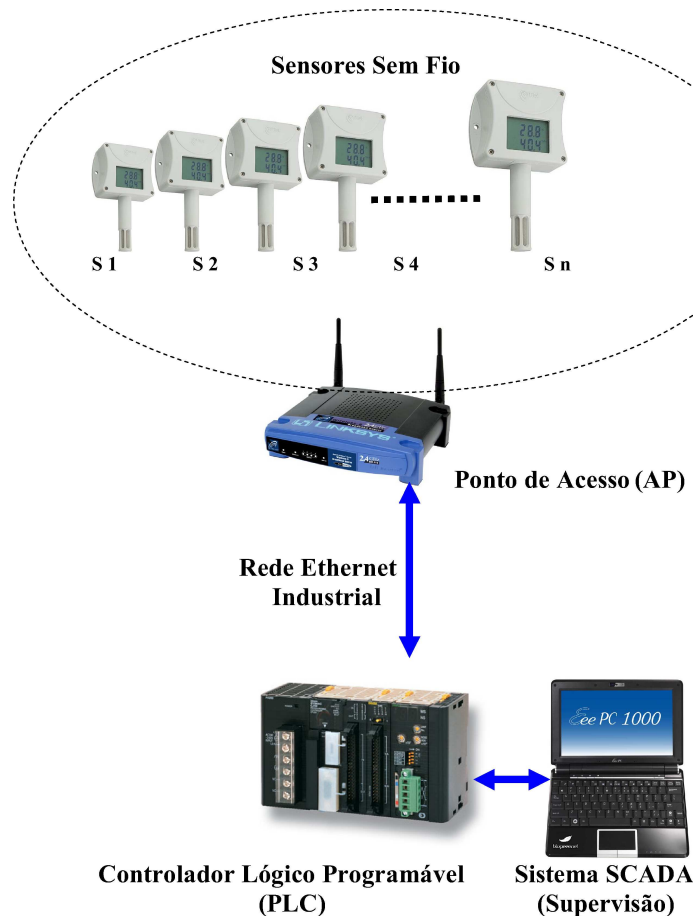
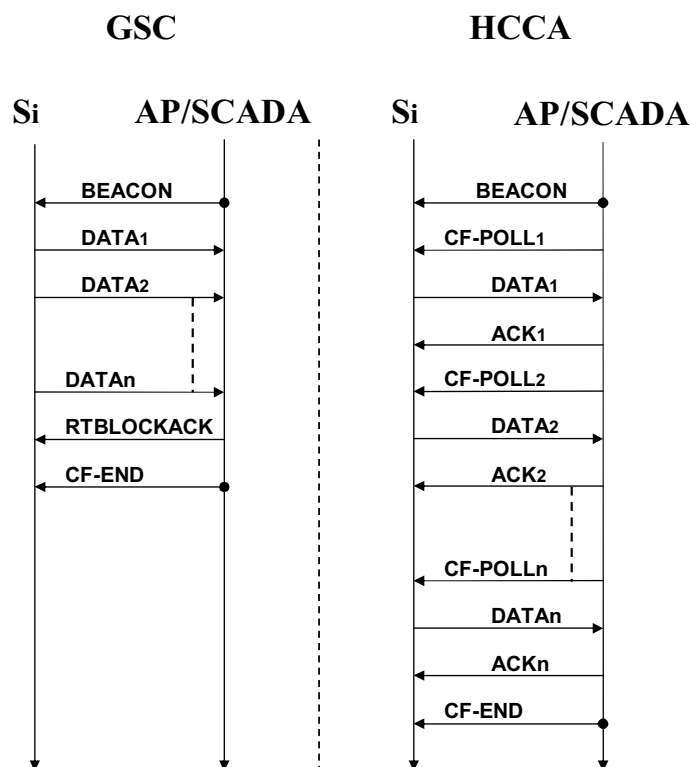


Figura 5.1: Cenário de Supervisão.



**Figura 5.2:** Diagrama Temporal do *GSC/HCCA* para Cenário de Supervisão.

- O segundo cenário de simulação (Figura 5.3), chamado cenário de controle foi definido como um grupo de vinte (20) estações de tempo real (sensores sem fio) que medem uma grandeza física (por exemplo, temperatura e pressão) e converte em uma mensagem de tempo real transmitida para o controlador da planta industrial através do *AP*. No caso do *HCCA*, o *AP* oferece uma função de retransmissão local (*relay*) para este cenário, pois utiliza-se o paradigma de comunicação mestre/esravo. O controlador lógico programável (*PLC*) processa os dados e envia os resultados para os atuadores através da rede de campo (*Fieldbus*) afim de atuar no controle do processo industrial.

Para o mecanismo *GSC*, o cenário de controle opera de maneira análoga ao cenário de supervisão, pois apresenta um comportamento idêntico para ambos os cenários ao utilizar o paradigma de comunicação produtor/consumidor. Esta mudança de cenário em nada modifica o diagrama temporal do *GSC*, mas diferentemente deste comportamento, a mudança de cenário afeta principalmente o desempenho do mecanismo *HCCA*. Pode-se ver o cenário de controle em formato de diagrama temporal para o *GSC* e *HCCA* ilustrados na Figura 5.4, respectivamente.

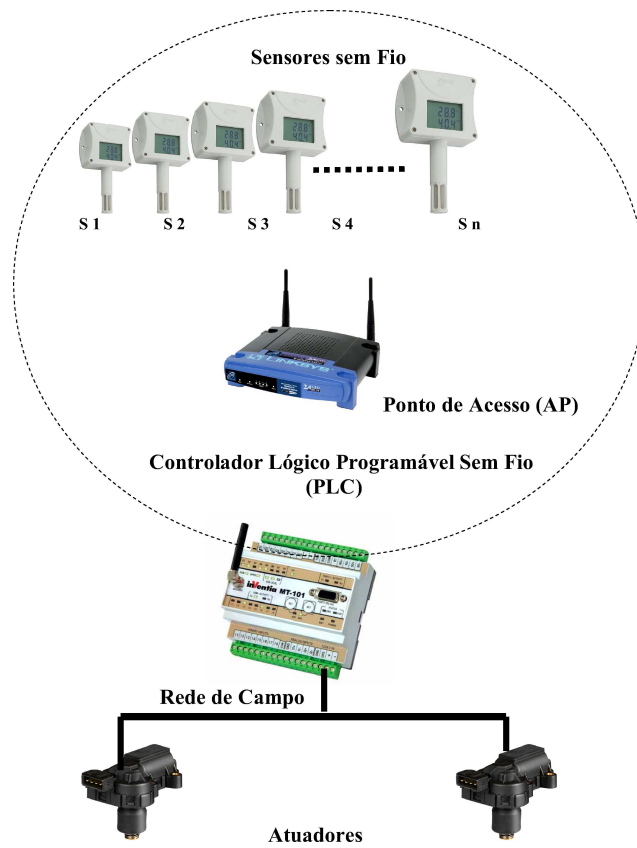


Figura 5.3: Cenário de Controle.

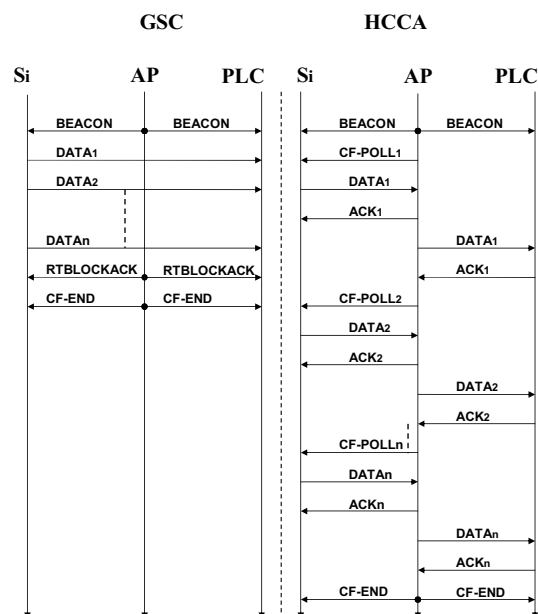


Figura 5.4: Diagrama Temporal do GSC/HCCA para Cenário de Controle.



### 5.1.1 Avaliação dos Mecanismos GSC e HCCA

O mecanismo de GSC foi desenvolvido e implementado em C/C++ dando origem ao simulador *GSCSimu* [Viégas Jr et al. 2009]. Esta implementação do GSC foi comparada com o mecanismo HCCA que foi implementado de forma análoga em [Viegas 2009] chamado *HCCASimu*, tendo como base o simulador de redes NS2 baseado em [Cicconetti et al. 2005b] com modificações de [Demarch 2007]. Foi considerado somente o tráfego GSC e HCCA no período livre de contenção (CFP), onde o tempo de transmissão de um fluxo de mensagem de tempo real por estação é conhecido antecipadamente.

Todas as estações GSC/HCCA inseridas no ambiente de simulação têm o mesmo padrão de geração de tráfego com taxa de bit constante (*Constant Bit Rate - CBR*) e todos os pacotes de dados (40 bytes<sup>11</sup>) em ambos os mecanismos são marcados como de maior prioridade (tempo real), não ocorrendo o fenômeno de estações ocultas e nem a perda de quadros de *Beacon* e de reconhecimento (*ACK* e *RTBlockACK*). A taxa de perda de pacote de dados (*PER*) é considerada independente.

Os testes foram realizados utilizando o padrão IEEE 802.11b/e com taxa de transmissão nominal de 11Mbps e 1Mbps para as taxas básicas de acordo com [IEEE 1999c] conforme apresenta a Tabela 5.1.

**Tabela 5.1:** Parâmetros da Camada Física do IEEE 802.11b.

Parâmetros	802.11b
Tempo de SIFS	10 $\mu$ s
Tempo de aSlotTime	20 $\mu$ s
Tempo de PIFS	30 $\mu$ s
Tempo de DIFS	50 $\mu$ s
Tempo de Preâmbulo	144 $\mu$ s
Tempo de Cabeçalho PLCP	48 $\mu$ s
Taxa Nominal de Dados	11Mbps
Taxa Básica de Dados	1Mbps
Atraso de Propagação	2 $\mu$ s

<sup>11</sup>Tamanho mínimo de um pacote de dados da rede PROFINET [Trsek et al. 2006].

### 5.1.2 Simulação da Rede Utilizando Canal Livre de Ruídos

As simulações foram realizadas variando o número de estações de tempo real (sensores sem fio) de 02 até 20 <sup>12</sup>. Foram configurados os mecanismos *GSC* e *HCCA* para transmitir um fluxo de mensagem contendo 40 bytes de dados por estação. O ambiente de teste é livre de ruídos e a taxa de chegada de pacotes nas estações (*Mean Data Rate*) é fixado em 3.200 bps. O percentual de estações *GSC* e *HCCA* com pacotes na fila de transmissão foi variado em duas faixa de valores: a faixa de 50% (metade do número de sensores transmitem mensagens no SI) e 100% (todos os sensores enviam mensagens no SI). O intervalo de Serviço (SI) é fixado em 100ms, que é um valor típico para uso em AP's comerciais [Pavón & Shankar 2004] e atende os requisitos de controle de admissão e escalonamento para ambos os mecanismos de transmissão. O tempo de simulação foi fixado em 10.000 segundos (100.000 ciclos de *CFP* (*rounds*)), sendo suficiente para assegurar confiabilidade estatística dos resultados.

A função básica do quadro de *Beacon* é sincronizar todas as estações da rede sem fio. Assim, considera-se que para operar em tempo real, todas as estações devem adquirir os seus dados em um tempo mínimo igual a  $\delta \ll SI$  antes da recepção dos quadros de *Beacon*, transmitindo suas mensagens de acordo com o seu respectivo mecanismo de comunicação. Todos os parâmetros utilizados na simulação da rede para os dois cenários (Supervisão e Controle) são apresentados na Tabela 5.2.

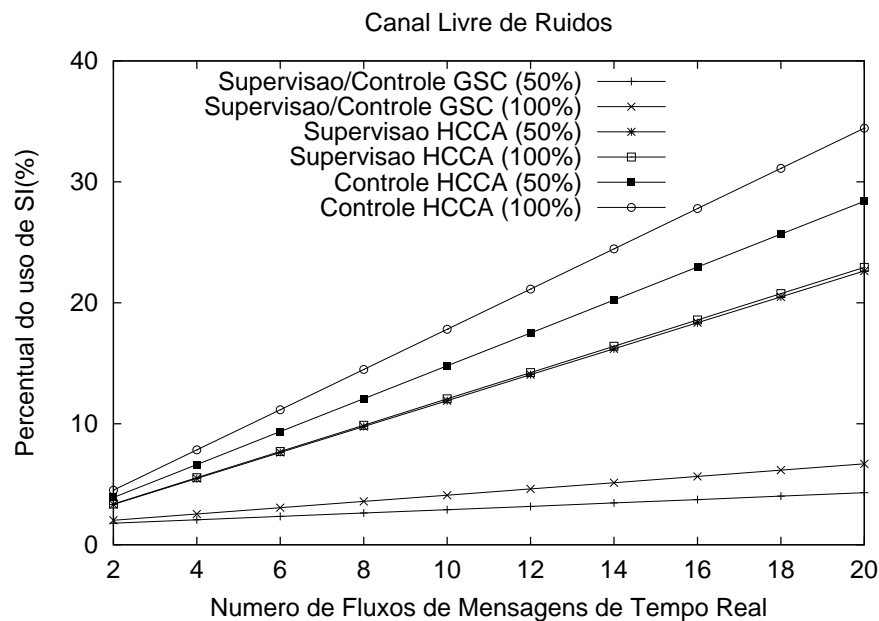
**Tabela 5.2:** Parâmetros de Simulação da Rede.

Parâmetros	Valores
Tempo de Simulação	10.000 s
Ciclos Simulados de CFPs	100.000 ciclos
Tempo de Intervalo de Serviço	100 ms
Tamanho do Pacote de Dados	40 Bytes
Estações de Tempo Real Admitidas	20 estações
Taxa de Chegada dos Pacotes	3.200 bps
Percentual da Estações Transmissoras	50 e 100 %
Taxa de Erro de Pacotes ( <i>PER</i> )	0 %

As três métricas que foram utilizadas para avaliar o desempenho dos mecanismos de comunicação *GSC* e *HCCA* são rapidamente descritas a seguir.

<sup>12</sup>O número máximo de estações admitidas pelo padrão IEEE 802.11b é de 32 estações [IEEE 1999c].

- A primeira métrica define-se como o tempo duração máxima do período livre de contenção (*CFP*), que é igual à somatória de todos os TXOPs dos sensores mais o *overhead* da rede para ambos os mecanismos de transmissão (*GSC/HCCA*).
- A segunda métrica define-se como latência média de transmissão da rede. Esta métrica baseia-se na diferença entre os tempos de aquisição de dados e a efetiva transmissão das mensagens de tempo real utilizando ambos os mecanismos. Os valores médios de tempo de latência podem ser obtidos através da somatória de todas as latências dos sensores dividindo pelo número total de ciclos de *CFP* simulados.
- A terceira métrica define-se como taxa de sucesso da entrega de mensagens de tempo real para ambientes ruidosos com taxas de perdas independente.
- **Resultado 1** - A Figura 5.5 mostra a duração do *CFP* (expresso em percentual de utilização de *SI*) em função do percentual de sensores aptos ( $M_t$ ) a transmitir, que pode variar de 50% (em média metade dos sensores) e 100% (todos os sensores), para um fluxo de mensagem de 40 bytes de dados utilizando os mecanismos *GSC/HCCA* por estação de tempo real.



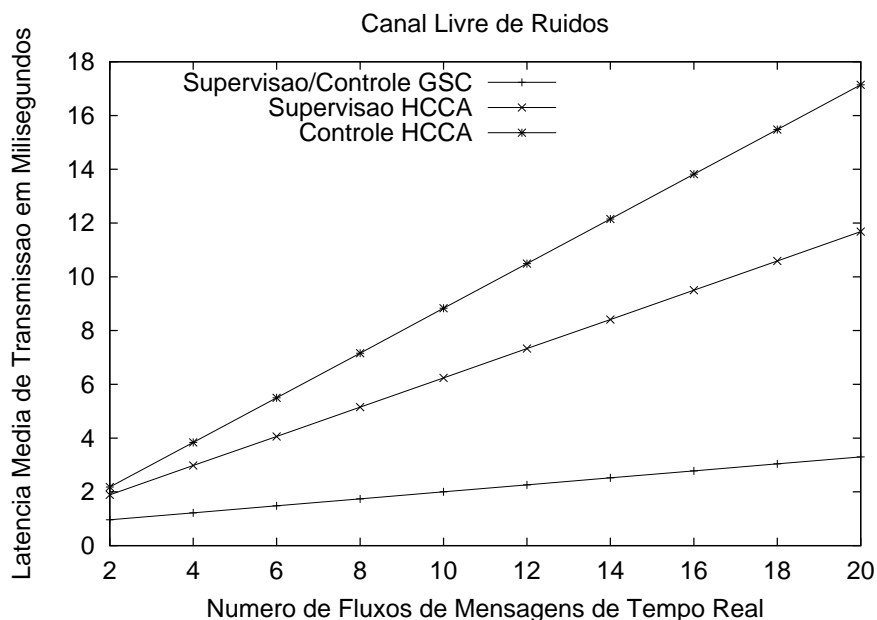
**Figura 5.5:** Percentagem do SI para Canal Livre de Ruídos.

O mecanismo *GSC* usa o percentual (%) de *SI* (*CFP*) em torno de 6,7%, para uma rede onde todos os 20 (vinte) sensores transmitem dados ( $M_t = 100\%$ ), enquanto

que o mecanismo *HCCA* usa 22,9% de *SI (CFP)* para o cenário de supervisão e 34,4% de *SI (CFP)* para o cenário de controle. Quando o percentual total dos sensores aptos a transmitir ( $M_t$ ) é alterado e fixado em torno de 50%, foi verificado que o *GSC* reduz o valor do *CFP* anterior para 4,3% de *SI*, enquanto o *HCCA* reduz para 22,6% de *SI* para o cenário de supervisão e 28,4% de *SI* para o cenário de controle.

Analisando este caso, o *GSC* tem a vantagem de economizar largura de banda diminuindo a utilização do *CFP* quando parte dos sensores não tiverem mensagens a transmitir, isto comparado ao *HCCA*. Isto acontece porque, o mecanismo *HCCA* tem a obrigação segundo a norma IEEE 802.11e de sempre transmitir um pacote nulo "*QoS Null*" quando a estação requisitada não tiver nenhum pacote a transmitir, desta forma há uma redução mínima da utilização do *CFP*, por outro lado, não há um aumento do CP reservado para as transmissões dos dados das estações genéricas.

- **Resultado 2** - A Figura 5.6 mostra a latência média de transmissão com base no número máximo de sensores admitidos ( $M_t = 100\%$ ) utilizando taxa de chegada dos pacotes (*Mean Data Rate*) fixo em 3.200 bps para o *GSC/HCCA*.

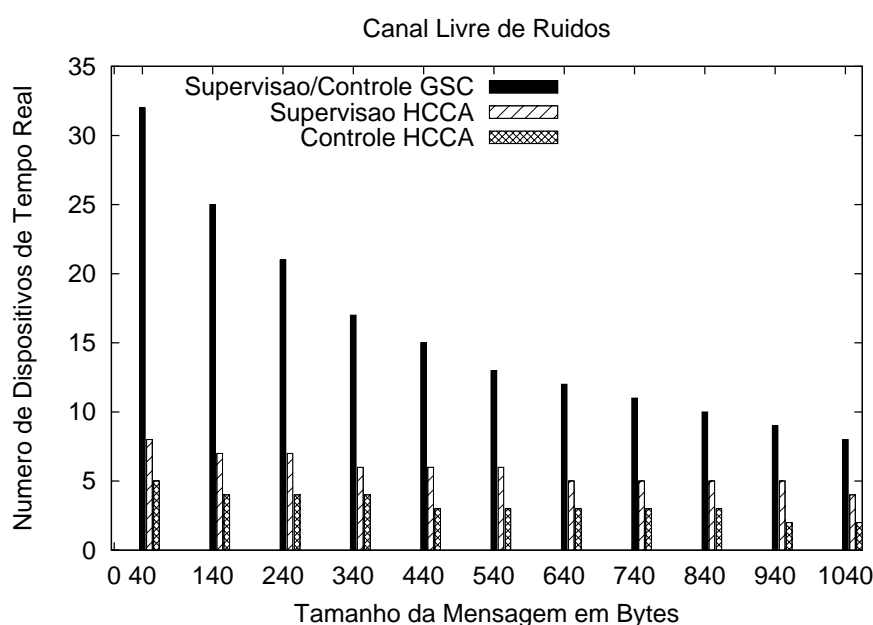


**Figura 5.6:** Latência Média de Transmissão para Canal Livre de Ruídos.

Os resultados mostram que a latência média de transmissão do *GSC* tem um melhor desempenho se comparados ao *HCCA*. Por exemplo, para os vinte sensores

(estações de tempo real), o mecanismo *GSC* tem uma latência média de 3,3ms, enquanto os valores de latência do mecanismo *HCCA* são de 11,7ms para o cenário de supervisão e 17,1ms para o cenário de controle. Isto mostra que o mecanismo *GSC* tem uma latência média de transmissão inferior *HCCA* em todos os casos de comparação, significando que o *GSC* é capaz de transmitir mais mensagens de tempo real que o *HCCA* quando utiliza-se mesmo percentual de utilização do intervalo de serviço (*CFP*).

- **Resultado 3** - A Figura 5.7 ilustra a comparação entre os mecanismos de transmissão (*GSC/HCCA*), onde ambos operam com o número máximo de sensores possíveis dentro do *CFP* de valor fixo em 10 ms. Neste cenário é permitido variar o tamanho da mensagem de tempo real de 40 bytes até 1040 bytes (em passos de 100 bytes). Neste teste existe apenas um fluxo de dados por estação e o destino de todas as mensagens é o *AP* (equivalente ao cenário de Supervisão). O resultado é mostrado a seguir.



**Figura 5.7:** Tamanho dos Pacotes Versus Numero de Estações de Tempo Real.

Como pode ser visto na Figura 5.7, o mecanismo *GSC* é capaz de transmitir mensagens de 32 sensores, enquanto o mecanismo *HCCA* é capaz de transmitir mensagens de apenas 08 sensores para o cenário de supervisão e 05 sensores para o cenário de controle, utilizando um pacote de dados fixo em 40 bytes usando o mesmo *CFP*.

Analisando-se o extremo do gráfico para o tamanho do pacote de dados fixo em 1040 bytes o *GSC* consegue transmitir mensagens de 08 sensores versus 04 sensores para o cenário de supervisão, 02 sensores para o cenário de controle utilizando o mecanismo *HCCA*. No entanto, quando o tamanho da mensagem aumenta, o número de sensores *GSC* diminui mais rapidamente quando comparado com o *HCCA*. Pode-se observar que o número de sensores é dependente do tamanho do pacote e diminui com o aumento deste. Isto se torna claro, porque quanto menor o tamanho do pacote, maior o número de estações aptas para transmitir mensagens de tempo real no *CFP*. De outra forma, a latência média de transmissão para este cenário de supervisão dependerá do número de estações de tempo real (*HCCA/GSC*), que pode usar o *CFP*. Devido a esta característica, o *GSC* apresenta desempenho superior ao *HCCA* principalmente para transmissão de pequenos pacotes (caso característico de aplicações em automação industrial).

A Tabela 5.3 mostra agora os tempos envolvidos na duração do *CFP* para o mecanismos *HCCA/GSC* transmitindo um fluxo de dados de 40 bytes para cenário de controle. Pode-se notar que o mecanismo *GSC* tem uma maior redução do *overhead* da rede se comparado ao *HCCA* devido ao uso do *VTP* baseado na comunicação produtor/consumidor, pois não utiliza quadros de *Polling* e *ACK*. Isso significa cerca de 38,27% de diminuição no custo de transmissão de mensagens do *GSC* durante a vigência do *CFP* comparando-se com o *HCCA*.

**Tabela 5.3:** Tempo de Transmissão do *HCCA* e *GSC*.

Mecanismo	HCCA	GSC
SIFS	10 $\mu s$	10 $\mu s$
aSlotTime	20 $\mu s$	20 $\mu s$
PIFS	30 $\mu s$	—
Cabeçalho Físico	192 $\mu s$	192 $\mu s$
Cabeçalho MAC	26.18 $\mu s$	26.18 $\mu s$
Atraso de Transmissão	2 $\mu s$	2 $\mu s$
Quadro de Beacon	826 $\mu s$	826 $\mu s$
Quadro de Interrogação ( <i>Poll</i> )	482 $\mu s$	—
Pacote de Dados	249.27 $\mu s$	249.27 $\mu s$
Quadro ACK	306 $\mu s$	—
Quadro RTBlockACK	—	304,55 $\mu s$
Quadro CF-End	354 $\mu s$	354 $\mu s$
Tempo Total de Duração do CFP	2848,54 $\mu s$	1758,36 $\mu s$

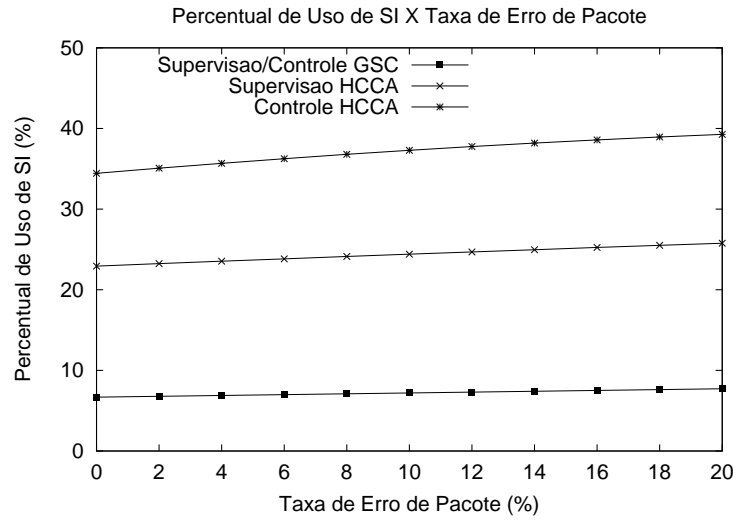
### 5.1.3 Simulação da Rede Utilizando Canal com Ruídos

As simulações foram realizadas com um número fixo de 20 sensores sem fio transmitindo um fluxo de dados de 40 bytes ( $M_t = 100\%$ ). O ambiente de teste é susceptível a ruídos e os eventos são considerados independentes. A taxa de erro de pacote foi variada em intervalos fixos de 0% à 20% com passos de 2%. O ruído pode afetar somente o pacote de dados (*payload*). Todos os cenários de simulação foram executados à semelhança dos ensaios anteriores. O tempo de simulação é igual a 10.000 segundos (100.000 ciclos de CFP) para garantir a confiabilidade estatística. O SI é fixado em 100ms. Todas as estações transmitem um fluxo de mensagem. Os parâmetros da simulação da rede são apresentados na Tabela 5.4.

**Tabela 5.4:** Parâmetros de Simulação para Cenário com Ruído .

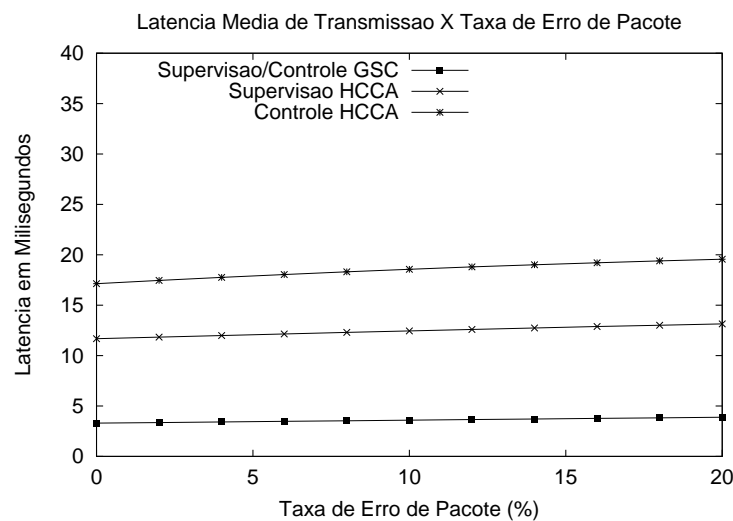
Parâmetros	Valores
Tempo de Simulação	10.000 s
Ciclos Simulados de CFP	100.000 ciclos
Intervalo de Serviço	100 ms
Tamanho do Pacote de Dados	40 Bytes
Estações Admitidas	20 estações
Taxa de Chegada dos Pacotes	3.200 bps
Percentual de Estações Transmissoras	100 %
Taxa de Erro de Pacote ( <i>PER</i> )	0 a 20 %

- Resultado 1** - A Figura 5.8 mostra a duração do CFP em função da variação do *PER* de 0% à 20% em passos de 2%, para 20 (vinte) sensores sem fio, com  $M_t = 100\%$ . Como podemos ver, o mecanismo *GSC* usa o percentual de SI de 6,7% até 7,8% (variação de 1,1%), enquanto o *HCCA* usa de 22,9% até 25,7% (variação de 2,8%) para um cenário de supervisão, de 34,4% até 39,2% (variação de 4,8%) para o cenário de controle, respectivamente. Esta observação está relacionada ao aumento da duração do CFP em função do aumento de variação *PER*. Podemos notar que a *GSC* tem significativamente menos aumento de utilização do CFP se comparado ao *HCCA* para todos os cenários e, novamente, não há desperdício de largura de banda pelo *GSC*, mesmo em um ambiente ruidoso com o *PER* variável. Portanto, há um ganho de tempo no período de contenção (CP) quando se utiliza o *GSC* compartilhando a banda da rede com as estações genéricas.



**Figura 5.8:** Duração do *CFP* em função da variação do *PER*.

- Resultado 2** - Como pode-se notar na Figura 5.9, o mecanismo *GSC* tem o menor latência média de transmissão em relação ao *HCCA*. Mas se tomarmos um *PER* igual a 10% como referência (cenário típico de uma rede sem fio), o mecanismo *GSC* tem uma latência de transmissão média de 3,3 ms e um máximo de 3,6 ms, enquanto o *HCCA* tem 11,7 ms e 12,4 ms para cenário de supervisão, 17,1 ms e 18,5 ms para o cenário de controle, respectivamente. Isto significa que a *GSC* tem uma redução na latência média de transmissão em torno de 71,79% quando comparado com o cenário de supervisão e de 80,70% para o cenário de controle.



**Figura 5.9:** Latência Média de Transmissão para *PER* Variável.



- **Resultado 3** - Outro resultado interessante é a taxa de sucesso de pacotes transmitidos com *PER* variando de 0% até 20% com passos de 5%. Na Tabela 5.5, podemos ver que o mecanismo *GSC* tem o desempenho similar ao *HCCA* para cenário de supervisão devido ao uso do quadro de *RTBlockAck* (não afetado pelo ruído) para a confirmação das mensagens transmitidas, mas em comparação com o cenário de controle do *HCCA*, o *GSC* tem um desempenho superior que garante a entrega de pacotes até ao destino. Pois, segundo [K.Cardoso & J.Rezende 2007] há dificuldades de se representar um ambiente físico real para redes locais sem fio, pois o *PER* médio varia para cada tipo de ambiente.

**Tabela 5.5:** Taxa de Sucesso dos Pacotes Transmitidos.

Packet Error Rate ( <i>PER</i> )	0%	5%	10%	15%	20%
Cenário de Supervisão e Controle ( <i>GSC</i> )	100.00	99.75	98.99	97.71	95.90
Cenário de Supervisão ( <i>HCCA</i> )	100.00	99.75	98.99	97.71	95.90
Cenário de Controle ( <i>HCCA</i> )	100.00	99.39	97.69	95.07	91.62

### Considerações Sobre Desempenho dos Mecanismos *GSC* e *HCCA*

Neste capítulo foi apresentada uma metodologia para análise de desempenho dos mecanismos de transmissão denominados *GSC* e *HCCA*. Escolheram-se dois cenários reais de automação industrial, chamados cenário de supervisão e cenário de controle. De posse desses cenários os testes de simulação foram realizados considerando dois ambientes distintos: sem ruídos e com ruídos. Definiram-se como cenário básico 20 estações de tempo real (sensores sem fio) que transmitiam um fluxo de dados de 40 bytes por estação para um dispositivo supervisor/controlador de planta industrial. Considera-se também que em determinadas situações na rede somente a metade dos sensores (50%) tinham dados a transmitir e em outra situação todos os sensores (100%) transmitiam seus dados normalmente. Isto se pode justificar através do uso de sensores distintos que tenham um *deadline* diferente, pois dessa forma estes podem amostrar o sinal utilizando valores múltiplos de *SI*, por exemplo. Na segunda etapa da simulação foram considerados os testes com ruídos independentes que afetam somente os dados, esta é uma forma simples e eficaz para analisar os mecanismos de transmissão. Pois, devem-se manter as mesmas condições de transmissão de quadros, lógico mantendo as características originais de ambos os mecanismos.

Os testes realizados para cenários industriais levaram em consideração o percentual do intervalo de serviço (tempo duração máxima do período livre de contenção), a latência média de transmissão da rede e a taxa de sucesso da entrega de mensagens de tempo real para ambientes ruidosos com taxas de perdas independente. Para todas as métricas o mecanismo proposto *GSC* obteve um melhor desempenho. Os resultados do *GSC* mostraram uma forte redução do *overhead* e aumento da economia da largura de banda da rede quando comparado com o mecanismo *HCCA*, demonstrando que a abordagem utilizando o *VTP* aliado à comunicação produtor/consumidor usado pelo *GSC* é mais descentralizada e flexível que o mecanismo de *polling* baseado em comunicação mestre/escravo utilizado pelo *HCCA*. Comprovando assim, que mecanismos que operam comunicação produtor/consumidor são mais adequados para aplicações que demandam tempo real distribuído. No próximo capítulo faz-se a conclusão o trabalho, discutindo as novas possibilidades para trabalhos futuros sobre o mecanismo proposto.

---

## Capítulo 6

# Conclusão e Trabalhos Futuros

---

Esta tese apresentou uma nova proposta para tratar comunicações de tempo real para redes industriais, através de um mecanismo denominado *Group Sequential Communication (GSC)*. Trata-se de um mecanismo que melhora o desempenho de redes industriais em ambiente sem fio quando comparado ao *HCCA* pertencente ao Padrão IEEE 802.11e.

A proposta utiliza os conceitos básicos do paradigma Produtor/Consumidor. As mensagens em tempo real são transmitidas em modo *Broadcast*, onde todas as estações podem recebê-las e aqueles que estão interessadas podem consumi-las. A técnica garante que transmissões de mensagens de tempo real sejam priorizadas sobre as mensagens oriundas de estações genéricas com *overhead* menor.

A ideia principal é permitir que as estações *GSC* possam enviar suas mensagens de forma sequencial, a fim de evitar colisões ao longo do período livre de contenção (*CFP*) em um ambiente compartilhado com outras estações genéricas (*DCF/EDCA*), usando um procedimento de passagem de ficha virtual (*VTP*) para organizar o acesso ao meio.

O mecanismo *GSC* sincroniza todas as estações utilizando um quadro de *Beacon*. Ao final do processo de transmissão, todas as estações *GSC* devem receber um quadro de reconhecimento de bloco (*RTBlockAck*), relatando quais as estações que obtiveram sucesso na transmissão, através do Coordenador Híbrido (*HC*) que está geralmente embutido no ponto de acesso (*AP*). Acionando o mecanismo de recuperação de erro (*Recovery Error Mechanism*), chamado de algoritmo de segunda chance (*Second Chance Algorithm*), que habilitará a retransmissão das mensagens de tempo real que não foram bem sucedidas na primeira tentativa.

Os experimentos mostraram resultados promissores a favor do *GSC* em cenários que simulam aplicações de redes industriais, devido à ausência de quadros *Polling*, *ACK* e *Qos Null*. Três métricas principais foram analisadas nas simulações: a duração máxima do período livre de contenção (*CFP*) ilustrado como percentual de utilização do intervalo de serviço (*SI*), a latência média de transmissão das mensagens e a taxa de sucesso das

mensagens transmitidas para os cenários de supervisão e controle industrial operando em ambiente livre e com ruídos.

Observou-se nas simulações que o *GSC* sempre obteve o melhor desempenho do que o *HCCA* em todos os indicadores (para pacotes pequenos de dados) em todos ambientes e cenários. Isto significa que o *GSC* pode cumprir os prazos de *deadline* com um desempenho superior e transmitir mais mensagens que o *HCCA* para o mesmo *CFP*. Estes resultados mostram também que há uma forte redução de *overhead* ocasionando uma economia na largura de banda na rede, quando comparado com o mecanismo *HCCA*.

Outra vantagem refere-se à abordagem Produtor/Consumidor utilizado no mecanismo *GSC* que é mais flexível e descentralizado que a abordagem Mestre/Escravo utilizado pelo *HCCA*. Além disto, a comunicação *GSC* é intrinsecamente *Broadcast*, tornando-se mais adequado para sistemas distribuídos em tempo real. Isto permite ao *GSC* operar de forma similar tanto para cenário de supervisão como de controle, diferentemente do *HCCA*.

Então, podemos concluir que o mecanismo do *GSC* é uma solução viável para lidar com o problema do não-determinismo de acesso ao meio para sistemas de tempo real aplicados a redes sem fio compatível com Padrão IEEE 802.11/11e, utilizando pacotes de tamanho pequeno, como nos cenários industriais.

Como trabalhos futuros pretende-se analisar o impacto gerado pelo mecanismo *GSC* nos tráfegos de estações genéricas (IEEE 802.11/802.11e) para o período de contenção quando o ambiente for susceptível a ruídos no canal de comunicação, além de sua validação via implementação de um protótipo do mecanismo proposto utilizando *hardware/software* aberto através do uso de um *kit* de desenvolvimento *wireless* para sistemas embarcados.

---

## Referências Bibliográficas

---

- Afonso, J. (2004), *Acesso Local sem Fios em Redes de Comunicação - Escalonamento de Tráfego de Tempo Real em Sistemas de Aquisição de Dados e Controlo*, Tese de doutorado, Universidade do Minho, Portugal. 337 p.
- Al-Karaki, J. N. & J. M. Chang (2004), ‘A Simple Distributed Access Control Scheme for Supporting QoS in IEEE 802.11 Wireless LANs’, *IEEE Wireless Communications and Networking Conference (WCNC)*.
- Bellalta, B., C. Cano, M. Oliver & M. Meo (2006), ‘Modeling the ieee 802.11 e edca for mac parameter optimization’.
- Bianchi, Giuseppe, Ilenia Tinnirello & Luca Scalia (2005), ‘Understanding 802.11e contention-based prioritization mechanisms and their coexistence with legacy 802.11 stations.’, *IEEE Network* **19**(4), 28–34.
- Carreiro, F. B., J. A. Fonseca & P. Pedreiras (2003), ‘Virtual Token-Passing Ethernet - VTPE’, *5th IFAC International Conference on Fieldbuses Systems and their Applications, FET 2003*.
- Casetti, C., C.-F Chiasserini, M. Fiore & M. Garetto (2005), ‘Notes on the Inefficiency on 802.11e HCCA’.
- Castano, J.G, Jens Lönnblad, Mikael Ekström & Ylva Bäcklund (2003), *Wireless industrial sensor monitoring based on bluetooth/spl trade/., em ‘IEEE International Conference on Industrial Informatics, Banff, Canada 2003 : Proceedings’*, pp. 65–72.
- Cena, G., I.C. Bertolotti, A. Valenzano & C. Zunino (2007), ‘Evaluation of Response Times in Industrial WLANs’, *IEEE Transactions on Industrial Informatics* **3**(3), 191–201.
- Cena, G., I.C. Bertolotti, A. Valenzano & C. Zunino (2008), ‘Industrial Applications of IEEE 802.11e WLANs’, *Factory Communication Systems, 2008. WFCS 2008. IEEE International Workshop on* pp. 129–138.

- Chen, Yeong-Sheng, Yuan-Wei Lee & Guang-Horng Huang (2008), 'Modified hcca mechanism for improving qos support in ieee 802.11e networks', pp. 1–7.
- Chittamuru, J., A. Ramanathan & M. Sinha (2005), Simulation of Point Coordination Function for IEEE 802.11 Wireless LAN using Glomosim, Relatório técnico. Disponível em: <[http://www-unix.ecs.umass.edu/~aramanat/CN/report697\\_new.doc](http://www-unix.ecs.umass.edu/~aramanat/CN/report697_new.doc)>. Acesso em: 10 jan. 2009. 18 p.
- Cicconetti, C., L. Lenzini, E. Mingozzi & G. Stea (2005a), 'A scheduling algorithm for providing real-time guarantees in 802.11e WLANs', *2nd International Conference on Broadband Networks, 2005* .
- Cicconetti, C., L. Lenzini, E. Mingozzi & G. Stea (2005b), 'A Software Architecture for Simulating IEEE 802.11e HCCA', *3rd Internet Performance, Simulation, Monitoring and Measurement - IPS-MoMe 2005* .
- Cloran, M. (2004), Simulation of IEEE 802.11 PCF function in GloMoSim, Relatório técnico, School of Electronic Engineering - Dublin City University. Disponível em: <<http://www.eeng.dcu.ie/~murphys/projects/MichaelCloran-FinalReport.pdf>>. Acesso em: 2 fev. 2009. 46 p.
- Decotignie, J. D. (2005), 'Ethernet-based real-time and industrial communications', *Proceedings of the IEEE, no. 6* **93**, 1102–1117.
- Demarch, Douglas (2007), Uma Proposta de Escalonamento Confiável para Redes Sem Fio Baseadas no Padrão IEEE 802.11/11e, Dissertação de mestrado, Universidade Federal de Santa Catarina. 88 p.
- Ebert, Jean-Pierre, Andreas Willig, Adam Wolisz & Tu Berlin (1999), 'A Gilbert-Elliot Bit Error Model and the Efficient Use in Packet Level Simulation'.
- Fall, K. (2001), 'Network Simulators and Related Links'. Disponível em: <<http://www.cs.berkeley.edu/~kfall/netsims.html>>. Acesso em: 17 nov. 2008.
- Farooq, Jahanzeb & Bilal Rauf (2006), 'An Overview of Wireless LAN Standards IEEE 802.11 and IEEE 802.11e'. 26 p.
- Gao, Deyun, Jianfei Cai & King Ngi Ngan (2005), 'Admission control in IEEE 802.11e wireless LANs', *Network, IEEE* **19**(4), 6–13.

- Garg, P., R. Doshi, R. Greene, M. Baker, M. Malek & Xiaoyan Cheng (2003), 'Using IEEE 802.11e MAC for QoS over wireless', *Performance, Computing, and Communications Conference, 2003. Conference Proceedings of the 2003 IEEE International* pp. 537–542.
- Gast, Matthew (2002), *802.11 Wireless Networks: The Definitive Guide Creating and Administering Wireless Networks*, O'Reilly Media, USA.
- Grilo, A., M. Macedo & M. Nunes (2003), 'A Scheduling Algorithm for QoS Support in IEEE 802.11E Networks', *IEEE Wireless Communications* **10**, 36–43.
- Hamidian, A. & U. Korner (2006), 'An enhancement to the IEEE 802.11e EDCA providing QoS guarantees', *Telecommunication Systems*, no. 2-3, **31**, 195–212.
- Hanley, G. (2005), A Study of the Performance of Voice over IP over IEEE 802.11 Wireless Local Area Networks, Dissertação de mestrado, Department of Computer Science - Faculty of Science - University College Dublin. 145 p.
- Hwang, G. H. & D.-H. Cho (2005), 'New access scheme for VoIP packets in IEEE 802.11e wireless LANs', *IEEE Commun. Lett. (USA)*, no. 7 **9**, 667–669.
- IEEE, 802.11 (1999a), *IEEE 802.11 Standard - Part 11 - Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*.
- IEEE, 802.11 (1999b), *IEEE 802.11a Standard - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications High-speed Physical Layer in the 5 GHz Band*.
- IEEE, 802.11 (1999c), *IEEE 802.11b Standard - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: Higher-Speed Physical Layer Extension in the 2.4 GHz Band*.
- IEEE, 802.11 (2003), *IEEE 802.11g Standard - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications Amendment 4: Further Higher Data Rate Extension in the 2.4 GHz Band*.
- IEEE, 802.11 (2004), *IEEE 802.11i Standard for Information Technology - Telecommunications and Information Exchange between Systems - Local and Metropolitan Area Networks - Specific Requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications - Amendment 6: Medium Access Control (MAC) Security Enhancements*.



- IEEE, 802.11 (2005), *IEEE 802.11e Standard for Information Technology - Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment 8: Medium Access Control (MAC) Quality of Service Enhancements*.
- IEEE, 802.11 (2007), *IEEE 802.11 Standard - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*.
- IEEE, 802.1D (2004), *IEEE 802.1D Standard for Local and Metropolitan Area Networks Media Access Control (MAC) Bridges*.
- K.Cardoso & J.Rezende (2007), 'Um Modelo de Markov Oculto para Representação de Perda de Pacotes em Redes 802.11 em Ambientes Indoor', *V Workshop em Desempenho de Sistemas Computacionais e de Comunicação - SBC* . pp. 653–672.
- Lee, Hyun-Jin & Jae-Hyun Kim (2006), 'A optimal CF-poll piggyback scheme in IEEE 802.11e HCCA', *Advanced Communication Technology, 2006. ICACT 2006. The 8th International Conference* **3**, 1953–1959.
- Lee, Suk, Kyoung Nam Ha, Jee Hun Park & Kyung Chang Lee (2005), 'Ndis-based virtual polling algorithm of ieee 802.11b for guaranteeing the real-time requirements'.
- Li, Tianji, Qiang Ni, Thierry Turletti & Yang Xiao (2005), 'Performance Analysis of the IEEE 802.11e Block ACK Scheme in a Noisy Channel', *2nd International Conference on Broadband Networks (BROADNETS 2005), 3-7 October 2005, Boston, Massachusetts, USA* pp. 551–557.
- Lim, L. W., P. Y. Tan, C. Apichaichalermwongse, K. Ando & Y. Harada (2004), 'A QoS Scheduler for IEEE 802.11e WLANs', *Consumer Communications and Networking Conference, CCNC 2004* pp. 199–204.
- Lo, Shou-Chih, Guanling Lee & Wen-Tsuen Chen (2003), 'An efficient multipolling mechanism for IEEE 802.11 wireless LANs', *IEEE Transactions on Computers* **52**(6), 764–778.
- Mangold, S., Sunghyun Choi, G.R. Hiertz, O. Klein & B. Walke (2003), 'Analysis of IEEE 802.11e for QoS support in wireless LANs', *IEEE Wireless Communications* **10**(6), 40–50.
- Milhim, Ashraf D. & Yaw-Chung Chen (2007), 'An Adaptive Polling Scheme to Improve Voice Transmission over Wireless LANs', *ACS/IEEE International Conference on Computer Systems and Applications* **0**, 146–152.



- Moraes, R., F. Vasques, P. Portugal & J.A. Fonseca (2007), 'VTP-CSMA: A Virtual Token Passing Approach for Real-Time Communication in IEEE 802.11 Wireless Networks', *IEEE Transactions on Industrial Informatics* **3**, 215–224.
- Moraes, R., P. Portugal & F. Vasques (2006), 'Simulation analysis of the IEEE 802.11e edca protocol for an industrially-relevant real-time communication scenario', *11th IEEE International Conference on Emerging Technologies and Factory Automation - ETFA*.
- Moyne, James R. & Dawn M. Tilbury (2007), 'The Emergence of Industrial Control Networks for Manufacturing Control, Diagnostics, and Safety Data', *Proceedings of the IEEE* **95**(1), 29–47.
- Müller, A. (2006), Investigation of QoS Support of Wireless MAC Protocols, Dissertação de mestrado, Department of Informatics - University of Zurich - Switzerland. 119p.
- Ni, Qiang (2005), 'Performance analysis and enhancements for IEEE 802.11e wireless networks', *IEEE Network* **19**(4), 21–27.
- Pavón, J. & S. Shankar (2004), 'Impact of Frame Size, Number of Stations and Mobility on the Throughput Performance of IEEE 802.11e', *IEEE Wireless Communications and Networking Conference, V. 2*. pp. 789–795.
- Project, Network Simulator (Online - accessed 26 June 2008), 'NS-2 - Network Simulation Software'. <http://www.isi.edu/nsnam/ns/>.
- Sauter, T. & F. Vasques (2006), 'Editorial: Special section on communication in automation', *IEEE Transactions on Industrial Informatics* **2**(2), 73 – 77.
- Skyrianoglou, Dimitris, Nikos I. Passas & Apostolis K. Salkintzis (2006), 'Arrow: An efficient traffic scheduling algorithm for IEEE 802.11e hcca', *IEEE Transactions on Wireless Communications* **5**(12), 3558–3567.
- Sobrinho, J. & A. Krishnakumar (1999), 'Quality-of-service in ad hoc carrier sense multiple access wireless networks', *IEEE J. Sel. Areas Commun. (USA)*, no. 8 **17**, 1353–1368.
- Son, J., I. Lee & H. Yoo (2005), 'An Effective Polling Scheme for IEEE 802.11e', *IEICE Transactions on Communications*, no. 12 **E88-B**, 4690–4693.

- Tanenbaum, Andrew (2003), *Redes de Computadores*, Tradução da 4ª edição, Elsevier, Rio de Janeiro.
- Trsek, H., J. Jasperneite & S.P. Karanam (2006), 'A Simulation Case Study of the new IEEE 802.11e HCCA mechanism in Industrial Wireless Networks', *Emerging Technologies and Factory Automation, 2006. ETFA '06. IEEE Conference on* pp. 921–928.
- Vaidya, Nitin H., Paramvir Bahl & Seema Gupta (2000), 'Distributed fair scheduling in a wireless lan', *ACM The Sixth Annual Conference Mobile Computing and Networking* pp. 167–178.
- Viegas, Carlos M. (2009), *Análise de Desempenho de Estratégias de Retransmissão para o Mecanismo HCCA do Padrão de Redes Sem Fio IEEE 802.11e*, Dissertação de mestrado, Universidade Federal do Rio Grande do Norte. 74 p.
- Vitturi, S., I. Carreras, D. Miorandi, L. Schenato & A. Sona (2007), 'Experimental Evaluation of an Industrial Application Layer Protocol Over Wireless Systems', *IEEE Transactions on Industrial Informatics* **3**(4), 275–288.
- Viégas Jr, R., L. A. Guedes & F. Vasques (2009), 'A Proposal of Real-Time Publish-Subscribe Scheme Compatible with 802.11e Wireless Networks.', *35th Annual Conference of the IEEE Industrial Electronics Society - IECON 2009, Porto - Portugal*.
- Viégas Jr, R., R. Moraes, L. A. Guedes & F. Vasques (2007), 'GSC: A Real-time Communication Scheme for IEEE802.11e Industrial Systems.', *7th IFAC International Conference on Fieldbuses and Networks in Industrial and Embedded Systems -FET 2007, Toulouse - França*.
- Viégas Jr, R., R. Valentim, D. Texeira & L. A. Guedes (2006a), 'Analysis of Protocols to Ethernet Automation Networks.', *IEEE International Joint Conference - SICE - ICCAS 2006, Busan - Coréia*.
- Viégas Jr, R., R. Valentim, D. Texeira & L. A. Guedes (2006b), 'Performance Measurements of Protocols to Ethernet Real-Time Applications.', *11th IEEE International Conference on Emerging Technologies and Factory Automation - ETFA 2006, Praga - Rep. Theca*.
- Willig, A. (2008), 'Recent and Emerging Topics in Wireless Industrial Communications: A Selection', *Industrial Informatics, IEEE Transactions on* **4**(2), 102–124.

- Willig, A., K. Matheus & A. Wolisz (2005), 'Wireless Technology in Industrial Networks', *Proceedings of the IEEE* **93**(6), 1130–1151.
- Willig, A., M. Kubisch & A. Wolisz (2001), Measurements and Stochastic Modeling of a Wireless Link in an Industrial Environment, Relatório técnico, Telecommunication Networks Group - Technical University Berlin. 74 p.
- Work, SpectraLinkWireless (2006), 'Spectralink voice priority quality of service for voice traffic in wireless lans', *available at [www.spectralink.com](http://www.spectralink.com)* .
- Xiao, Y. & H. Li (2004), 'Evaluation of Distributed Admission Control for the IEEE 802.11e EDCA.', *IEEE Communication Magazine* **42**(9), S20–S24.
- Xiao, Yang (2004), 'IEEE 802.11e: QoS provisioning at the MAC layer.', *IEEE Wireless Communications* **11**(3), 72– 79.
- Yang, L. (2005), 'Enhanced HCCA for Real-Time Traffic with QoS in IEEE 802.11 based networks', *The IEE International Workshop on Intelligent Environments* .
- Ye, H., G.-C. Walsh & L. Bushnell (2000), 'Wireless Local Area Networks in the Manufacturing Industry', *Proceedings of the American Control Conference Chicago* .
- Ye, R. (2006), Performance Analyses of HCCA Polling and Scheduling Schemes in IEEE802.11e Standard, Relatório técnico, Nanyang Technological University.