Universidade Estadual de Campinas
FACULDADE DE ENGENHARIA ELÉTRICA E DE COMPUTAÇÃO
DEPARTAMENTO DE TELEMÁTICA

MÚLTIPLO ACESSO EM REDES DE COMUNICAÇÃO SEM FIO

Dissertação apresentada à Faculdade de Engenharia Elétrica e de Computação (FEEC) da Universidade Estadual de Campinas (UNICAMP) como parte dos requisitos exigidos para a obtenção do título de Mestre em Engenharia Elétrica

Rosanna Mara Rocha Silveira
Bacharel em Ciência da Computação pela Universidade Federal de Minas Gerais

Orientador: **Prof. Dr. Ivanil Sebastião Bonatti**
DT – FEEC – Unicamp

Banca examinadora:

Prof. Dr. Ivanil Sebastião Bonatti - Presidente — DT – FEEC – Unicamp
Dra. Eunice Luvizotto Medina Pissolato — Fundação CPqD
Prof. Dr. Amauri Lopes — DECOM – FEEC – Unicamp
Prof. Dr. Paulo Cardieri — DECOM – FEEC – Unicamp

Campinas- SP, Janeiro 2003

FICHA CATALOGRÁFICA ELABORADA PELA
BIBLIOTECA DA ÁREA DE ENGENHARIA - BAE - UNICAMP

Si39m

Silveira, Rosanna Mara Rocha
Múltiplo acesso em redes de comunicação sem fio / Rosanna Mara Rocha Silveira.--Campinas, SP: [s.n.], 2003.

Orientador: Ivanil Sebastião Bonatti.
Dissertação (mestrado) - Universidade Estadual de Campinas, Faculdade de Engenharia Elétrica e de Computação.

1. Sistemas de comunicação sem fio. 2. Sistemas de comunicação sem fio – Controle de acesso. 3. Simulação (Computadores). 4. Modo de transferência assíncrono. I. Bonatti, Ivanil Sebastião. II. Universidade Estadual de Campinas. Faculdade de Engenharia Elétrica e de Computação. III. Título.

RESUMO

Este trabalho analisa os mecanismos de múltiplo acesso em redes de comunicação sem fio e de faixa larga usando tecnologia ATM (*Asynchronous Transfer Mode*). O enlace de *uplink*, do terminal para a estação rádio-base, usa a técnica TDMA (*Time Division Multiple Access*) combinando janelas de contenção e janelas de reserva. O enlace de *downlink*, da estação rádio-base para os terminais, usa técnica TDM (*Time Division Multiplexing*) em modo de difusão. A metodologia de análise é baseada em simulações de Monte Carlo e de evento discreto. Os mecanismos analisados são: algoritmos de resolução de contenção; algoritmos para agendamento dos pedidos de reserva; *piggybacking*; algoritmos para alocação dinâmica das janelas de contenção e das janelas de reserva; tamanho dos quadros, tamanho das janelas de contenção; e prioridade de atendimento para terminais com tráfego em tempo real. Através da análise comparativa de desempenho dos diversos mecanismos envolvidos no múltiplo acesso, uma proposta, agregando todos os componentes, é avaliada para tráfegos em tempo real e não em tempo real, poissonianos ou não.

Palavras-chave: Redes de comunicação sem fio, redes ATM sem fio, múltiplo acesso, controle de acesso ao meio, simulação de evento discreto.

ABSTRACT

This work analyses the mechanisms of multiple access in broadband wireless communication networks using ATM (Asynchronous Transfer Mode) technology. The uplink - from terminals to the base station - uses the TDMA (Time Division Multiple Access) technique with contention and reservation slots. The downlink - from the base station to the terminals - uses the TDM (Time Division Multiplexing) in a broadcasting way. The methodology of analysis is based on Monte Carlo and discrete event simulations. The mechanisms under analysis are: contention resolution algorithms; algorithms to schedule reservation requests; piggybacking; dynamic allocation algorithms for contention and reservation slots; frame length and contention slot length; and priority for real time traffic. By means of the comparative analysis of performance of the several mechanisms involved in the multiple access, a proposal, joining all the components, is evaluated for real time traffic and non-real time traffic, with Poisson and non-Poisson traffics.

Key words: Wireless communication networks, wireless ATM networks, multiple access, medium access control, discrete event simulation.

AGRADECIMENTOS

À minha mãe, Mirtes, pela razão. Com sua determinação e fortaleza me ensinou a decidir e a persistir.

Ao meu pai, *in memorian*, Almir, pela emoção. Com sua calma e humildade me ensinou a dedicar e a superar.

À minha filha e companheira, Isadora, pela compreensão e cumplicidade.

Ao meu namorado e incentivador, Brito, pelo apoio e por ter partilhado comigo dos momentos difíceis e dos momentos de alegria durante toda essa trajetória.

Ao meu orientador, Prof. Ivanil, que com sua competência, aliada à gentileza, carinho, e solicitude na tarefa de orientação, me convenceu de que com disciplina, humildade e sistemática é possível alcançar um objetivo.

Ao Inatel, pela oportunidade de participar desse programa de capacitação.

Aos colegas do Setor de Editoração do Inatel, pelo apoio na edição e impressão deste trabalho.

A Deus, pela constante presença em todos os momentos da minha vida, iluminando, protegendo e conduzindo os meus passos.

ACRÔNIMOS

AAL	ATM Adaptation Layer
ABR	Available Bit Rate
ASK	Amplitude Shift Keying
ATM	Asynchronous Transfer Mode
BPSK	Binary Phase Shift Keying
BS	Base Station
BWLL	Broadband Wireless Local Loop
CAC	Connection Admission Control
CBR	Constant Bit Rate
CDMA	Code Division Multiple Access
CLP	Cell Loss Priority
CRA	Collision Resolution Algorithm
CRC	Cyclic Redundancy Check
CRI	Collision Resolution Interval
CRPMA/DS	Combination of Reservation and Polling Multiple Access with Distributed Scheduling
CSI	Channel State Information
DAVIC	Digital Audio Visual Council
DOCSIS	Data Over Cable Service Interface Specifications
DPRMA	Dynamic Packet Reservation Multiple Access
DPSK	Differential Phase Shift Keying
DQPSK	Differential Quadrature Phase Shift Keying
DQRUMA	Distributed Queueing Request Update Multiple Access
DSA	Dynamic Slot Assignment
EBR	Extended Bandwidth Request
EC	European Community
FDD	Frequency Division Duplex
FDMA	Frequency Division Multiple Access
FEC	Foward Error Correction
FSK	Frequency Shift Keying
GMSK	Gaussian Minimum Shift Keying
HFC	Hybrid Fiber Coax
IP	Internet Protocol
LOS	Line Of Sight
MAC	Medium Access Control
MASCARA	Mobile Access Scheme based on Contention and Reservation for ATM
MBS	Maximum Burst Size
MCNS	Multimedia Cable Network System
MPDU	MAC Protocol Data Units
MS	Mobile Station
MSC	Mobile Switching Center
MSK	Minimum Shift Keying
MUX	Multiplex
nrt	non real time
OAM	Operation And Maintenance
PCR	Peak Cell Rate
PDU	Protocol Data Unit

PRADOS	Priority Regulated Allocation Delay Oriented Scheduling
PRMA	Packet Reservation Multiple Access
PRMA/DA	Packet Reservation Multiple Access with Dynamic Allocation
PSK	Phase Shift Keying
PSTN	Public Switched Telephone Network
QAM	Quadrature Amplitude Modulation
QoS	Quality Of Service
QPSK	Quadrature Phase Shift Keying
R-Aloha	Reservation Aloha
RF	Radio Frequency
rt	real time
SCAMA	Synergistic Channel Adaptive Multiple Access
SCR	Sustainable Cell Rate
TDD	Time Division Duplex
TDM	Time Division Multiplexing
TDMA	Time Division Multiple Access
UBR	Unspecified Bit Rate
VBR	Variable Bit Rate
WAND	W-ATM Network Demonstrator
WLL	Wireless Local Loop

ÍNDICE

INTRODUÇÃO

A tecnologia ATM (*Asynchronous Transfer Mode*) é apropriada para o fornecimento de telecomunicações multiserviço e de banda larga. O acesso via rádio permite a mobilidade dos usuários da rede. Assim, a rede ATM sem fio representa uma importante e oportuna solução para o acesso móvel de banda larga nas redes de telecomunicações.

O espectro de freqüências é um recurso escasso e, portanto, o enlace de rádio deve ser usado da maneira mais eficiente possível. A capacidade e a qualidade de serviço de um sistema ATM sem fio dependem da eficácia do protocolo de múltiplo acesso. Os tráfegos CBR (*Constant Bit Rate*), como telefonia, VBR (*Variable Bit Rate*), como videoconferência, e ABR (*Available Bit Rate*), como dados de arquivo, têm diferentes requisitos de serviço em termos de atraso, tolerância de perda e vazão. Combinar esses diferentes requisitos, de maneira harmônica, com razoável qualidade de serviço e maximizar a utilização da banda disponível é uma tarefa difícil e desafiadora. Os protocolos de múltiplo acesso procuram partilhar o espectro, entre os vários usuários, de maneira equilibrada e eficiente.

O múltiplo acesso (compartilhamento do canal de comunicação) é implementado na camada de enlace de dados do modelo OSI. O protocolo MAC (*Medium Access Control*) de cada terminal e da estação rádio-base interage com a camada física, em rádio-freqüência, e com a camada ATM. A célula é a unidade de transporte do ATM e tem tamanho fixo de 53 bytes. A célula é convertida em bloco-rádio (acresce-se *overhead* da camada MAC) para transmissão, e, uma vez recebido, é reconvertido em célula ATM.

A estrutura típica de transporte da célula sobre a interface rádio é ilustrada na Figura 1. A célula ATM é encapsulada pelo cabeçalho e rodapé do protocolo utilizado. O cabeçalho contém *bits* de sincronização, um campo que indica o número de janelas solicitadas pelo terminal e campos de controle, enquanto o rodapé contém um campo de verificação de erro. O campo do cabeçalho que indica o número de janelas solicitadas pelo terminal é um campo importante na operação do protocolo, pois ele é usado pelo terminal a cada quadro, para informar a solicitação corrente de banda à estação rádio-base.

cabeçalho do protocolo	cabeçalho da célula ATM	48 bytes de informação	rodapé do protocolo

Figura 1 – Formato da célula na interface rádio.

A estação rádio-base comunica com os terminais em modo de difusão (*broadcasting*), através do canal de *downlink* e os terminais comunicam com a estação rádio-base através do canal de *uplink* utilizando um protocolo de múltiplo acesso (vide Figura 2). Os terminais não se comunicam entre si, mas, apenas com a estação rádio-base. É a estação rádio-base que define a cadência com que o tráfego é transmitido através do meio sem fio.

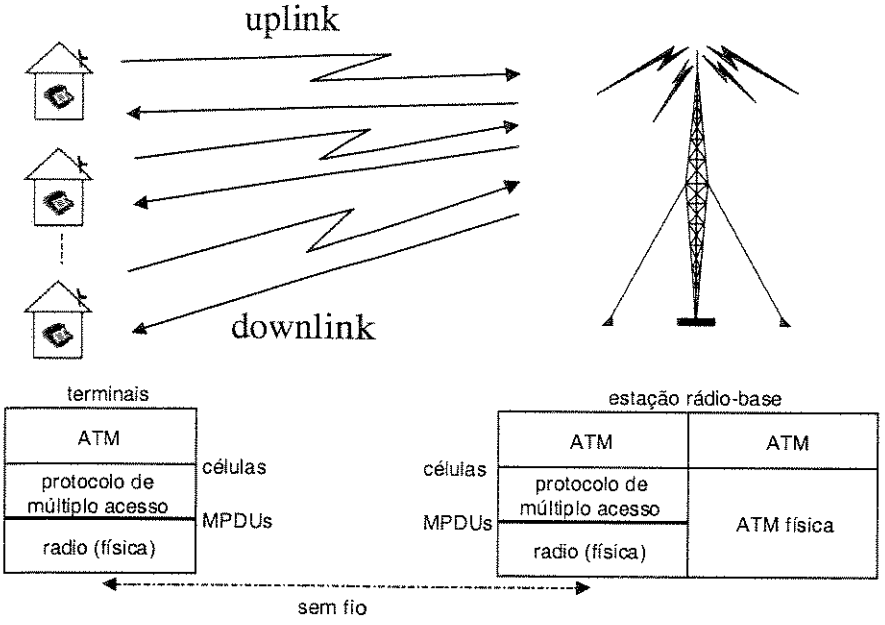


Figura 2 – Enlace sem fio.

Os canais de *uplink* e *downlink* podem usar uma freqüência específica para cada um, FDD (*Frequency Division Duplex*) ou, uma única freqüência compartilhada no tempo, ora para *uplink*, ora para *downlink*, esquema TDD (*Time Division Duplex*). Este trabalho não aborda questões particulares que distingüem esses dois esquemas e, simplificadamente, considera que os quadros de *uplink* sucedem-se uns aos outros e as respostas às solicitações de reserva são realimentadas prontamente ao final do quadro.

A técnica de múltiplo acesso utilizada no *uplink* é a TDMA (*Time Division Multiple Access*), na qual o tempo é dividido em quadros, que são subdivididos em janelas. A duração de uma janela é igual ao tempo necessário para se transmitir uma célula ATM mais o *overhead* da camada MAC. As janelas de contenção consideradas são iguais às janelas de reserva (sem mini) ou um quarto da janela de reserva (mini-janelas).

O canal de *uplink* é usado pelos terminais para enviar solicitações e pacotes de informação à estação rádio-base, em modo de contenção e reserva. Os terminais enviam solicitações à estação rádio-base para pacotes que chegam em seus *buffers* vazios. Para pacotes que chegam nos *buffers*, onde já existem pacotes armazenados, as solicitações são livres de colisão e utilizam a informação de *piggybacking*.

O canal de *downlink* é usado pela estação rádio-base para enviar reconhecimento e pacotes de informação aos terminais, em modo programado. A estação rádio-base controla de forma centralizada as transmissões e retransmissões dos terminais e é ela que determina como as janelas de cada quadro são alocadas aos terminais.

Este trabalho aborda os diferentes mecanismos envolvidos nos protocolos de múltiplo acesso com o intuito de avaliá-los individualmente e compô-los em um esquema de múltiplo acesso.

Dentre as questões abordadas neste trabalho, pode-se citar:

- qual o algoritmo de resolução de contenção a utilizar;
- como agendar as janelas de reserva do quadro;

- se o uso da informação de *piggybacking* melhora o desempenho apenas para cargas altas;
- como propor novas janelas de contenção no quadro e como tratá-las simultaneamente;
- se o quadro deve ser de tamanho fixo ou variável;
- como tratar o tráfego em tempo real e
- qual deve ser o tamanho da janela de contenção.

A metodologia usada no trabalho para a análise do protocolo de múltiplo acesso é a simulação de evento discreto e de Monte Carlo com tráfegos poissoniano, em *batch*, uniformes ou não. Em geral, as simulações são usadas com três objetivos distintos: para avaliar a precisão de modelos analíticos; para comparar estratégias distintas; ou para avaliar desempenho de um dado protocolo. A segunda abordagem é usada neste trabalho. Desta forma, uma série de simplificações no protocolo de múltiplo acesso pode ser realizada para facilitar as simulações, desde que não se altere qualitativamente o sentido das comparações. Por exemplo: o canal foi considerado sem erro; o *downlink* e os mecanismos de retransmissão de células erradas não foram considerados, pois afetam igualmente os mecanismos analisados.

A redação da dissertação está organizada da seguinte forma.

No Capítulo I são descritos, de forma abreviada, as redes de comunicação sem fio e alguns dos protocolos de múltiplo acesso da literatura.

No Capítulo II, os algoritmos envolvidos no múltiplo acesso são descritos: resolução de contenção; agendamento de reserva; alocação dinâmica das janelas de contenção e de reserva; e prioridade de atendimento para terminais com tráfego em tempo real.

No Capítulo III são apresentados os modelos de simulação desenvolvidos para análise comparativa de desempenho dos diversos mecanismos que compõem o protocolo de múltiplo acesso.

No Capítulo IV, cada um dos mecanismos foi avaliado através de resultados de simulação obtidos pelos modelos desenvolvidos no Capítulo III. Em função das conclusões preliminares sobre cada um dos mecanismos, uma proposta compondo as melhores escolhas foi realizada. A proposta foi avaliada para terminais com tráfegos de diferentes prioridades e distribuições estatísticas.

As conclusões do trabalho são apresentadas no último capítulo.

CAPÍTULO I - CENÁRIO

O par metálico foi a principal forma de conexão dos usuários às redes de serviços, e, apesar dos avanços tecnológicos alcançados, sempre se caracterizou por custos elevados e prazos longos de implantação. Com o desenvolvimento dos sistemas de comunicação sem fio, a expansão das redes de telecomunicações pode ocorrer de uma forma rápida e com um custo menor de implantação e manutenção. Dentre os sistemas de comunicação sem fio, podem-se citar os enlaces de microondas, os satélites de comunicação, os sistemas celulares e as redes de acesso fixo sem fio.

I.1 – APLICAÇÕES

I.1.1 – Enlace de microondas

Existem numerosas aplicações que utilizam o espectro de freqüências e cada serviço de rádio utiliza uma porção desse espectro. As freqüências são alocadas para especificar serviços com similaridade em termos de poder de irradiação e características de interferência. Esse plano de alocação pode ser feito para que as interferências sejam minimizadas e a demanda de tráfego seja satisfeita. O aumento na demanda dos serviços de radiodifusão e de telecomunicações trouxe um congestionamento e uma saturação nas faixas de freqüências mais baixas. Esse aumento estimulou a utilização de freqüências cada vez mais elevadas. As freqüências de microondas têm sido empregadas em diversos sistemas de comunicações: na telefonia à longa distância, em telefones celulares, na transmissão de dados e na transmissão de programas de televisão. A Tabela 1.1 mostra o uso do espectro de freqüências [Ma99].

Freqüência	Comprimento de Onda	Aplicação
10 KHz	30 km	Comunicações submarinas de baixa freqüência
100 KHz	3 km	Radiodifusão em ondas longas
1 MHz	300 m	Radiodifusão AM
10 MHz	30 m	Radiodifusão em ondas curtas (ionosfera)
100 MHz	3 m	Radiodifusão FM
150 MHz	2 m	Rádio móvel
300 MHz	1 m	Difusão de TV UHF e enlaces de microondas UHF ponto a ponto
3-60 GHz	10 cm – 0.5 mm	Enlaces de microondas
230 THz	1300 nm	Fibras ópticas
420-750 THz	400-700 nm	Luz visível
1000 THz	300 pm	Raios X

Tabela 1.1 – Uso do espectro de rádio-freqüências.

Os enlaces de microondas são os sistemas portadores de informações que operam na faixa de freqüências de 300 MHz a 60 GHz, correspondendo a comprimentos de onda que variam de 1m a 0.5 mm, respectivamente. Nesses sistemas, o sinal irradiado por uma antena é captado por uma outra que deve ser visível a partir da primeira, ou seja, deve haver visada direta entre as antenas transmissora e receptora. Os enlaces empregam altas torres com antenas tipo lente ou refletor como estações intermediárias dispostas ao longo do percurso. As estações intermediá-

rias, existentes entre a origem e o destino, são chamadas de repetidoras. Essas estações recebem o sinal de microondas, amplificam esse sinal e o retransmitem à repetidora seguinte, com o objetivo de garantir a condição de visada direta entre as antenas e compensar a atenuação. Um outro meio de comunicação por microondas é o uso de satélites como estações repetidoras.

Os sinais telefônicos ou de dados são agrupados por um equipamento multiplexador MUX (*Multiplex*), passam a ocupar uma faixa de freqüência bem definida e formam o que se denomina banda básica. O sinal de banda básica do MUX é levado, através de um cabo, até uma estação rádio, onde um transmissor entrega à antena um sinal de microondas que carrega as informações de banda básica. O sinal de microondas é então irradiado ao destino. No destino, o sinal é recebido e a banda básica é separada em canais por um outro equipamento MUX e encaminhados aos equipamentos convenientes, dependendo da informação que transportam. No caso de transmissão de programas de televisão, existe um centro de TV que, a partir do sinal fornecido pela emissora local, forma a banda básica de televisão e a entrega ao transmissor de microondas da estação rádio. No destino, a banda básica de TV será endereçada, através de outro centro de TV, à emissora local. A ligação entre as emissoras locais e os centros de televisão pode ser realizada via um outro sistema rádio ou via cabo.

O conjunto dos meios envolvidos para a transmissão de informação via rádio num sentido é chamado de canal de radiofreqüência, RF (*Radio Frequency*). As ligações de microondas operam em modo bidirecional. Na ligação bidirecional existem dois canais rádio envolvidos, um em cada sentido, que operam em freqüências distintas para a transmissão e a recepção. O transmissor e o receptor de cada estação rádio estão ligados a uma mesma antena, que serve tanto para receber como para transmitir o sinal de microondas.
Quanto à transmissão de sinais telefônicos, podem-se transportar vários canais de voz através de um único canal rádio com a utilização do equipamento MUX. Entretanto, como os sinais de televisão ocupam uma faixa de freqüência muito grande, é necessário um canal RF para transportar cada sinal de televisão. Em geral, em um sistema de microondas existem vários canais rádio entre as localidades de origem e destino, cada um operando numa determinada freqüência, com transmissores e receptores distintos, mas fazendo uso da mesma antena. Alguns desses canais de RF se destinam exclusivamente à transmissão de sinais provenientes do MUX, e outros, ao tráfego de sinais de televisão.

I.1.2 – Comunicação por satélite

Um sistema de comunicação por satélite consiste de um satélite colocado no espaço e de diversas estações terrenas. A estação terrena é responsável por receber o sinal do usuário, através da rede de comunicação terrestre, processá-lo e transmiti-lo ao satélite através da modulação de uma portadora.
O satélite é uma estação repetidora de microondas colocado no espaço. Sua função é receber o sinal oriundo da estação terrena (do *uplink*), amplificá-lo e retransmiti-lo (pelo *downlink*) de volta à Terra em uma faixa de freqüência diferente daquela em que o recebeu, de modo a evitar interferências. O sinal proveniente do satélite é então recebido pela estação terrena de destino, que se encarrega de processá-lo e adequá-lo para a transmissão ao usuário final, através da rede terrestre. Os satélites podem oferecer meios de comunicação das seguintes categorias: ponto a ponto, ponto a multiponto, multiponto a ponto e multiponto a multiponto.

Os sistemas de comunicação por satélite eram utilizados basicamente para transmissão de sinais de telefonia e TV, apresentando algumas vantagens sobre as demais formas disponíveis

de comunicação, tais como: facilidade para alcançar regiões remotas; facilidade para transmissões em *broadcasting*; flexibilidade para crescimento do sistema (estações terrenas); alta capacidade de transmissão. Com o decorrer de seu uso e com o incremento da transmissão de telefonia e TV, utilizando-se técnicas digitais, outras aplicações foram surgindo, dentre as quais podem-se destacar as redes de comunicação de dados.

Os satélites estão em órbita ao redor da Terra e podem ser classificados, de geo-estacionários e não geo-estacionários, de acordo com a sua órbita. Os geo-estacionários estão a uma distância que varia entre 35.786 e 41.680 km das estações terrenas. Também conhecidos como síncronos, viajam ao redor da Terra na mesma velocidade da rotação do planeta, e portanto, estão parados em relação a um ponto da superfície da Terra. Esta condição permite aos satélites serem mantidos sob a mira de antenas na Terra, fixas e alinhadas com eles. Os satélites não geo-estacionários são utilizados em órbita baixa e passam sobre os pólos a uma altitude de cerca de 1.000 km. Isso resulta em uma redução da área útil, ou seja, só ficam visíveis para a estação terrena durante um pequeno período de tempo, e, para garantir uma cobertura eficiente, é necessária a existência de um número elevado de satélites.
O problema do atraso na comunicação por satélite, causado pela longa distância entre a estação terrena e o satélite não pode ser negligenciado. A existência desse atraso, da ordem de 250ms, faz com que a ocorrência de eco seja um sério inconveniente para a comunicação de voz, implicando na necessidade do uso de equipamentos supressores de eco.

I.1.3 – Sistema celular

Um sistema celular típico é constituído de três componentes básicos, além das conexões entre eles: a central de comutação celular MSC (*Mobile Switching Center*), a estação rádio-base BS (*Base Station*) e o terminal móvel MS (*Mobile Station*).
A central de comutação celular é responsável pelas funções operacionais da rede móvel: a comutação, o controle, a tarifação e a conexão com a rede fixa.
A estação rádio-base realiza a interface entre a central de comutação e os terminais móveis e estabelece o enlace radio com o terminal móvel dentro da área de cobertura de uma célula, área na qual o sinal de uma estação rádio-base é adequadamente recebido. As ligações entre BS e MSC são feitas, em geral, via rádio, sendo também possíveis conexões por troncos de linhas físicas utilizando fibras ópticas ou cabos coaxiais.
O terminal móvel é a unidade que possui uma antena e um transceptor, e se conecta à estação rádio-base via rádio.

A idéia básica do sistema celular é o reuso da freqüência, em que o mesmo conjunto de canais pode ser utilizado em diferentes áreas geográficas, suficientemente distantes umas das outras, de forma que a interferência co-canal (canal de mesmo número) esteja dentro de limites toleráveis. Estas áreas são subdivididas em áreas menores, as células, que utilizam um subconjunto de canais designados de acordo com o espectro disponível. A representação da estrutura das células é hexagonal indicando que são colocadas lado a lado, evitando os problemas de superposição. Contudo, o formato real das células é amorfo e a sobreposição é necessária para que o *handoff* seja possível. O tamanho da célula é definido através da potência dos transmissores do terminal móvel e pela atenuação do sinal. Cada célula possui uma estação rádio-base e antenas direcionais que são responsáveis por oferecer cobertura aos vários setores da célula, que são, tipicamente, três.

Quando é estabelecida uma chamada entre um terminal móvel e um terminal fixo, a transmissão dos dados será via rádio para a BS situada mais próxima da MS. Em seguida, os dados

chegam à MSC e a chamada é comutada para a rede telefônica onde o terminal fixo está conectado. Da mesma forma, um terminal fixo acessa uma MS, através de busca e comutação automática processadas pela MSC.
À medida que o terminal móvel se distancia da BS, o sinal fica cada vez mais fraco até chegar a uma distância em que deve ser trocado o canal da célula de origem para um outro canal na célula vizinha. Este processo de troca automática é chamado de *handoff* (ou *handover*), e é bem sucedido se a célula vizinha tem canais livres disponíveis. Se não houver canal disponível, a conexão é perdida. O canal utilizado na célula de origem torna-se disponível para outra alocação. Um grande número de estações rádio-base é controlado pela MSC. Um terminal móvel é registrado em uma MSC, e quando ele se desloca para uma outra região servida por outra MSC, um novo registro deve ser feito pela nova MSC. Esse processo de registro automático é chamado de *roaming*.

I.1.4 – Rede de acesso fixo sem fio

O resultado de um projeto utilizando redes de acesso com fio é muito sensível às alterações das tendências de crescimento da população e conseqüente demanda de terminais. Além das redes de acesso com fio serem obras praticamente irremovíveis após sua implantação, o tempo de maturação do projeto, entre a coleta de dados e sua conclusão, é longo, sendo freqüente seu descompasso com a distribuição geográfica da demanda quando da sua conclusão. Esta falta de flexibilidade é freqüentemente responsável pela escassez de facilidades da rede em um local que se desenvolveu repentinamente e pela ociosidade de cabos em locais que pararam de crescer. Sendo uma atividade mobilizadora de recursos humanos desde a mão-de-obra braçal, na construção de galerias, dutos, puxamento e emenda de cabos, até a qualificação especializada na execução do projeto, a rede com fio tem-se mostrado, ao longo dos anos, uma atividade de difícil gerenciamento de prazos, de custos e de qualidade.

Redes de acesso fixo sem fio são empregadas nas redes telefônicas públicas comutadas PSTN (*Public Switched Telephone Network*), como um método opcional de acesso. A ligação dos terminais à PSTN é realizada através de sinais de rádio e é um substituto para o par simétrico entre a central comutadora local e a casa do assinante. As redes de acesso fixo sem fio costumam ser implementadas como uma alternativa para que as operadoras de rede fixa diminuam seus custos com a infra-estrutura da rede, objetivando uma redução dos custos de manutenção e também diminuam seus prazos para ligar os terminais às centrais, que são menores do que os prazos para a construção de uma rede metálica, geralmente subterrânea. Além da redução de custos e prazos, as redes de acesso fixo sem fio facilitam o rápido crescimento da rede e permitem o acesso à central comutadora em locais de difícil colocação de cabos.

O desenvolvimento da telefonia móvel celular estimulou as pesquisas, o desenvolvimento de novas tecnologias e uma compreensão mais profunda da propagação de rádio-freqüências nos espaços urbanos. Da utilização de telefones celulares móveis para celulares fixos foi um passo natural, iniciando-se pela instalação de telefones públicos em locais remotos que não dispunham de rede. Como nas áreas cobertas pela telefonia móvel, onde seria economicamente inviável a implantação de uma rede de cabos, havia milhares de potenciais assinantes, desenvolveram-se equipamentos terminais fixos de assinantes com antenas e transceptores semelhantes aos dos terminais móveis com a utilização de um *loop* de assinante via enlace de rádio, dispensando o uso do par de cobre. Essa nova aplicação denominou-se celular fixo, por derivar-se do sistema móvel celular e desprovido de mobilidade. Surgia o embrião de uma nova tecnologia: o *loop* de assinante via rádio, onde a plataforma de transmissão e interligação à central comutadora local é comum a uma grande quantidade de terminais, diferente dos sistemas de

rádio monocanal, onde o transmissor, o receptor, o mastro e a antena eram individuais para cada assinante. Essa nova tecnologia, com o uso de uma plataforma comum a vários assinantes, torna o custo do terminal mais acessível.

I.2 – CANAL DE COMUNICAÇÃO

O diagrama de blocos mostrado na Figura 1.1 ilustra o fluxo do sinal de informação através de um sistema de comunicação digital [Sk01].

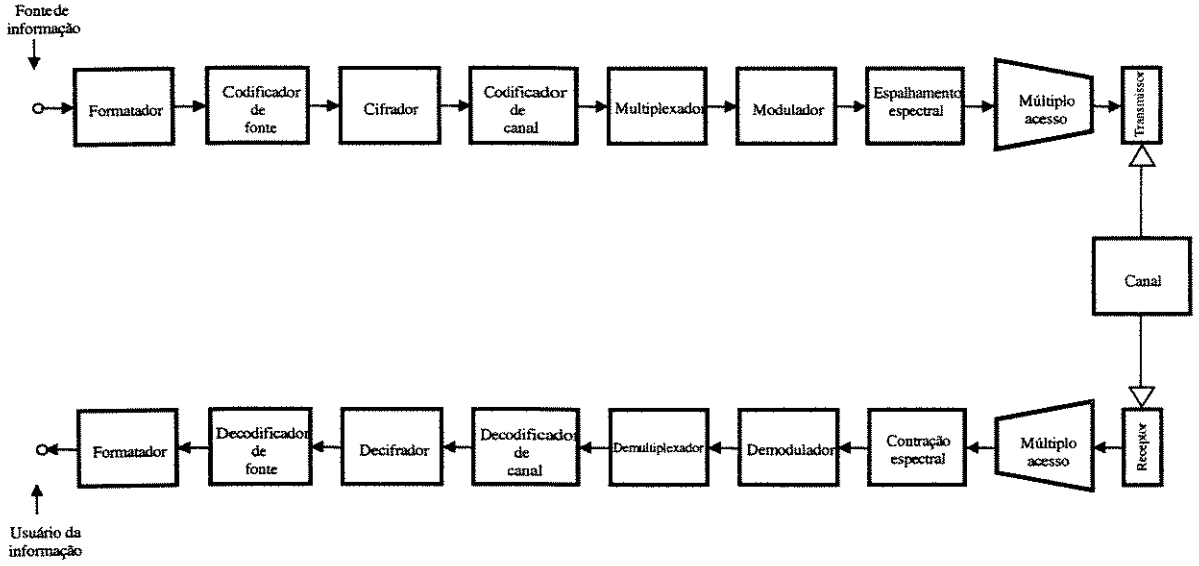


Figura 1.1 – Diagrama de blocos de um sistema de comunicação digital.

Os blocos superiores na Figura 1.1 indicam as transformações do sinal desde a fonte até o transmissor e os blocos inferiores indicam as transformações desse sinal, do receptor para o usuário da informação. Os blocos inferiores revertem os passos executados pelos blocos superiores. O transmissor consiste, em geral, de um estágio conversor de freqüência, de um amplificador de potência e de uma antena. O receptor compõe-se de uma antena, de um amplificador de baixo ruído e de um estágio de conversão, tipicamente para uma freqüência intermediária. De todos os blocos por onde passa o sinal, somente a formatação, a modulação e a demodulação são essenciais para um sistema de comunicação digital.

A formatação transforma a informação da fonte em símbolos digitais. A codificação de fonte remove as redundâncias. A cifragem impede que usuários não autorizados acessem o conteúdo da informação. A codificação de canal pode reduzir a probabilidade de erro ou reduzir os requisitos de relação sinal-ruído. Na modulação os símbolos são convertidos em formas de onda compatíveis com o canal de transmissão. O espalhamento espectral produz um sinal que é menos vulnerável a interferências e pode ser usado para garantir a privacidade da comunicação. Procedimentos de multiplexação e múltiplo acesso combinam sinais que podem apresentar características diferentes ou que podem ter origem em fontes diferentes, o que permite o compartilhamento dos recursos de comunicação. O sistema de comunicação digital precisa de sincronização, para que sejam definidos o início e o fim do procedimento de detecção do sinal.

O fluxo do sinal mostrado na Figura 1.1 representa uma arranjo típico de um sistema de comunicação digital. No entanto, os blocos são implementados, algumas vezes, em uma ordem diferente. Por exemplo, a multiplexação pode aparecer antes da codificação de canal e o espalha-

mento espectral pode aparecer em qualquer lugar ao longo do caminho de transmissão, pois sua localização vai depender da técnica utilizada. A Figura 1.1 ilustra, também, a reciprocidade do procedimento, ou seja, todo passo executado no caminho da transmissão precisa ser revertido no caminho da recepção. Da fonte até o modulador, a informação é caracterizada por uma seqüência de símbolos, em geral, binários. Depois da modulação, a informação tem a forma de uma onda codificada digitalmente. Da mesma forma, na direção inversa, a informação recebida aparece como uma forma de onda digital até que ela seja demodulada. Após este passo, a informação se torna uma cadeia de símbolos e trafega assim por todos os blocos restantes do sistema.

I.2.1 – Modulação

Na modulação alguma característica de um sinal portador é alterada, de acordo com a informação a ser transmitida. As técnicas de modulação têm por objetivo transportar a informação de modo adequado ao meio e com eficiência de espectro. Em uma portadora senoidal podem-se alterar três características: amplitude, freqüência e fase. A característica alterada dá origem à modulação associada: ASK (*Amplitude Shift Keying*), FSK (*Frequency Shift Keying*) e PSK (*Phase Shift Keying*). Existem tipos híbridos de modulação envolvendo variações de amplitude, freqüência e/ou fase: DPSK (*Differential Phase Shift Keying*), QPSK (*Quadrature Phase Shift Keying*), DQPSK (*Differential Quadrature Phase Shift Keying*), MSK (*Minimum Shift Keying*), GMSK (*Gaussian Minimum Shift Keying*), QAM (*Quadrature Amplitude Modulation*). Na modulação QAM são alteradas a amplitude e a fase [Gu97].

A modulação por chaveamento de fase diferencial (DPSK) é uma variação da PSK, em que há a inversão de 180^0 na fase da portadora sempre que ocorre o bit 0. Esta técnica é também chamada de BPSK (*Binary* PSK). As alterações consecutivas em uma seqüência de bits 0 auxiliam no sincronismo da comunicação. A técnica DPSK possui variações em que se considera a unidade de informação como o conjunto de 2 (*dibit*) ou até 3 bits (*tribit*). Desta forma, para cada variação da portadora, transmitem-se 2 ou 3 bits. No caso *dibit*, chama-se técnica DPSK-4, uma vez que dois bits definem 4 possíveis estados. Cada estado é representado por uma alteração, no ângulo da portadora, múltipla de 90°. No caso de DPSK-8, a unidade de informação é o *tribit*, pois para cada conjunto de 3 bits faz-se uma variação de fase múltipla de 45°.

A modulação por chaveamento de fase e quadratura (QPSK) é uma variação do PSK, em que dois sinais BPSK são transmitidos defasados de 180°. Isto duplica a quantidade de informação transmitida. A variação de fase resultante da combinação das componentes em fase e quadratura a ser aplicada na portadora dependerá de cada *dibit* a ser transmitido. Como as variações de fase são múltiplas de $\pi/4$ define-se esta técnica como $\pi/4$-QPSK.

A modulação por chaveamento de fase em quadratura diferencial (DQPSK) aproveita a característica de sincronismo da transmissão do DPSK e a aplica ao QPSK. Esta técnica resolve o problema de portadora com amplitude zero em determinadas transições. No DQPSK as transições dos 2 bits do *dibit* ocorrem em instantes de tempo defasados de meio bit. Isto faz com que apenas um dos bits do *dibit* sofra alteração implicando que a variação de fase máxima deste bit seja de $\pi/2$, eliminando assim as transições de π radianos. Isto elimina a possibilidade da portadora passar pela amplitude zero.

A modulação por chaveamento mínimo (MSK) é uma modulação em freqüência, FSK, com uma separação mínima entre as portadoras utilizadas de modo a garantir a ortogonalidade entre elas. A modulação por chaveamento mínimo gaussiano (GMSK) é uma modificação da técnica

MSK, na qual a seqüência de pulsos em banda base na entrada do modulador é filtrada por um filtro passa-baixa com resposta a um pulso retangular gaussiano. A saída deste filtro é responsável por modular em MSK as portadoras utilizadas. O efeito do filtro é o de conformar os pulsos de entrada do modulador MSK tornando as transições de freqüência mais suaves e, com isso, reduzindo a largura de faixa do lóbulo principal do sinal modulado.

A modulação por amplitude em quadratura (QAM) é uma combinação dos esquemas ASK e PSK modificando simultaneamente a amplitude e a fase da portadora, por isto também é conhecida como AMPSK.

As técnicas QPSK, MSK e BFSK são as freqüentemente usadas em sistemas WLL (*Wireless Local Loop*), apesar de suas eficiências espectrais serem moderadas [FAG95].

I.2.2 – Múltiplo acesso

As técnicas de múltiplo acesso são utilizadas para permitir o compartilhamento de uma determinada faixa de radiofreqüência entre os vários terminais solicitantes. Ao conjunto das regras que são estabelecidas, para que um determinado recurso de comunicação seja compartilhado por diversos usuários, denomina-se protocolo de múltiplo acesso.

As técnicas básicas de múltiplo acesso são: múltiplo acesso por divisão de freqüência FDMA (*Frequency Division Multiple Access*), múltiplo acesso por divisão temporal TDMA e múltiplo acesso por divisão de código CDMA (*Code Division Multiple Access*).

Na técnica FDMA a faixa de transmissão é dividida em um determinado número de canais, que são atribuídos aos usuários através de um processo de consignação por demanda. A largura de banda do canal e o tipo de modulação determinam qual a máxima taxa de bits suportada. As faixas de guarda são introduzidas entre os canais de freqüência para evitar a interferência de canal adjacente. Essa técnica somente se aplica aos sistemas analógicos.

Na técnica TDMA cada usuário dispõe da faixa de freqüência durante um determinado período de tempo denominado *slot* (janela).

Na técnica CDMA uma seqüência de código distinta é atribuída a cada usuário e todos os usuários utilizam a mesma faixa de freqüência durante todo o tempo.

O desempenho do protocolo depende de vários fatores: do método de reserva do canal de *uplink*, do algoritmo de resolução de colisão executado pelos terminais que tiveram seus pacotes de solicitação colididos no canal de acesso aleatório, do algoritmo de alocação de janelas utilizado pela estação rádio-base para alocar janelas aos terminais e do tamanho do quadro.

Para suportar a comunicação bidirecional entre os terminais e a estação rádio-base, os recursos de transmissão provêem comunicação nas direções de *uplink* (do terminal para a estação rádio-base) e de *downlink* (da estação rádio-base para o terminal). Esta comunicação pode ocorrer através de um esquema FDD, no qual as comunicações de *uplink* e *downlink* são fisicamente separadas em canais de freqüências diferentes, ou através de um esquema TDD, onde as transmissões de *uplink* e *downlink* ocorrem na mesma freqüência, mas em tempos alternados. Para a transmissão de *uplink*, os terminais compartilham o canal de comunicação usando um protocolo de múltiplo acesso e o canal de *downlink* opera com o modo *broadcasting*,

livre de contenção TDM (*Time Division Multiplexing*) para *polling*, transporte de dados e garantir a realimentação às solicitações dos terminais.

Os protocolos baseados na técnica TDMA, objeto de estudo desse trabalho, englobam as transmissões de *uplink* e *downlink* e são de controle centralizado, no qual as estações rádio-base controlam as transmissões e retransmissões dos terminais. O canal de *uplink* é usado pelos terminais para enviar solicitações e pacotes de informações, em modo de contenção e reserva. O canal de *downlink é* usado pela estação rádio-base para enviar reconhecimento e pacotes de informação em modo programado. Os quadros dos canais de *downlink* e *uplink* são divididos em janelas de tempo e a alocação desses canais é feita por demanda. No quadro de *uplink* existem janelas de contenção, de reserva, de *paging*, e de *reporting*. No quadro de *downlink* existem janelas de informação, de reconhecimento de solicitações, de *polling* e de anúncio. Em geral, no *uplink*, o cabeçalho das janelas de reserva contém bits de sincronização, um campo que indica o número de janelas solicitadas pelo terminal e outros campos de controle; o cabeçalho das janelas de contenção contém bits de sincronização, o tipo de serviço: CBR, VBR ou ABR, a banda efetiva, o identificador do terminal, o *deadline* do dado e campos de controle. O rodapé, em ambas as janelas, contém um campo para controle de erro. O tamanho do quadro pode ser fixo ou variável.

A estação rádio-base envia regularmente mensagens para verificar se os terminais, em sua área de cobertura, desejam iniciar uma nova sessão. Esse processo é chamado de *polling*. Se algum terminal deseja transmitir, ele espera, no mínimo, até o início do próximo quadro para sincronizar o seu *clock* com o da estação rádio-base. Além da sincronização, o terminal executa o controle de potência para compensar as atenuações do sinal causadas pelas diferentes distâncias entre os terminais e a estação rádio-base; e pelas diferentes condições de propagação. As janelas de contenção do quadro de *uplink* são usadas pelos terminais na solicitação de banda e as janelas de reserva são usadas na transmissão de dados. Alguns protocolos utilizam mini-janelas para solicitação.
Em cada quadro de *downlink*, a estação rádio-base indica as janelas de contenção concedidas para cada terminal solicitar reserva. Tipicamente, a estação rádio-base aloca mais de uma janela de contenção para cada terminal e uma janela pode ser alocada a múltiplos terminais. Os terminais que solicitaram janelas de reserva e foram atendidos, recebem, no quadro de *downlink*, quais são as suas janelas de reserva do quadro de *uplink*.

O número de janelas de contenção, alocadas em cada quadro de *uplink*, e o número de terminais que podem fazer uso dessas janelas de contenção são variáveis e controlados pela estação rádio-base.

Os possíveis estados dos terminais são mostrados na Figura 1.2:
Inativo, sincronizado e com controle de potência, apto a receber *polling*;
Passivo, quando o terminal já solicitou janela de contenção e ouve as janelas de contenção que lhe foram alocadas;
Em contenção, quando o terminal participa do processo de contenção e pode sofrer colisão;
Em transmissão ou conectado, quando o terminal transmite informação nas janelas de reserva.

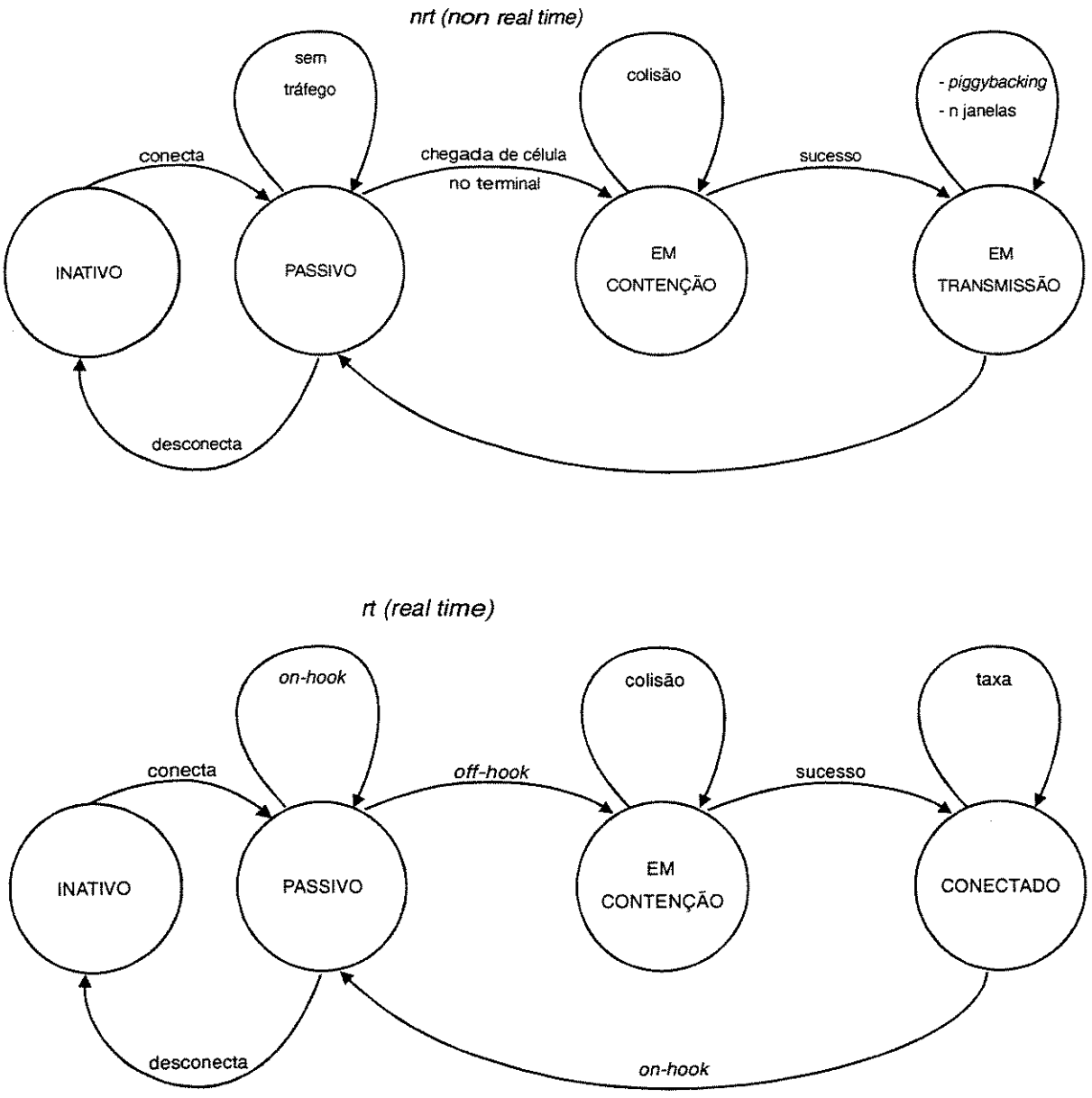


Figura 1.2 – Diagrama de estados dos terminais

O terminal, no estado em contenção, escuta as janelas, escolhe uma janela de contenção, de um sub-grupo de janelas sem colisão, de um grupo de janelas disponibilizado para sua classe, e envia uma mensagem de solicitação na janela escolhida. Como o canal de contenção é um canal de acesso aleatório, podem ocorrer colisões nessas janelas. No quadro de *downlink* a estação rádio-base transmite o resultado da solicitação enviada, que pode ser de sucesso, de colisão, ou indicando que não houve o uso efetivo da janela de contenção por nenhum dos terminais para os quais esta janela foi alocada.

A utilização simultânea da janela de contenção, por mais de um terminal, causa colisão que é resolvida utilizando-se um algoritmo de resolução de colisão, como, por exemplo, a geração de um atraso aleatório para determinar quantos quadros o terminal deve esperar antes de re-

transmitir o seu pedido. Todas as solicitações envolvidas na colisão são perdidas. No caso da chegada de novas células, e se a solicitação precisa ser novamente submetida por causa da colisão, o terminal atualiza o tamanho do pedido para incluir as novas células recém-chegadas. Para tráfegos CBR e VBR que possuem rigorosos limites de tempo, não tem sentido a repetição ilimitada do procedimento de contenção. Para esses tráfegos, existe um tempo máximo para contenção. Se o terminal gasta um tempo maior que este tempo máximo de contenção, sua solicitação é descartada e o terminal retorna ao modo passivo.

Quando o terminal consegue transmitir a sua solicitação com sucesso, ele recebe da estação rádio-base as suas janelas de reserva, de acordo com a carga de tráfego corrente, com o *deadline* do dado e com a política de agendamento adotada. Os terminais transmitem suas células nas janelas de reserva; é também possível o envio de dados na janela de contenção. A alocação de janelas de contenção e de reserva, que pode ser fixa ou adaptativa, é controlada pela estação rádio-base através de um algoritmo de alocação de janelas.

I.2.2.1 – Protocolos de Múltiplo Acesso

A seguir são apresentados alguns dos protocolos de múltiplo acesso propostos na literatura.

I.2.2.1.1 – DAVIC 1.2 MAC

DAVIC (*Digital Audio Visual Council*) é uma organização sem fins lucrativos com o objetivo de definir especificações para interfaces, protocolos e arquiteturas de aplicações e serviços digitais áudio-visuais.
O controle de acesso ao meio MAC da recomendação DAVIC 1.2 possui um protocolo específico para as redes de acesso fixo sem fio BWLL (*Broadband Wireless Local Loop*) [DAVIC 1.4]. O tamanho das janelas dos quadros de *downlink* e *uplink* é fixo e de 68 bytes, sendo 53 bytes da célula ATM, 1 byte de tempo de guarda, 4 bytes de sincronização e 10 bytes do código corretor de erro *Reed-Solomon*.

No canal de *uplink* cada quadro é subdividido em janelas de resposta *poll*, janelas de contenção e janelas de reserva, todas elas de tamanho fixo. As janelas de resposta *poll* contêm somente mensagens MAC. As janelas de contenção podem transportar mensagens MAC, para requisição de janelas de reserva ou transportar dados das camadas superiores. E as janelas de reserva transportam células das camadas AAL (*ATM Adaptation Layer*), para voz e dados, e células de operação e gerência OAM (*Operation And Maintenance*).

As janelas de *poll* são alocadas para um ou vários terminais e são utilizadas somente para resposta de *poll*, após o recebimento de um pedido de *poll* feito pela estação rádio-base.
As janelas de contenção são alocadas para mais de um terminal na solicitação inicial de banda e as janelas de reserva são utilizadas para transportar dados à BS.
A estação rádio-base, através do mecanismo de *polling*, inquire periodicamente (tipicamente a cada 2 segundos) cada terminal para descobrir se ele está ativo e se ele pretende usar janelas de contenção. Quando um terminal tenta estabelecer uma chamada, ele adquire um canal de *downlink* e espera um *poll* direcionado para ele. Após executar a sincronização, o controle de potência e ter pelo menos uma célula para transmitir, o terminal está liberado para participar do processo de contenção para envio de suas células.
Em caso de colisão, o DAVIC não especifica um algoritmo de resolução de colisão em particular.

I.2.2.1.2 – IEEE 802.14 WG MAC

As propostas do IEEE 802.14 Working Group foram preparadas por um comitê de trabalho composto por membros da Lucent, AT&T, NIST e outros. O objetivo foi definir um padrão para transmissão de dados sobre redes HFC (*Hybrid Fiber Coax*) [RBK97]. O IEEE 802.14, além dos modos ATM e MPEG-2, suporta o modo de transporte com pacotes de tamanho variável. O protocolo não especifica uma estrutura de quadro em particular. Os terminais utilizam o procedimento de contenção para solicitação inicial de banda e o procedimento de *piggybacking* (ver II.2.2) para solicitações adicionais de banda.

O canal de *downlink* possui um conjunto de várias unidades de 6 bytes, que são concatenadas para transportar células ATM ou pacotes de tamanho variável. As células ATM são usadas para transmitir informações do usuário e mensagens de controle MAC.

O canal de *uplink* é segmentado em unidades básicas chamadas de mini-janelas, de tamanho fixo, que são usadas para contenção e para reserva. As mini-janelas de contenção possuem 16 bytes (6 bytes de dados, 2 bytes de CRC (*Cyclic Redundancy Check*) e 8 bytes para o cabeçalho da camada física). As mini-janelas de reserva possuem 6 bytes de cabeçalho MAC, 48 bytes de informação, 1 byte de tempo de guarda, 3 bytes de sincronização e 6 bytes de FEC (*Foward Error Correction*). As janelas de dados são formadas concatenando um número fixo de mini-janelas de reserva para, no modo ATM, transmitir uma célula ATM e concatenando um número variável de mini-janelas para um modo de transporte com pacotes de tamanho variável no tráfego IP (*Internet Protocol*).

O IEEE 802.14 usa o mecanismo de realimentação de três mensagens: colisão, sucesso e livre, às solicitações de banda, através do quadro de *downlink* e especifica o algoritmo *ternary tree* para resolução das colisões. Os terminais que estão ativos e desejam solicitar banda adicional, substituem o procedimento de contenção pelo processo de *piggybacking* através do quarto bit no campo EBR (*Extended Bandwidth Request*) do cabeçalho da janela de dados.

I.2.2.1.3 – MCSN DOCSIS MAC

O documento DOCSIS (*Data Over Cable Service Interface Specifications*) descreve as interfaces de rede, que permitem transmissão de tráfego IP bidirecional entre uma estação rádio-base e os terminais de uma rede HFC com utilização de ATM e outros formatos PDU (*Protocol Data Unit*). Esse projeto teve início através de uma organização chamada MCNS (*Multimedia Cable Network System*) cujos participantes eram Comcast Cable Communications, Time Warner Cable e Cox Communications and Telecommunications Inc. Posteriormente, a Rogers Cable Systems Limited e a Media One and CableLabs se associaram à MCNS em contribuição ao documento DOCSIS [DOCSIS99].

Os canais de *uplink* e *downlink* consistem de mini-janelas de tamanho variável. O tamanho varia dependendo do esquema de modulação e da taxa do canal utilizados. A duração de uma mini-janela é dada pela relação 6.25×2^M μsegs, onde M = 1,2,3,..7. Por exemplo, para uma capacidade de *uplink* de 5Mbps usando QPSK (2 bits/símbolo), o tamanho da mini-janela é de 8 bytes e na modulação 16-QAM (4 bits/símbolo), tem-se uma mini-janela de 16 bytes. Não existe uma estrutura de quadro especificada pelo protocolo. Várias mini-janelas são concatenadas para formar células ATM e PDUs de tamanho variável. No transporte ATM, o tamanho da janela de dados é de 72 bytes (6 bytes do cabeçalho MAC, 53 bytes da célula ATM, 10 bytes do FEC

e 3 bytes de quantização). Os terminais utilizam o procedimento de *piggybacking/polling*, usando 8 bytes de cabeçalho extendido, para solicitações adicionais de banda. O DOCSIS especifica o algoritmo *binary exponential backoff* para resolução de colisões.

I.2.2.1.4 – MASCARA

O protocolo MASCARA (*Mobile Access Scheme based on Contention and Reservation for ATM*) proposto em [BDMMP96] para o projeto WAND (*W-ATM Network Demonstrator*) foi desenvolvido com o suporte do EC (*European Community*). O protocolo utiliza o esquema TDD para suas transmissões e opera em modo hierárquico através de um programa *Master* na estação rádio-base e um programa *Slave* em cada terminal. O tráfego de *downlink* é transmitido em modo TDM, enquanto os pacotes do *uplink* são transmitidos em modo de contenção/reserva. O MASCARA possui o quadro de tamanho variável com dois subquadros: um para os canais de *uplink* e o outro para os canais de *downlink*. No subquadro de *downlink*, tem-se o cabeçalho do quadro e o tráfego de reserva. No subquadro de *uplink* tem-se: o tráfego de reserva e o tráfego de contenção. Os subquadros são subdivididos em janelas de tamanho variável e esse tamanho depende do tráfego na rede. O quadro no protocolo MASCARA sempre começa com um cabeçalho usado pela estação rádio-base para informar o tamanho do quadro, os resultados dos procedimentos de contenção do quadro anterior e a alocação de janelas para cada terminal ativo. Os terminais usam as janelas de contenção para transmitir as suas solicitações de reserva ou informações de controle. Os períodos em modo reserva podem não ter nenhuma janela, enquanto o período de contenção é sempre mantido com algumas janelas para atender os terminais que desejam transmitir.

Para gerenciar o processo de transmissão, o programa *Master* define um trem de células que é uma seqüência de pacotes ATM de um terminal com um cabeçalho comum. O cabeçalho possui o tamanho de uma célula ATM e inclui bits de sincronização e bits específicos do cabeçalho.

O programa *Master* analisa várias informações, como a classe de serviço, a QoS (*Quality Of Service*) negociada, a quantidade de tráfego e o número de solicitações de reserva para determinar a natureza e o volume de tráfego que será transmitido no próximo quadro. Essas informações são mantidas em um mapa que especifica o tamanho dos três períodos (*uplink*, *downlink* e contenção) no próximo quadro e a alocação das janelas do quadro para cada terminal envolvido. A estação rádio-base comunica o mapa dentro do cabeçalho no início de cada quadro. Com a informação desse mapa, cada terminal pode determinar se ele tem permissão para receber ou transmitir informações no quadro corrente. O programa *Master* usa um algoritmo chamado PRADOS (*Priority Regulated Allocation Delay Oriented Scheduling*) que faz a alocação adaptativa das janelas do quadro. O programa *Slave*, executado em cada terminal, é responsável por determinar, na janela alocada para ele, qual o tráfego a ser enviado com prioridade no período de *uplink*.

Uma desvantagem do protocolo MASCARA é o tamanho da janela de contenção (2 pacotes ATM). Em caso de alto tráfego, onde a probabilidade de colisão cresce, pode resultar em uma redução de vazão. Com um tamanho de quadro variável, dificulta-se a alocação de tráfego CBR. Por exemplo, no caso de uma chamada de voz (64 kbps), se o tamanho do quadro for menor que o tempo para preencher um pacote ATM (~6 ms), pode haver quadro sem janela, e se o tamanho do quadro for maior que 6 ms, será necessário alocar mais de uma janela no quadro para essa chamada [SMM97].

I.2.2.1.4.1 – PRADOS

O PRADOS é o algoritmo de alocação adaptativa das janelas do quadro, proposto para o *Master Scheduler* do MASCARA, cujo objetivo é regular o tráfego, baseado nas características do contrato, e manter os limites de tráfego de cada conexão da interface rádio [MWAND98], [PMSBMD98].

O PRADOS determina o mapa de alocação das janelas de acordo com as solicitações, as características de tráfego e os acordos de qualidade de serviço. No início de cada quadro, o PRADOS tem o número de solicitações pendentes, seja de células do *downlink*, em espera para serem transmitidas pela BS ou, de solicitações do *uplink*, que são *piggybacked* em dados MPDUs (*MAC Protocol Data Units*). O algoritmo é separado em duas ações independentes que podem ser executadas em paralelo: a especificação de quantas solicitações, para cada conexão ativa, serão atendidas no quadro corrente e a determinação da localização no quadro das janelas alocadas para cada serviço requerido. Para a primeira ação, o algoritmo combina prioridades com um regulador de tráfego do tipo balde furado utilizando um depósito de senhas. Ele classifica as conexões, baseado em suas classes de serviço e atribui prioridades às conexões (o maior número de prioridade representa a conexão de mais alta prioridade), conforme mostrado na Tabela 1.2.

Número de prioridade	Classe de serviço
5	CBR (Constant Bit Rate)
4	VBR rt (Variable Bit Rate real time)
3	VBR nrt (Variable Bit Rate non real time)
2	ABR (Available Bit Rate)
1	UBR (Unspecified Bit Rate)

Tabela 1.2 – Prioridades das classes de serviço no PRADOS.

Um tipo de senha é introduzido a cada nova conexão. As senhas são geradas a uma taxa fixa igual à taxa média de células, SCR (*Sustainable Cell Rate*) e o tamanho do depósito é igual à duração dos surtos da conexão, MBS (*Maximum Burst Size*), acordados no estabelecimento da chamada. A cada janela alocada para uma conexão, uma senha é removida do seu depósito correspondente. Desta forma, a qualquer instante, o estado de cada depósito dá uma indicação da largura de banda consumida pela conexão. O depósito de senhas é implementado como uma variável que é incrementada unitariamente a todo instante que uma senha é gerada, e decrementada unitariamente, a todo instante que uma janela é alocada para a conexão correspondente. É permitido que esta variável fique com valor negativo se janelas são alocadas quando o seu depósito de senhas está vazio.

A primeira ação, (alocação), é dividida em dois passos. No primeiro passo, o algoritmo atende às solicitações que estão em conformidade com o contratado, ou seja, solicitações que pertencem às conexões cujos depósitos de senhas não estão vazios. O regulador de tráfego do tipo balde furado começa da classe de prioridade 5 (CBR), e chega até a classe de prioridade 2 (ABR). O algoritmo atende às solicitações de *uplink* e *downlink* pertencentes a cada classe de serviço enquanto houver senhas e janelas disponíveis. As conexões da classe UBR não têm garantia de atendimento e não são mantidos depósitos de senhas para esse tipo de conexão. Assim, as conexões da classe UBR não são servidas durante esse passo. Para as conexões da classe CBR, os terminais não transmitem solicitações e o algoritmo antecipa o tráfego, alocando janelas com base na geração de solicitações, à taxa PCR (*Peak Cell Rate*), imaginárias no *uplink*.

Em toda classe de prioridade, é provável que exista mais de uma conexão com células ATM para serem transmitidas. Nesse caso, o PRADOS aloca gradualmente uma janela de cada vez para a conexão (ou conexões) que possui o depósito com o maior número de senhas, removendo uma senha do depósito correspondente (decrementando a variável de um). Desta forma, a conexão que possui o depósito com o maior número de senhas consumiu proporcionalmente menos largura de banda do que a contratada, e assim, tem maior prioridade para alocar janelas. Nesse estado, uma ou duas das situações seguintes acontecem:

i) todos os depósitos de senhas estão vazios (as variáveis estão menores ou igual a zero);

ii) todas as solicitações foram atendidas.

Se somente a situação i) acontece e ainda existem janelas disponíveis, o algoritmo executa o segundo passo, que é tentar atender às solicitações que não estão em conformidade com o contratado, ou seja, as conexões cujos depósitos não possuem senhas (variáveis negativas). Nesse estado, como as variáveis de todas as conexões estão menores ou iguais a zero, o algoritmo segue o mesmo procedimento anterior, começando com o atendimento da prioridade 5 (CBR) até a prioridade 1 (UBR). Se mais de uma conexão, pertencente à mesma classe de prioridade solicita janelas, uma janela de cada vez é alocada para a conexão cujo depósito possui o maior número de senhas (variável com maior valor). Se existe um excesso de tráfego, o decremento irá resultar em valores negativos para as variáveis, como no passo anterior. O procedimento termina quando todas as solicitações são satisfeitas ou todas as janelas disponíveis já estão alocadas.
Prevendo um possível congestionamento à frente, na rede, as células de *uplink* excedentes têm seu bit de CLP (*Cell Loss Priority*) estabelecido em zero (0).

O *deadline* de uma célula ATM não é um limiar absoluto, mas uma indicação da rapidez com que uma célula ATM deve ser transmitida. A estratégia do algoritmo é alocar as células segundo os seus *deadlines*. Quando isso não é possível, alguns atrasos extras são permitidos (*deadline* extra).

Para a segunda ação, (determinação da localização, no quadro, das janelas alocadas para cada serviço requerido), o PRADOS baseia-se na idéia de que, para maximizar a fração de células ATM que são transmitidas antes de seus *deadlines*, cada célula ATM é transmitida o mais próximo possível de seu *deadline*.

A construção do trem de células e a separação dos períodos de *uplink* e *downlink* dentro de cada quadro são itens importantes na operação do PRADOS. Os trens de células são construídos gradualmente, de acordo como o algoritmo do balde furado e ordenados de acordo com os seus *deadlines*. O *deadline* do trem de células é considerado igual ao *deadline* de sua mais antiga célula ATM. Uma célula ATM é agrupada ao final de seu trem se esta ação não causar violação do *deadline* do trem.

I.2.2.1.5 – DQRUMA

O protocolo DQRUMA (*Distributed Queueing Request Update Multiple Access*) proposto em [KLE95] é um dos protocolos MAC existentes para redes ATM sem fio. O DQRUMA utiliza o esquema FDD para suas transmissões e considera um sistema de janelas sem referência de quadro. Os canais de acesso à solicitação e de transmissão de pacotes são formados janela a janela. O canal de *uplink* é dividido em duas seções: a primeira para contenção, com utilização de mini-janelas, e a segunda para transmissão de dados de informação com utilização de uma

janela. A estação rádio-base pode aumentar o número de mini-janelas de contenção para aliviar a contenção. O canal de *downlink* consiste de uma série de mini-janelas, para notificação do recebimento das solicitações, e de uma janela para transmissão de pacotes.

Os terminais podem estar em um dos estados: vazio, em solicitação, ou esperando para transmitir. Quando o terminal envia o seu identificador à estação rádio-base, usando o canal de acesso do *uplink,* ele passa ao estado em solicitação. Após receber a solicitação do terminal, a estação rádio-base atualiza a tabela de solicitações que tem uma entrada para cada um dos terminais, e envia um reconhecimento ao terminal. Cada entrada na tabela contém o identificador do terminal e a informação se ele tem mais pacotes para transmitir. Quando o terminal recebe o reconhecimento à sua solicitação, ele muda para o estado em espera para transmitir. A estação rádio-base determina qual terminal tem permissão para transmitir na próxima janela. O terminal transmite um pacote na próxima janela e muda para o estado vazio (se ele não tem mais pacotes na fila), ou para o estado em espera para transmitir (se ele tem mais pacotes). A cada instante, um terminal transmite um pacote ATM, e se ele tem mais pacotes para transmitir, ele inclui uma mensagem de *piggybacking*. Essa característica permite aos usuários VBR atualizarem sua demanda de tráfego sem a necessidade de contenção extra.

O DQRUMA utiliza uma generalização do algoritmo *binary tree* para a resolução de contenção. Apesar da estrutura dinâmica de quadro e do mecanismo de *piggybacking* serem úteis para uma melhor utilização do canal, o protocolo DQRUMA não possui uma estratégia de alocação adaptativa de janelas. Assim, a QoS para os diferentes usuários, CBR, VBR e ABR não é otimizada [KL01].

I.2.2.1.6 – SCAMA

O SCAMA (*Synergistic Channel Adaptive Multiple Access*) proposto em [LK00] para redes ATM sem fio, é um protocolo que utiliza o esquema FDD em suas transmissões. O protocolo trabalha em conjunto com a camada física, através da verificação da informação do estado do canal CSI (*Channel State Information*) de cada terminal nas solicitações do quadro de *uplink* e aloca as janelas de informação aos terminais de acordo com uma função de prioridade que provê um balanço entre o tipo de tráfego, o CSI, a urgência e a vazão. A operação do protocolo SCAMA é dividida em duas fases: de solicitação e de transmissão.

Os quadros de *uplink* e *downlink* são divididos em subquadros. No *uplink*, o quadro é dividido em três subquadros: de solicitação, de tráfego e de *report*. Mesmo que o terminal ABR tenha sido admitido para transmitir suas células no subquadro de tráfego corrente, ele tem que participar de uma nova contenção no próximo quadro, se ele ainda possui células para transmitir. Quando os terminais CBR e VBR fazem uma solicitação bem sucedida de transmissão em uma das mini-janelas, eles não necessitam participar de uma nova contenção. O quadro de *downlink* é dividido em quatro subquadros: de reconhecimento (para informação de qual terminal teve sucesso), de *polling*, de tráfego e de anúncio (para informação do esquema de alocação das janelas de tráfego do *uplink*). No *uplink* e no *downlink* a codificação e modulação adaptativas com vazão variável são aplicadas somente às janelas de tráfego. Para as mini-janelas dos demais subquadros, a modulação QPSK é usada por garantir a menor probabilidade de erro. O terminal envia a sua solicitação na mini-janela com uma certa probabilidade de permissão. As probabilidades de permissão para os terminais CBR, VBR e ABR são distintas.

A estação rádio-base coleta os pedidos de solicitação e os pedidos pendentes dos quadros anteriores, antes de fazer as alocações das janelas de tráfego do *uplink*. Todas as solicitações

são atendidas prioritariamente de acordo com o *deadline*, o CSI, o tipo de serviço e o tempo de espera da solicitação (número decorrido de quadros desde a aceitação do pedido). Como a camada física oferece uma vazão variável, que é dependente do CSI, o SCAMA dá maior prioridade aos terminais de melhor condição de canal, no processo de alocação de banda, porque os terminais podem desfrutar de uma vazão maior e usar o sistema de largura de banda de uma forma mais eficiente.

I.2.2.1.7 – DSA++

O DSA++ (*Dynamic Slot Assignment*) proposto em [Pe95] é um protocolo para redes ATM sem fio. O DSA++ usa o esquema FDD e tem quadro de tamanho variável. O *uplink* é decomposto em sinalização (mini-janelas) e transmissão (reserva) assim como o *downlink*. Um deslocamento entre o ponto de início de cada período (*uplink* e *downlink*) é usado para compensar o atraso de propagação (*round-trip*). Para cada período de sinalização de *downlink* corresponde um período de sinalização do *uplink* de mesmo tamanho.

O algoritmo de contenção utilizado é o *binary tree*. O número de mini-janelas de contenção é atualizado dinamicamente baseado em estimativa feita pela estação rádio-base.

O algoritmo de alocação das janelas de reserva considera o número de células agendadas e os *deadlines* determinando prioridades na ordem: CBR > VBR > ABR > UBR.

I.2.2.1.8 – DPRMA

O DPRMA (*Dynamic Packet Reservation Multiple Access*) proposto em [DH99] é um protocolo de controle de acesso ao meio para aplicações multimídia sem fio. Utiliza o esquema FDD com quadro de tamanho fixo contendo janelas de contenção e de reserva (de mesmo tamanho) no *uplink*.

O algoritmo de contenção é do tipo *R-Aloha* (*Reservation Aloha*) e o número de janelas de contenção é fixo no quadro.

O tamanho do quadro e o tamanho das janelas são determinados para que terminais de voz transmitam uma janela por quadro. As demais janelas de reserva são alocadas aos outros tráfegos em tempo real prioritariamente em relação ao tráfego não em tempo real. Os terminais CBR e VBR recebem um número de janelas no quadro proporcional às taxas solicitadas. As parcelas não atendidas das taxas são tratadas assim que os terminais passem para o estado passivo liberando janelas no quadro. O retorno do estado passivo para em transmissão passa necessariamente pelo processo de contenção. A variação de taxas solicitadas pelos terminais e atendidas pela estação rádio-base é incremental e é resultado da avaliação do tamanho das filas nos buffers.

I.2.2.1.9 – CRPMA/DS

O CRPMA/DS (*Combination of Reservation and Polling Multiple Access with Distributed Scheduling*) proposto em [CCL01] é um protocolo MAC para redes ATM sem fio. Ele utiliza o esquema TDMA-FDD nas transmissões de *uplink* e *downlink* e classifica os tipos de serviços em duas categorias: grupo de tráfego de alta prioridade (CBR e VBR) e grupo de tráfego de baixa prioridade (ABR e UBR). Os quadros de *uplink* e de *downlink* possuem o mesmo tamanho fixo.

No quadro de *uplink* existem cinco tipos de janelas: mini-janelas de reserva (para reserva do tráfego de alta prioridade), janelas de tráfego de alta prioridade (para transmissão do tráfego CBR e VBR), janelas de tráfego de baixa prioridade (para transmissão do tráfego ABR e UBR), mini-janelas de controle (o terminal informa à estação rádio-base o número de janelas que ele precisa para transmitir o tráfego VBR em surtos), janelas de compensação (para transmitir tráfego VBR em surtos). Nas janelas de tráfego de alta prioridade e de tráfego de baixa prioridade, existe um campo para informação de *piggybacking*. A estação rádio-base define o número de cada tipo de janela no quadro de acordo com a carga de tráfego. Ela informa aos terminais o esquema de alocação no início de cada quadro.

No quadro de *downlink* existem cinco tipos de janelas: janelas de sincronização, janelas de informação 1 (para informar aos terminais os resultados dos pedidos de reserva, o esquema de alocação das janelas e a posição das janelas em que a estação rádio-base irá transmitir pacotes aos terminais), janelas gerais (para transmissão de pacotes aos terminais), janelas de mensagem (para informação das posições das mini-janelas de reserva), janelas de *polling* (para inquirir os terminais sobre a necessidade de janelas para escoamento do tráfego VBR em surtos) e janelas de informação 2 (para informar o esquema de alocação para o tráfego VBR em surtos e para o tráfego de baixa prioridade).

O protocolo utiliza dois algoritmos para coordenar a transmissão dos pacotes: *master scheduler* (na estação rádio-base) e *slave scheduler* (em cada terminal). O *master scheduler* tem a responsabilidade de alocar janelas aos terminais e o *slave scheduler* determina qual tipo de tráfego será transmitido.

I.2.2.1.10 – PRMA/DA

O PRMA/DA (*Packet Reservation Multiple Access with Dynamic Allocation*) proposto em [KW96] é um protocolo MAC para redes locais sem fio. O protocolo é uma versão estendida do protocolo PRMA (*Packet Reservation Multiple Access)*, um método de multiplexação projetado para entrega de sinais de voz em um sistema TDMA. O PRMA foi desenvolvido para atender tráfego de voz e dados [GVGR89] e a versão estendida permite que usuários VBR façam reservas baseadas em suas taxas de bits. O protocolo PRMA/DA opera com quadros de tamanho fixo. O quadro de *uplink* é segmentado em janelas de contenção e janelas de reserva e o quadro de *downlink* opera em um formato TDM livre de contenção sob controle da estação rádio-base. O acesso e a utilização da rede são feitos em modo contenção/reserva.

O protocolo PRMA/DA considera três estados para representar o terminal: inativo, em contenção e em reserva. Um terminal está inicialmente no estado inativo. Quando um pacote é gerado, o terminal passa ao estado em contenção e tenta transmitir um pacote de acordo com o procedimento *slotted aloha*. Se o acesso à rede tem sucesso, o terminal passa ao estado em reserva.

A característica adicional do protocolo PRMA/DA com relação ao PRMA é o mecanismo de estruturação dinâmica dos quadros, no qual a estação rádio-base ajusta a proporção relativa de janelas de contenção e janelas de reserva de acordo com o estado de congestionamento da rede. O objetivo do algoritmo de alocação adaptativa no PRMA/DA é ter um número maior de janelas de reserva enquanto mantém um número mínimo de janelas de contenção para atender os terminais em contenção. O número de janelas de contenção é mantido em um quando não existe demanda. Se a estação rádio-base detecta a colisão em uma janela de contenção, ela disponibiliza, no próximo quadro, mais uma janela de contenção. Mas, se a estação rádio-base verifica que alguma janela de contenção não foi usada, ela converte essa janela em janela de reserva no próximo quadro. O número de janelas de reserva atribuído aos terminais, que estão em reserva, é determinado pelo algoritmo de alocação adaptativa. O principal fator considerado no algoritmo são as propriedades estatísticas de tráfego que o terminal pretende transmitir. A estação rádio-base ajusta o número de janelas de reserva alocadas a cada terminal em reserva de acordo com o número de janelas de reserva solicitadas pelo terminal e a banda efetiva. Ao final de cada quadro, a estação rádio-base comunica o número de janelas de reserva atribuídas aos terminais em reserva e suas respectivas localizações.

O mecanismo de estruturação dinâmica dos quadros aumenta a estabilidade e a utilização do sistema porque a contenção é resolvida rapidamente. O protocolo não utiliza mini-janelas para solicitação, portanto a troca de uma janela de reserva por uma janela de contenção significa em uma perda de largura de faixa, principalmente em situações de alto tráfego [KL01].

A Tabela 1.3 apresenta as principais características de múltiplo acesso dos protocolos propostos na literatura.

Protocolo	Camada física	Tipo do quadro	Tamanho (duração) do quadro	Mini-janelas para contenção	Tamanho da (mini) janela de contenção	Alocação adaptativa do quadro	Resolução de contenção	Piggybacking
CRPMA/DS	FDD	fixo	7.4 ms	sim	-	sim	slotted aloha	sim
DAVIC 1.2 MAC	FDD	fixo	68 bytes	não	68 bytes	não	não especifica	não
DPRMA	FDD	fixo	-	não		sim	*slotted aloha*	não
DQRUMA	FDD	-	-	sim	fração do pacote ATM	não	*binary tree*	sim
DSA ++	FDD	variável	-	sim	¼ ou 1/8 do pacote ATM	sim	*binary tree*	sim
IEEE 802.14 MAC	FDD	-	-	sim	16 bytes	não	*ternary tree*	sim
MASCARA	TDD	variável	-	não	2 pacotes ATM	sim	*slotted aloha, stack algorithm*	não
MCSN DOCSIS MAC	FDD	variável	-	sim	variável	não	*binary exponential backoff*	sim
PRMA/DA	FDD	fixo	6 ms	não	1 pacote ATM	sim	*slotted aloha*	não
SCAMA	FDD	fixo	2.5 ms	sim	24 bits	não	*slotted aloha*	não

Tabela 1.3 – Principais características dos protocolos da literatura.

I.2.3 – Distorções e ruídos

Um sinal transmitido através de um sistema de telecomunicações sofre perturbações de várias naturezas. As distorções são alterações na forma de onda do sinal que atravessa um circuito ou um meio de transmissão, ou seja, elas são originadas do próprio sistema de comunicação. Os ruídos são sinais indesejados que aparecem somados ao sinal original, sendo independentes da natureza do sinal.

I.2.3.1 – Distorção (Atenuação)

I.2.3.1.1 – Perda de percurso

A perda de percurso é a atenuação da intensidade do sinal que ocorre sob condições de linha de visada direta LOS (*Line-Of-Sight*), devido ao espalhamento da potência. Outras perdas também podem ocorrer devido aos multipercursos causados por difração e reflexão. As perdas aumentam com a distância entre a estação rádio-base e o assinante. Isto é explicado pelo fato de, quando os sinais se propagam junto ao nível da Terra, uma grande parte da energia é absorvida por ela. A atenuação da intensidade do sinal no espaço livre é proporcional ao quadrado da distância desde a antena transmissora; este também é o caso das redes de acesso fixo sem fio, pois a maior parcela da energia se propaga em linha de visada direta. Para os sistemas móveis, o aumento dessa atenuação é proporcional (aproximadamente) à quarta potência da distância.

I.2.3.1.2 – Desvanecimento por sombra

Os terminais móveis, em geral, se movem em áreas com obstáculos de várias dimensões, como montanhas, construções e túneis. Ocasionalmente, esses obstáculos provocam sombras ou interrupção completa do sinal; as conseqüências desses efeitos de sombreamento dependem do tamanho do obstáculo e da distância até ele. Esse fenômeno não é diretamente aplicável às redes de acesso fixo sem fio, porque o receptor não se move. Entretanto, obstáculos podem interpor-se no caminho do sinal, como por exemplo, ônibus em movimento.

Minimizar os efeitos do desvanecimento por sombra em uma rede é parte do processo de planejamento da rede. Nos sistemas celulares obtêm-se resultados satisfatórios, instalando-se estações rádio-base o mais alto possível ou mais próximas umas das outras, de modo que os terminais móveis possam se comunicar "ao redor" de grandes obstáculos pela troca de estações rádio-base. Nas redes de acesso fixo sem fio, para que os assinantes "sombreados" sejam capazes de receber um sinal adequado, é preciso, no projeto do sistema, utilizar procedimentos de reflexões e refrações, instalação de repetidor, ou alinhamento do receptor com a estação rádio-base de uma área adjacente. Se um sinal é recebido por uma reflexão em um sistema de rede de acesso fixo sem fio, a intensidade do sinal cai, tipicamente, 15*dB* se comparado ao recebido por um caminho sem obstáculos.

I.2.3.1.3 – Desvanecimento por caminhos múltiplos

O desvanecimento por caminhos múltiplos é um tipo de atenuação que aparece devido à chegada de réplicas do sinal no receptor por vários caminhos. Os sinais chegam de diferentes direções, refletidos pelos objetos na vizinhança e poderão estar fora de fase quando alcançarem a

antena receptora, por percorrerem distâncias diferentes ou devido às reflexões. Nos sistemas celulares, com o movimento, a diferença de fase varia e causa um reforço ou uma contraposição aos sinais, resultando em um desvanecimento que apresenta variações de atenuação muito altas. O desvanecimento por caminhos múltiplos é menos relevante nas redes de acesso fixo sem fio. Tipicamente, onde existe um caminho LOS, há um pequeno desvanecimento por caminhos múltiplos porque os sinais refletidos tendem a ser mais fracos que o sinal LOS. Apenas onde o caminho principal está obstruído é que o desvanecimento por caminhos múltiplos se torna um problema.

Além dos obstáculos topográficos, o sinal de rádio também sofre atenuação pela heterogeneidade da atmosfera. Nesse caso, a propagação por múltiplos caminhos é causada por reflexões nas camadas atmosféricas, por refração causada pela formação de dutos ou por reflexões causadas pelo solo e pela água.

Dispositivos de correção que monitoram a continuidade do sinal recebido podem ser usados para minimizar os efeitos do desvanecimento por caminhos múltiplos.

I.2.3.2 – Ruído e interferência

Ruídos e interferências são fatores limitantes do desempenho de qualquer sistema de comunicação e, em geral, seus efeitos são similares.

I.2.3.2.1 – Ruído térmico

O ruído térmico é gerado pelo movimento de elétrons livres em um meio condutor. Em comunicação via rádio, o ruído térmico é gerado pelo ambiente atmosférico. A determinação da potência do ruído é complexa, mas verifica-se, experimentalmente, que as componentes em freqüência da potência de ruído cobrem uma ampla faixa de freqüências. O ruído térmico é um caso típico de ruído branco, pois contém, praticamente, todas as componentes do espectro de freqüência.

I.2.3.2.2 - Interferência co-canal

A interferência co-canal ocorre quando o sinal interferente ocupa a mesma faixa de freqüência do sinal recebido. O sinal interferente é produzido por irradiação proveniente do próprio sistema de rádio, quando as freqüências são usadas nas células, sem um mínimo de separação física entre as BSs aceitável, para aumentar a eficiência de reuso. A interferência co-canal não pode ser atenuada através de filtragem, e é, em geral, classificada em interferência de sobre-alcance e interferência de estação adjacente.

Interferência de sobre-alcance – Os sistemas de comunicação sem fio operam de forma bidirecional, alternando as freqüências de transmissão e recepção dos canais rádio dos dois sentidos, a cada repetidor. Assim, por exemplo, da estação origem A no sentido da estação B se transmite na freqüência f_1 e se recebe em f'_1. Da estação B para a estação C, se recebe em f_1 e se transmite em f'_1, com a finalidade de evitar que se tenha uma mesma estação de transmissão e recepção em freqüências idênticas, pois os sinais transmitidos, de potência muito superior aos recebidos, causariam interferências sobre estes últimos, mesmo com os filtros de recepção. Pode ocorrer que o sinal transmitido de A para B venha a causar interferência na

recepção de uma estação D. Este sinal interferente, por ser exatamente de mesma freqüência que o sinal recebido, não pode ser separado por filtragem.

Interferência de estação adjacente – Esta interferência decorre de se trabalhar com freqüências de recepção idênticas, numa certa estação, considerando-se os dois enlaces adjacentes. Como as antenas apresentam um valor de relação frente-costa finito há sempre recepção de uma pequena parcela da energia correspondente à ligação do enlace adjacente.

I.2.3.2.3 - Interferência de canal adjacente

A interferência no sinal desejado, causada por sinais adjacentes em freqüência, é denominada interferência de canal adjacente. Ela ocorre, principalmente, devido a produtos de intermodulação gerados nos amplificadores dos transmissores e não eliminados pelos filtros de canal dos receptores. A separação desses sinais na recepção é feita, através de filtragem, sendo mantida uma certa separação padronizada entre as freqüências utilizadas. A interferência de canal adjacente pode ser minimizada através do aperfeiçoamento de técnicas de modulação, da utilização de elaborados filtros de recepção e de uma adequada alocação de freqüências para cada usuário. Uma alocação adequada implica na utilização de faixas de freqüência não contíguas em uma mesma região geográfica.

CAPÍTULO II – MÚLTIPLO ACESSO

II.1 – RESOLUÇÃO DE CONTENÇÃO

Uma colisão ocorre em uma janela do quadro de *uplink* quando dois ou mais terminais transmitem mensagens de solicitação em uma mesma janela. As colisões são detectadas pela estação rádio-base e as mensagens de realimentação das solicitações são transmitidas de volta aos terminais. Um algoritmo de resolução de colisão CRA (*Collision Resolution Algorithm*) é empregado para resolver colisões. Ele pode ser definido como uma seqüência lógica de passos, que organiza a retransmissão de terminais colididos de tal maneira que toda mensagem seja transmitida com sucesso. Existem duas principais categorias de algoritmos de resolução de colisão: os algoritmos baseados em *aloha* como o *binary exponential backoff* e os algoritmos de divisão como o *n-ary tree* [BA01].

II.1.1 – *N-ary tree*

O algoritmo é chamado de *tree* porque o processo de divisão das colisões pode ser visualizado como uma árvore [Ca79], [Li98].
O algoritmo *n-ary tree* divide os terminais que colidiram em *n* subconjuntos. Cada um dos terminais sorteia um número aleatório entre 0 e *n-1*, que representa um dos subconjuntos e passa a fazer parte desse subconjunto selecionado. A idéia é permitir que os diferentes subconjuntos resolvam as suas colisões seqüencialmente. Os terminais do primeiro subconjunto retransmitem primeiro, enquanto os subconjuntos 1 a *n-1* esperam por sua vez.
A espera dos terminais pode ser visualizada como uma pilha virtual. A posição na pilha representa o número de janelas que o terminal tem de esperar antes de retransmitir sua solicitação. Quando a espera ocupa o nível mais baixo da pilha (= 0), o terminal pode retransmitir a sua solicitação. Se uma segunda colisão ocorre, o primeiro subconjunto é dividido novamente e a espera dos terminais, não envolvidos na colisão, é deslocada *n-1* posições acima na pilha. Se não ocorre colisão, a espera dos terminais é deslocada 1 posição abaixo na pilha.
Quando *n* é igual a 2, tem-se o *binary tree* e quando *n* é igual a 3, tem-se o *ternary tree*.

Pseudocódigo do algoritmo *ternary tree*

No terminal:
i = indica quantas janelas o terminal espera antes da sua transmissão.
se i = 0, o terminal pode transmitir na janela.

```
begin
        i ← 0;
        envia solicitação
        repeat
                recebe realimentação da estação rádio-base: sucesso, colisão ou livre;
                if houve colisão
                  then if o terminal está envolvido?
                          then i ← random[0,2] {escolhe um subconjunto}
                          else i ← i + 2
                  else i ← i – 1; {houve sucesso ou a janela estava livre}
                if i = 0
                  then envia solicitação;
         until i < 0; {processo de colisão resolvido}
end.
```

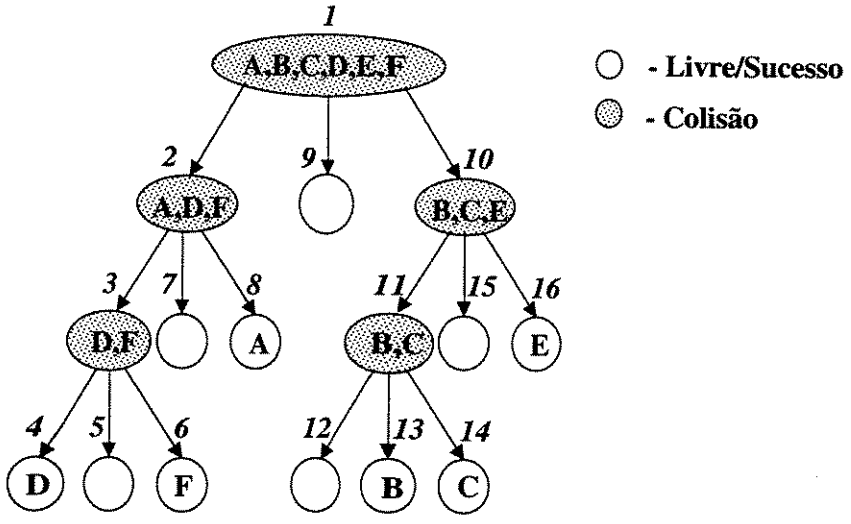


Figura 2.1 – Algoritmo *ternary tree* do tipo *depth-first*.

A Figura 2.1 mostra um exemplo de aplicação do algoritmo *ternary tree* (n=3). Inicialmente, seis terminais colidem na janela #1. A estação rádio-base não sabe a quantidade e nem a identificação dos terminais colididos [TGSF99]. Os seis terminais que colidem entre si, dividem-se em três subconjuntos, cada um selecionando um (de 0 a 2) desses subconjuntos, aleatoriamente. No exemplo, supõe-se que os terminais A, D e F se associam ao primeiro subconjunto, os de letra B, C e E se associam ao terceiro subconjunto e nenhum terminal seleciona o segundo subconjunto. Assim, o primeiro subconjunto irá transmitir na janela de número #2, e os outros subconjuntos irão esperar, até que esse primeiro esteja completamente resolvido. Para cada colisão, três novos subconjuntos são formados. Uma nova colisão ocorre na janela #2 e a divisão ternária é repetida, até que a situação seja resolvida (nesse caso, com a duração de 8 janelas). Logo em seguida, inicia-se a resolução do próximo galho da árvore na janela #9. Esse é um exemplo de resolução de colisão do tipo *depth-first* (profundidade). O intervalo de resolução de colisão CRI (*Collision Resolution Interval*) é o número de janelas utilizadas para a resolução da colisão, que, neste exemplo, é de 16 janelas. Se um outro terminal, o de letra G, por exemplo, deseja transmitir sua solicitação na janela #2, isto não será possível. Essa solicitação só poderá ocorrer na janela #17.

As Tabelas 2.1 e 2.2 exemplificam a resolução de colisão, apresentada na Figura 2.1 utilizando-se o algoritmo *ternary tree*. As colisões ocorrem nas janelas 1, 2, 3, 10 e 11.

	Terminal A	Terminal B	Terminal C	Terminal D	Terminal E	Terminal F	
Janela	**i**	**i**	**i**	**i**	**i**	**i**	**Realimentação**
#1	0	0	0	0	0	0	colisão
#2	0	2	2	0	2	0	colisão
#3	2	4	4	0	4	0	colisão
#4	4	6	6	0	6	2	sucesso: D
#5	3	5	5		5	1	livre
#6	2	4	4		4	0	sucesso: F
#7	1	3	3		3		livre

#8	0	2	2		2		sucesso: A
#9		1	1		1		livre
#10		0	0		0		colisão
#11		0	0		2		colisão
#12		1	2		4		livre
#13		0	1		3		sucesso: B
#14			0		2		sucesso: C
#15					1		livre
#16					0		sucesso: E

Tabela 2.1 – Posição de transmissão dos terminais com o algoritmo *ternary tree*.

				D		F		A					B	C		E
Janela #	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
	← Intervalo de resolução de colisão (CRI) →															

Tabela 2.2 – Intervalo de resolução de colisão com o algoritmo *ternary tree*.

II.1.2 – *Binary exponential backoff*

O propósito do algoritmo *binary exponential backoff* é espalhar os instantes de retransmissão dos terminais colididos e adaptar dinamicamente o número de terminais que estão tentando transmitir. Todos os terminais ativos transmitem na primeira janela e os terminais colididos esperam por um tempo aleatório de retransmissão. Os terminais que entraram no estado em contenção durante o processo de resolução de colisão transmitem na primeira oportunidade, ou seja, não há bloqueio para o acesso às janelas de contenção. A operação do algoritmo é descrita a seguir [SM98]:
o terminal transmite na primeira janela de contenção disponível;
se a mensagem de realimentação da estação rádio-base for de colisão, o terminal gera um número aleatório, *i*, uniformemente distribuído entre [0, 2^j–1], espera *i* janelas e retransmite o seu pedido na (*i*+1) ésima janela, onde:

j é o número de colisões que o terminal sofreu no processo de transmissão do pedido em questão;

i é o número de janelas de espera para retransmissão;

se *j* for maior que um limiar (tipicamente 10), *j* é mantido constante.

Pseudocódigo do algoritmo *binary exponential backoff*

No terminal:
i = indica quantas janelas o terminal espera antes da sua transmissão.
Se i = 0, o terminal pode transmitir na janela;
j = número de colisões que o terminal sofreu no processo de transmissão do seu pedido.

```
begin
        i ← 0;
        j ← 0; {o terminal não sofreu nenhuma colisão na transmissão do seu
                pedido}
        envia solicitação
        repeat
```

```
                recebe realimentação da estação rádio-base: sucesso, colisão ou livre;
                if houve colisão
                  then if o terminal está envolvido?
                         then j ← j + 1;
                              i ← random[0 , 2^j-1] {define o número de janelas
                                                     de espera para retransmissão}
                         else i ← i - 1
                  else i ← i – 1; {houve sucesso ou a janela estava livre}
                if i = 0
                  then envia solicitação;
          until i < 0; {processo de colisão resolvido}
end.
```

As Tabelas 2.3 e 2.4 exemplificam a resolução de colisão utilizando-se o algoritmo *binary exponential backoff* para seis terminais: A, B, C, D, E e F. O intervalo de resolução de colisão neste exemplo é de 15 janelas e as colisões ocorrem nas janelas 1, 2, 3, 5, 6 e 10.

	Terminal A		Terminal B		Terminal C		Terminal D		Terminal E		Terminal F		
Janela	**i**	**j**	**i**	**j**	**i**	**j**	**i**	**j**	**i**	**j**	**i**	**j**	**Realimentação**
#1	0	1	0	1	0	1	0	1	0	1	0	1	colisão
#2	1		1		0	2	1		0	2	0	2	colisão
#3	0	2	0	2	3		0	2	2		0	3	colisão
#4	3		1		2		0		1		6		sucesso: D
#5	2		0	3	1				0	3	5		colisão
#6	1		4		0	3			0	4	4		colisão
#7	0		3		2				8		3		sucesso: A
#8			2		1				7		2		livre
#9			1		0				6		1		sucesso: C
#10			0	4					5		0	4	colisão
#11			0						4		2		sucesso: B
#12									3		1		livre
#13									2		0		sucesso: F
#14									1		0		livre
#15									0				sucesso: E

Tabela 2.3 – Posição de transmissão dos terminais com o algoritmo *binary exponential backoff.*

				D			A		C		B		F		E
Janela #	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
	← Intervalo de resolução de colisão (CRI) →														

Tabela 2.4 – Intervalo de resolução de colisão no algoritmo *binary exponential backoff.*

II.1.3 – Comparação dos algoritmos de resolução de contenção

Os algoritmos de resolução de contenção, em geral, são comparados supondo-se que a transmissão de informação é realizada através da contenção. Assim, a comparação entre os algoritmos é sempre realizada para diferentes situações de cargas nos terminais [BMN96], [GMPS97], [MB88].
Neste trabalho, o algoritmo de contenção é usado para solicitação de reserva e, portanto, é nesse contexto que deve ser analisado. Assim, a simulação de Monte Carlo (Matlab [MATLAB5.3]) é utilizada para selecionar o algoritmo mais apropriado, variando-se o número de terminais que disputam uma única janela de contenção.

A Figura 2.2 mostra a comparação entre o *binary tree* (linha contínua) e o *ternary tree* (linha tracejada) usando simulação de Monte Carlo com 200 amostras. Pode-se observar que o desempenho do *binary* é similar ao *ternary*. As curvas superiores apresentam o número médio de janelas necessárias, dividido pelo número de terminais, para resolução da contenção. Por exemplo, com 10 terminais, o número médio de janelas necessárias para resolução de contenção é de 27. As curvas inferiores apresentam o número médio, dividido pelo número de terminais, de janelas dispendidas por terminal para conseguir um acesso com sucesso. Por exemplo, com 10 terminais disputando a janela de contenção, cada terminal gastaria, em média, 16 janelas para conseguir um acesso bem sucedido.

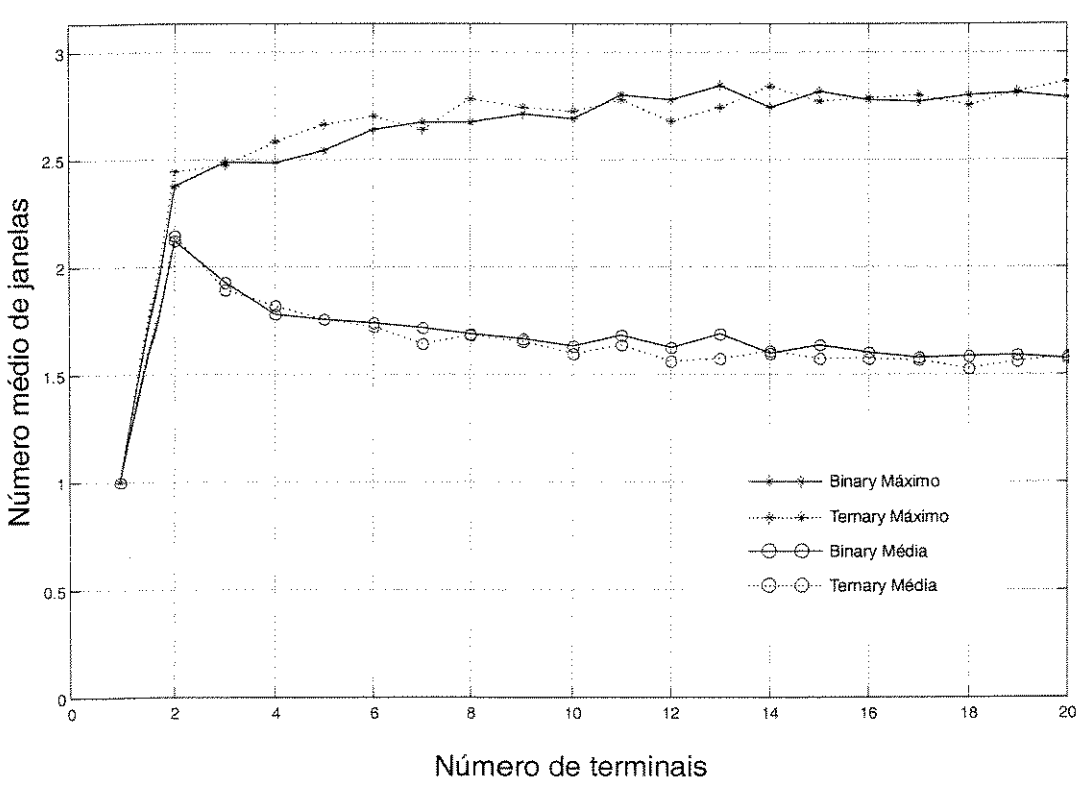


Figura 2.2 – Comparação *binary tree* versus *ternary tree*

A Figura 2.3 mostra a comparação entre o *binary exponential backoff* (linha contínua) e o *ternary tree* (linha tracejada) usando simulação de Monte Carlo com 200 amostras. As curvas superi-

ores apresentam o número médio de janelas necessárias para resolução da contenção, dividido pelo número de terminais. Por exemplo, com 10 terminais, o número médio de janelas necessárias para resolução de contenção, no *ternary tree* é de 28, enquanto no *backoff* esse número cresce, como mostra a Figura 2.3, para 2 terminais, o número médio é de 5.8 janelas, para 10 terminais é de 51 janelas. As curvas inferiores apresentam o número médio, dividido pelo número de terminais, de janelas dispendidas por terminal para conseguir um acesso com sucesso. Por exemplo, quando 10 terminais disputam uma janela de contenção, no *ternary tree*, eles gastam, em média, 16 janelas para conseguir um acesso bem sucedido, enquanto no *backoff*, eles gastam 23 janelas para conseguir um acesso com sucesso.

Pode-se, portanto, concluir pela superioridade dos algoritmos *tree* sobre o algoritmo *backoff*. Do ponto de vista do número médio de janelas necessárias para a resolução da contenção, os algoritmos *binary tree* e *ternary tree* têm desempenho similar.

Esta conclusão se aplica quando esses algoritmos são usados como mecanismo de reserva e não para transmissão de informação.

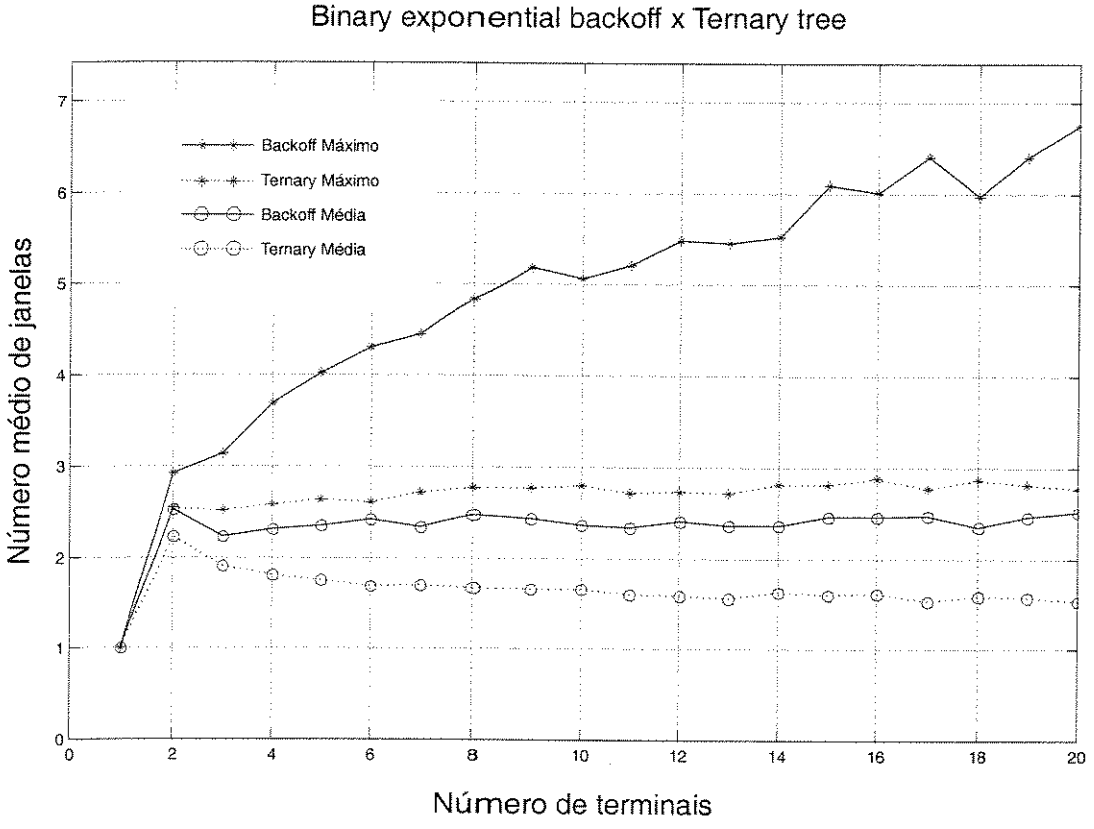


Figura 2.3 – Comparação binary exponential backoff versus ternary tree

A Figura 2.4 ilustra os algoritmos *binary tree* e *ternary tree* considerando-se erros (10%) aleatórios na transmissão. Os erros são tratados pelos algoritmos, na estação rádio-base, como colisão e, portanto, há uma degradação do desempenho. Observa-se que o número médio de janelas necessárias para resolução da contenção aumentou (vide Figura 2.2).

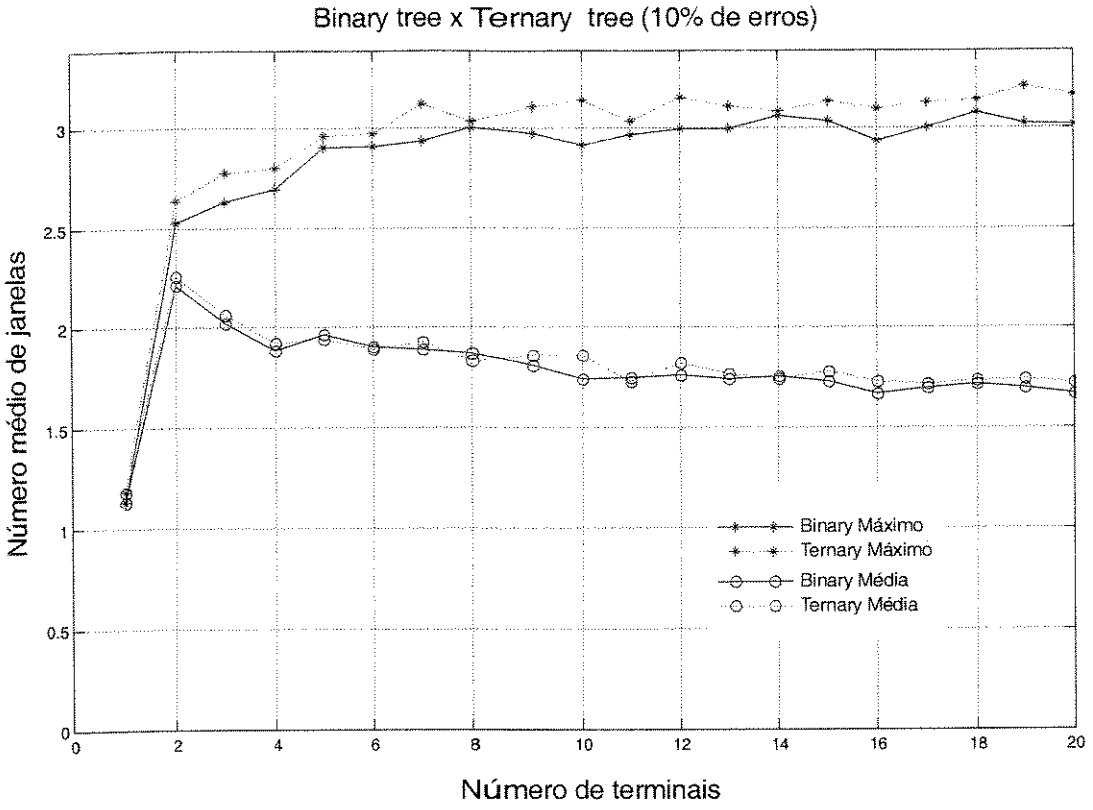


Figura 2.4 – Comparação *binary tree* versus *ternary tree* (10% de erros)

Um aspecto que distingue os algoritmos *binary* e *ternary trees* é o número médio de janelas livres na resolução da contenção. Quando nenhum terminal transmite, não há potência gerada no canal e, portanto, não se produz interferências nos sistemas vizinhos co-canais.

O número médio de janelas livres e de colisão para a resolução da contenção foi também obtido nas simulações para os algoritmos *binary* e *ternary trees*. Os resultados, em porcentagens relativas ao número médio de janelas necessárias para a resolução de contenção, são mostrados nas Figuras 2.5 (*binary*) e 2.6 (*ternary*). Em média, o número de janelas necessárias para a resolução da contenção é a soma do número de janelas de sucesso, do número médio de janelas de colisão e do número médio de janelas livres.

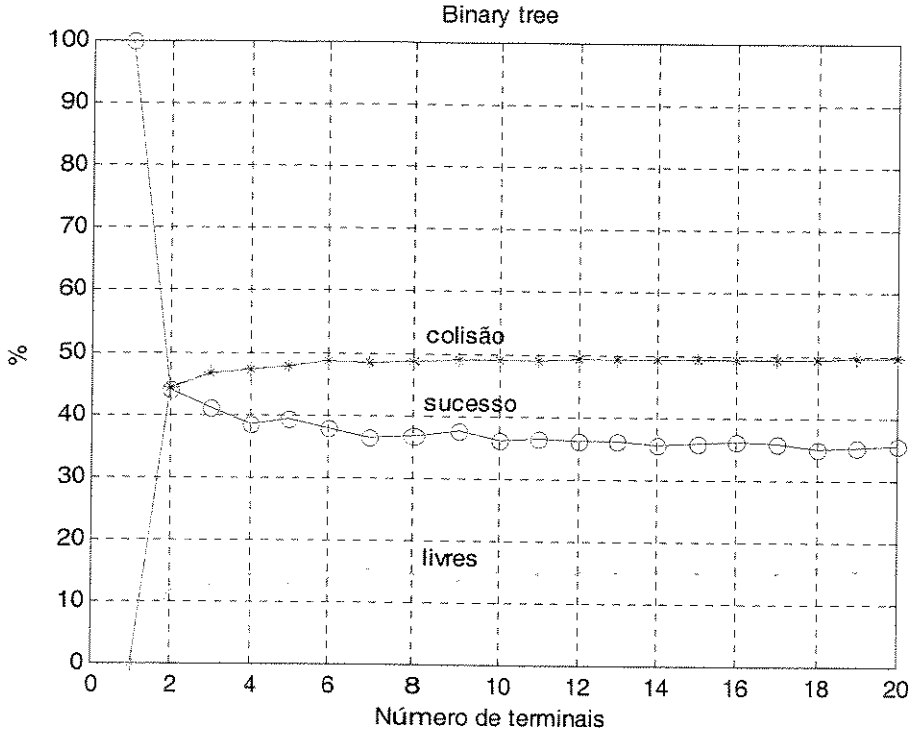


Figura 2.5 – Percentual das janelas de sucesso, livres e de colisão no *binary tree.*

Observa-se que no algoritmo *binary tree*, o percentual do número médio de janelas livres é da ordem de 15%, o percentual do número médio de janelas de colisão é de 49% e o percentual do número médio de janelas de sucesso é de 36%.

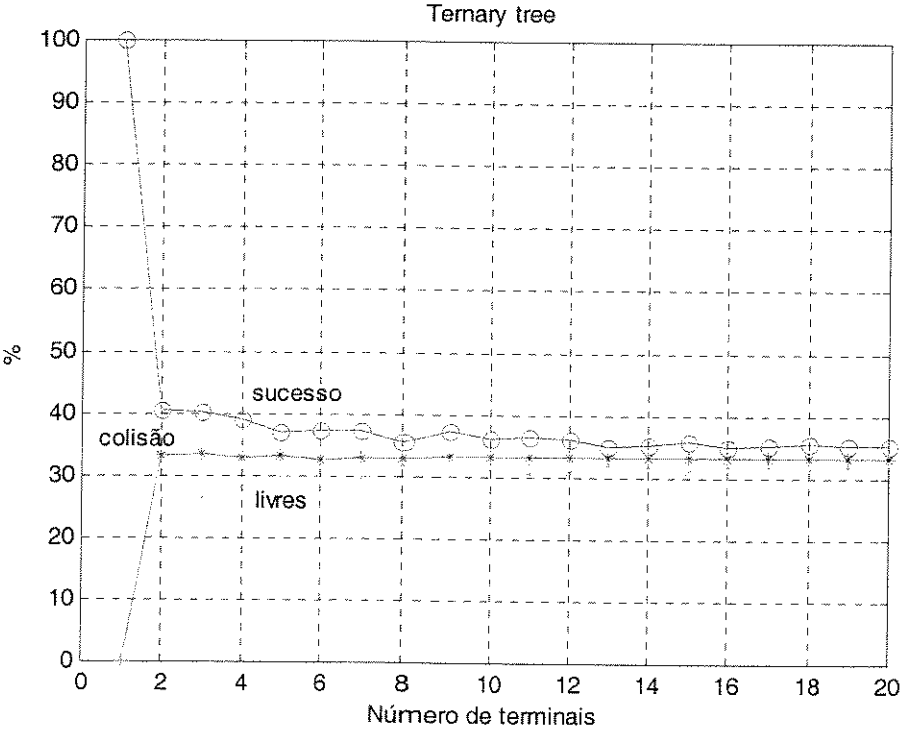


Figura 2.6 – Percentual das janelas de sucesso, livres e de colisão no *ternary tree.*

Observa-se que, no algoritmo *ternary tree,* o percentual do número médio de janelas livres é da ordem de 31%, o percentual do número médio de janelas de colisão é de 33% e o percentual do número médio de janelas de sucesso é de 36%.

O número médio de janelas livres no *ternary tree* é significativamente superior ao do *binary tree* (31% e 15%, respectivamente). Assim, considerando-se o aspecto interferência, conclui-se pela opção do *ternary tree.*

A simulação de evento discreto foi usada para validar os resultados obtidos na simulação de Monte Carlo com a utilização dos algoritmos de resolução de contenção *binary tree* e *ternary tree.* Variou-se o número de terminais (3, 5 e 10) na disputa de uma única janela de contenção. As Tabelas 2.5 e 2.6 mostram, respectivamente, o número médio de janelas dispendidas por terminal para se obter uma transmissão com sucesso e o número médio de janelas necessárias para a resolução da contenção.

	Simulação Monte Carlo	**Simulação Evento Discreto**
Algoritmos de resolução	200 amostras	200 amostras
Binary tree com 3 terminais	5.72	5.96
Ternary tree com 3 terminais	5.74	5.52
Binary tree com 5 terminais	9.08	9.30
Ternary tree com 5 terminais	8.48	8.16
Binary tree com 10 terminais	16.83	16.93
Ternary tree com 10 terminais	15.98	15.86

Tabela 2.5 – Número médio de janelas dispendidas por terminal para se obter uma transmissão com sucesso

	Simulação Monte Carlo	**Simulação Evento Discreto**
Algoritmos de resolução	200 amostras	200 amostras
Binary tree com 3 terminais	7.33	7.57
Ternary tree com 3 terminais	7.50	7.45
Binary tree com 5 terminais	13.19	13.50
Ternary tree com 5 terminais	12.76	12.18
Binary tree com 10 terminais	27.71	27.76
Ternary tree com 10 terminais	26.91	26.63

Tabela 2.6 – Número médio de janelas necessárias para a resolução da contenção.

Os algoritmos *binary* e *ternary trees* podem ser usados, concomitantemente, para dar prioridade a determinados terminais na resolução da contenção. Os terminais que utilizam o *binary tree* escolhem aleatoriamente entre 0 e 1 e os que utilizam o *ternary tree* escolhem entre 0, 1 e 2. Portanto, os terminais que utilizam o *binary tree*, em média, serão atendidos prioritariamente. A Figura 2.7 mostra um exemplo, considerando-se metade dos terminais com tráfego em tempo real rt (*real time*) utilizando *binary tree* e outra metade, com tráfego não em tempo real nrt (*non real time*), utilizando *ternary tree.*

A curva superior mostra o número médio de janelas dispendidas pelos terminais com tráfego não em tempo real que utilizam o *ternary tree.* A curva inferior mostra o número médio de janelas dispendidas pelos terminais com tráfego em tempo real que utilizam o *binary tree.* A curva

central mostra o número médio de janelas dispendidas, quando todos os terminais utilizam o *ternary tree*. Quando 10 terminais disputam a janela de contenção, cada terminal gasta, em média, com tráfego em tempo real, 10 janelas para conseguir um acesso bem sucedido e com tráfego não em tempo real, 23 janelas.

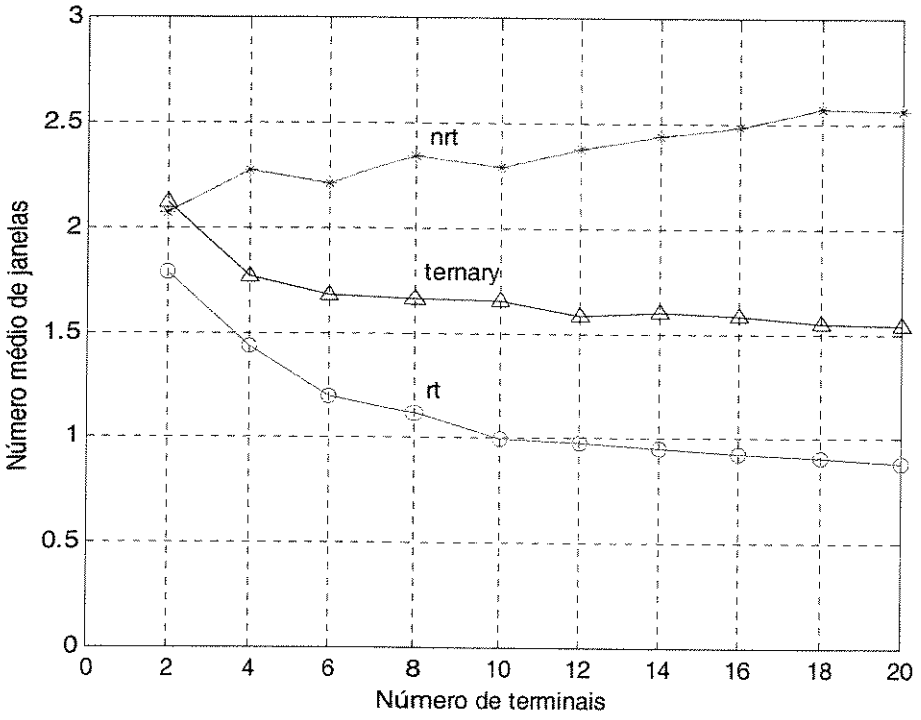


Figura 2.7 – Número médio, dividido pelo número de terminais, de janelas dispendidas por terminal para se obter uma transmissão com sucesso.

II.2 – AGENDAMENTO DAS JANELAS DE RESERVA

As soluções de agendamento são necessárias quando um conjunto de clientes precisa ser atendido por um único recurso. O agendamento é uma regra que seleciona o próximo cliente a ser atendido pelo recurso e tem por objetivo minimizar o tempo de espera de atendimento dos clientes.

II.2.1 – Agendamento cíclico

A disciplina de agendamento cíclico das janelas de reserva consiste em atender uma célula (peso 1) de cada terminal por ciclo. Na estação rádio-base, um vetor denominado *reserva* contém o número de janelas reservadas por cada terminal, que é atualizado quando a célula liberada pelo terminal para participar do processo de contenção é transmitida com sucesso.

O vetor *reserva* é percorrido ciclicamente e se existir reserva para um dos terminais, uma célula é liberada pelo *buffer* desse terminal para ocupar a janela de reserva. O índice utilizado para percorrer o vetor *reserva* é incrementado unitariamente e determina o próximo terminal a ser atendido no próximo ciclo. Caso não exista nenhuma reserva, uma célula vazia ocupa a janela de reserva. O processo é repetido até esgotar-se o número total de janelas de reserva do quadro.

Pseudocódigo do agendamento cíclico (peso 1)

```
r = número de janelas de reserva no quadro;
nt = número de terminais;
reserva = vetor reserva dos terminais (nt);
indice = índice que percorre o vetor reserva dos terminais;
temreserva = sinaliza se existe ou não reserva para utilização da janela de reserva;
i, j = variáveis auxiliares;

begin
    input r; {número de janelas de reserva do quadro}
    indice ← 0;
    for i ← 1 to r {utiliza as janelas de reserva do quadro}
        temreserva ← 0;
        j ← 1;
        while j <= nt {percorre a variável reserva para todos os terminais}
            if indice = nt
                then indice ← 1
                else indice ← indice + 1;
            if reserva[indice] > 0
                then libera célula de reserva do terminal para utilizar a janela
                     de reserva;
                     temreserva ← 1;
                     reserva[indice] ← reserva[indice] – 1;
                     j ← nt;
            j ← j + 1;
        end;
        if temreserva = 0 {se não houve liberação de célula normal}
            then libera célula vazia para ocupar a janela livre (de reserva);
    end;
end.
```

Um exemplo de agendamento cíclico peso 1 para sete quadros é mostrado na Figura 2.8. O quadro tem um tamanho fixo de quatro janelas, onde a primeira é de contenção e as demais são de reserva. Após a transmissão dos quadros (n a n+6), o vetor *reserva* dos cinco terminais possui o conteúdo mostrado na Tabela 2.7.

$\boxed{A_{(2)}}$ = janela com sucesso (o terminal A transmitiu com sucesso e reservou 2 janelas)

	C	R	R	R
quadro n:		A	C	E
quadro n+1:	$A_{(1)}$	B	C	E
quadro n+2:		A	C	E
quadro n+3:		C	C	C
quadro n+4:	$D_{(1)}$			
quadro n+5:	$A_{(2)}$	D		
quadro n+6:		A	A	

Figura 2.8 – Agendamento cíclico na transmissão de sete quadros.

Terminal	Número de janelas reservadas após a transmissão do quadro						
	quadro n	quadro n+1	quadro n+2	quadro n+3	quadro n+4	quadro n+5	quadro n+6
A	0	1	0	0	0	2	0
B	1	0	0	0	0	0	0
C	5	4	3	0	0	0	0
D	0	0	0	0	1	0	0
E	2	1	0	0	0	0	0

Tabela 2.7 – Conteúdo do vetor *reserva* após a transmissão de sete quadros.

II.2.2 – *Piggybacking*

Piggybacking é o mecanismo usado pelos terminais para comunicar à estação rádio-base sua intenção de transmitir durante o próximo quadro (ou o número necessário de janelas para transmissão). É um mecanismo de reserva de quadros, para transmissão de pacotes, que complementa o procedimento de contenção. O procedimento de *piggybacking* consiste em enviar solicitações para transmissão adicional de dados, controlando um ou mais bits, dependendo do protocolo utilizado, para fazer uso dos próximos quadros.

Pseudocódigo do *piggybacking*

A informação de *piggybacking* é usada na transmissão de células por contenção e por reserva.

r = número de janelas de reserva no quadro;
c = número de janelas de contenção no quadro;
nt = número de terminais;
reserva = vetor reserva dos terminais (nt);
indice = índice que percorre o vetor reserva dos terminais;
quantbuf/agenda = vetor que registra o número de células no *buffer* do terminal;
indice = índice que percorre o vetor reserva dos terminais (agendamento cíclico);
identificacao = variável que identifica o terminal;
i = variável auxiliar

```
begin
        input  r; {número de janelas de reserva do quadro}
        input c; { número de janelas de contenção do quadro}
        indice ← 0;
        for i ← 1 to r  {utiliza as janelas de reserva do quadro}
                {verifica se existe reserva para todos os terminais}
                        if indice = nt
                                then indice ← 1
                                else indice ← indice + 1; {agendamento cíclico – peso1}
                        if reserva[indice] > 0
                                then {libera célula de reserva do terminal para utilizar a
                                      janela de reserva};
                                     reserva[indice] ← reserva[indice] – 1;
                                     reserva[indice] ← quantbuf[indice]; {piggybacking}
```

```
        end; {r}

        for i ← 1 to c {utiliza as janelas de contenção do quadro}
                {verifica o uso da janela de contenção}
                if {a janela pode ser usada para contenção}
                        then {o terminal transmite e utiliza a janela de contenção;
                                {atualiza vetor agenda }
                        else {a janela está em processo de contenção}
                                if {o terminal está em contenção na janela}
                                        then {o terminal transmite na janela};
                                                {atualiza vetor agenda};
                if {houve colisão}
                        then {aplica o algoritmo de resolução de contenção}
                        else {faz agendamento/piggybacking}
                                reserva[identificacao] ← reserva[identificacao] +
                                                        agenda[identificacao];
        end; {c}
end.
```

Um exemplo de *piggybacking* para dez quadros é mostrado na Tabela 2.8. O quadro tem um tamanho fixo de cinco janelas; as quatro primeiras são de reserva e a última é de contenção. Na transmissão de 10 quadros (n a n+9), o vetor *reserva*, para os dez terminais, possui o conteúdo mostrado na Tabela 2.8.

[ ] = ocorrência de *piggybacking*

Terminal	**Número de janelas reservadas após a transmissão do quadro**									
	quadro n	quadro n+1	quadro n+2	quadro n+3	quadro n+4	quadro n+5	quadro n+6	quadro n+7	quadro n+8	quadro n+9
A	0	2	2	1	0	0	0	0	0	0
B	3	2	2	5	4	3	3	2	1	1
C	5	4	4	3	2	2	1	3	3	2
D	0	0	0	0	0	0	0	0	0	0
E	0	0	0	4	4	3	2	1	1	3
F	1	1	0	0	0	0	0	0	6	5
G	10	10	9	8	8	7	10	10	9	8
H	4	3	4	4	3	2	1	1	0	0
I	2	1	0	0	0	0	0	0	0	0
J	0	0	0	0	0	0	2	1	0	0

Tabela 2.8 – Conteúdo do vetor *reserva* após a transmissão de dez quadros utilizando *piggybacking*.

Observa-se que na passagem do quadro *n* para o *n+1* os terminais B, C H e I transmitiram nas janelas de reserva e o terminal A transmitiu na de contenção e informou via *piggybacking* 2 células para transmissão.
Observa-se ainda que, entre os quadros *n+1* e *n+2*, os terminais F, G, H e I transmitiram por reserva e o terminal H informou, via *piggybacking*, a existência de mais duas células.

II.2.3 – Agendamento cíclico com prioridade

A disciplina de agendamento cíclico com prioridade das janelas de reserva consiste em atender uma célula de cada terminal de um determinado tipo de tráfego por quadro. As prioridades são atribuídas de acordo com a relação de prioridade: CBR > rtVBR > nrtVBR, vide Figura 2.9.

Na estação rádio-base, o vetor *reserva* contém o número de janelas reservadas por cada terminal, que é atualizado em duas situações nos casos VBR: quando a célula liberada pelo terminal para participar do processo de contenção é transmitida com sucesso (agendamento) e quando a célula é liberada pelo terminal para utilizar a janela de reserva (*piggybacking*).
No caso CBR, o vetor *reserva* conterá um elemento a cada 1/PCR unidades de tempo. Quando uma janela de reserva é usada por um terminal CBR, o *timestamp* (instante de chegada no *buffer*) da célula é usado para a determinação da próxima inserção de um elemento no vetor *reserva*.

O vetor *reserva* é percorrido ciclicamente e se existir reserva para um dos terminais, de um determinado tipo de tráfego a ser atendido no quadro, uma célula é liberada pelo *buffer* desse terminal para ocupar a janela de reserva. O índice utilizado para percorrer o vetor *reserva* é distinto para cada tipo de tráfego e é incrementado unitariamente. O índice determina o próximo terminal a ser atendido. O processo é repetido até esgotar-se o número total de janelas de reserva do quadro conforme mostrado na Tabela 2.9.

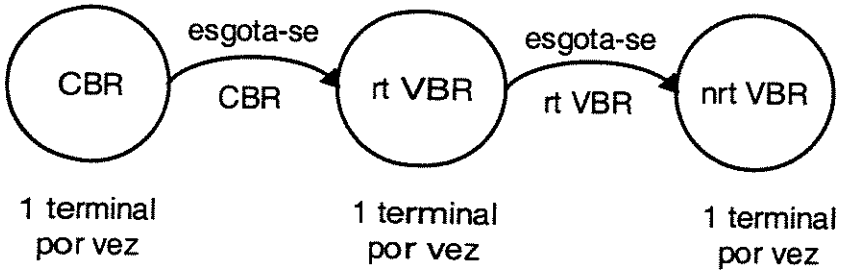


Figura 2.9 – Agendamento cíclico com prioridade.

Pseudocódigo do agendamento cíclico com prioridade

r = número de janelas de reserva no quadro;
ntcbr = número de terminais com tráfego CBR;
ntrt = número de terminais com tráfego rtVBR;
ntnrt = número de terminais com tráfego nrtVBR;
reserva = vetor reserva dos terminais (nt);
indcbr = índice que percorre o vetor reserva dos terminais CBR;
indrt = índice que percorre o vetor reserva dos terminais rtVBR;
indnrt = índice que percorre o vetor reserva dos terminais nrtVBR;
tem = indica que existe tráfego CBR (=0), rtVBR (=1) ou nrtVBR (=2);
i,k = variáveis auxiliares.

```
begin
    input r; {número de janelas de reserva do quadro para atender um determinado
              tipo de tráfego}
    input ntcbr; {número de terminais com tráfego CBR}
    input ntrt; {número de terminais com tráfego rtVBR}
```

```
input  ntnrt; {número de terminais com tráfego nrtVBR}
input  tem; {indica o tipo de tráfego existente}
indcbr ← 0;
indrt ← ntcbr;
indnrt ← ntcbr + ntrt;

for i ← 1 to r  {utiliza as janelas de reserva do quadro}
        if tem = 0
                then k ← 1;
                     while k <= ntcbr  {percorre a variável reserva para todos os
                                        terminais CBR}
                       if indcbr = ntcbr
                              then indcbr ← 1
                              else indcbr ← indcbr + 1;
                       if reserva[indcbr] > 0
                              then libera célula do terminal para utilizar a janela de
                                   reserva;
                                   reserva[indcbr] ← reserva[indcbr] – 1;
                                   reserva[indcbr] ← quantbuf[indcbr]; {piggyback}
                                   k ← ntcbr;
                       k ← k + 1;
                     end
                else if tem = 1
                       then k ← 1;
                            while k <= ntrt  {percorre a variável reserva para
                                              todos os terminais rtVBR}
                              if indrt = ntcbr + ntrt
                                     then indrt ← ntcbr + 1
                                     else indrt ← indrt + 1;
                              if reserva[indrt] > 0
                                     then libera célula do terminal para utilizar a ja
                                          nela de reserva;
                                          reserva[indrt] ← reserva[indrt] – 1;
                                          reserva[indrt] ← quantbuf[indrt]; {piggy}
                                          k ← ntrt;
                              k ← k + 1;
                            end
                       else k ← 1;
                            while k <= ntnrt  {percorre a variável reserva para
                                               todos os terminais nrtVBR}
                              if indnrt =  ntcbr + ntnrt + ntrt
                                     then indnrt ← ntcbr + ntrt + 1
                                     else indnrt ← indnrt + 1;
                              if reserva[indnrt] > 0
                                     then libera célula do terminal para utilizar a ja
                                          nela de reserva;
                                          reserva[indnrt] ← reserva[indnrt] – 1;
                                          reserva[indnrt] ← quantbuf[indnrt];
                                          k ← ntrt;
```

```
                k ← k + 1;
            end;

        end;
end.
```

Um exemplo de agendamento cíclico com prioridade é mostrado na Tabela 2.9. O quadro tem tamanho variável e o número de janelas de reserva do quadro é igual ao número de terminais, de um determinado tipo de tráfego, que possuem reserva agendada. Inicialmente os terminais CBR são atendidos, depois os terminais rtVBR, e por último, os terminais nrtVBR. Os terminais A e B possuem tráfego CBR, os terminais C a F possuem tráfego rtVBR e os demais (G a J), tráfego nrtVBR.

		Número de janelas reservadas após a transmissão do quadro							
Terminais		quadro n	quadro n+1	quadro n+2	quadro n+3	quadro n+4	quadro n+5	quadro n+6	quadro n+7
CBR	**A**	1	0	0	0	0	1	0	0
	B	0	1	0	0	0	0	0	1
rtVBR	**C**	0	0	1	0	0	0	0	0
	D	1	1	1	0	0	0	0	0
	E	0	2	2	1	1	0	0	0
	F	0	0	0	0	0	0	0	0
nrtVBR	**G**	1	1	1	1	1	1	1	0
	H	2	2	2	2	2	2	2	1
	I	0	0	0	4	4	4	4	3
	J	0	0	0	0	1	1	1	5

Tabela 2.9 – Conteúdo do vetor *reserva* após a transmissão de oito quadros utilizando-se agendamento cíclico com prioridade.

Observa-se que na passagem do quadro *n* para o *n+1* o terminal A transmitiu na única janela de reserva disponível do quadro e determinou-se o instante de se inserir 1 elemento em seu vetor *reserva* (quadro *n+5*). O terminal E transmitiu na janela de contenção e informou, via *piggybacking*, 2 células. Inseriu-se 1 elemento no vetor *reserva* do terminal B.

Na passagem do quadro *n+1* para o *n+2*, o terminal B transmitiu na única janela de reserva do quadro e determinou-se o instante de se inserir 1 elemento em seu vetor *reserva* (quadro *n+7*). O terminal C transmitiu na janela de contenção e informou, via *piggybacking*, 1 célula.

Na passagem do quadro *n+2* para o *n+3*, os terminais C, D e E transmitiram nas 3 janelas de reserva do quadro e o terminal I transmitiu na janela de contenção e informou 4 células.

Na passagem do quadro *n+3* para o *n+4*, o terminal E transmitiu na única janela de reserva do quadro e informou, via *piggybacking*, a existência de mais 1 célula. O terminal J transmitiu na janela de contenção e informou 1 célula.

Na passagem do quadro *n+4* para o *n+5*, o terminal E transmitiu na única janela de reserva do quadro e inseriu-se 1 elemento no vetor *reserva* do terminal A.

Na passagem do quadro *n+5* para o *n+6*, o terminal A transmitiu na única janela de reserva do quadro e determinou-se o instante de se inserir 1 elemento em seu vetor *reserva* (quadro *n+10*).

Na passagem do quadro *n+6* para o *n+7*, os terminais G, H, I e J transmitiram nas 4 janelas de reserva do quadro e o terminal J informou, via *piggybacking*, a existência de mais 5 células. Inseriu-se 1 elemento no vetor *reserva* do terminal B.

II.3 – ALOCAÇÃO DINÂMICA DE JANELAS DE CONTENÇÃO NO QUADRO

O propósito do algoritmo é alocar dinamicamente as janelas de contenção no quadro. O número de janelas no quadro é fixo e no início de cada quadro, o algoritmo fornece o número de janelas de reserva e o número de janelas de contenção do quadro corrente. As janelas de reserva são alocadas no início do quadro e as janelas de contenção, no final do quadro. O algoritmo é separado em três ações: utilização das janelas de reserva; utilização das janelas de contenção e determinação de eliminar e propor janela de contenção.

Na primeira ação, é realizado o agendamento cíclico de utilização das janelas de reserva.
Na segunda ação, utilização das janelas de contenção, o algoritmo verifica quais janelas de contenção estão em processo de resolução de colisão e quais estão liberadas para participarem de um novo processo de contenção. Os terminais transmitem na nova janela de contenção se possuem célula para transmitir, não possuem reserva agendada e não estão participando de outro processo de contenção. Os terminais que estão participando de um processo de resolução de contenção formam um grupo. E eles saem desse grupo quando obtêm sucesso. A estação rádio-base é capaz de prever quando o grupo se extingue. Os terminais que participam de um grupo utilizam a janela de contenção quando sua variável pilha é igual a zero.

Na terceira ação, determinação de eliminar ou propor uma nova janela de contenção, o número de terminais com reserva agendada (*ntr*) e o número de células reservadas (*ncr*) são computados a cada quadro.

O algoritmo elimina uma janela de contenção no próximo quadro quando se resolve uma contenção completamente no quadro corrente. Para estimar o fim do processo, a estação rádio-base comporta-se como um terminal que participa do processo de contenção, mas não participa da colisão, isto é, ela soma 2 na posição da pilha virtual em cada colisão.

O algoritmo não propõe uma nova janela de contenção no próximo quadro quando todos os terminais conectados têm reserva agendada no quadro corrente.

O algoritmo propõe uma nova janela de contenção no próximo quadro sempre que, no quadro corrente, o número de células reservadas for menor que o número de janelas de reserva.

Se no quadro corrente houve proposição de uma nova janela de contenção e houve uso desta janela, uma nova janela de contenção é proposta no próximo quadro e a variável *espera*, que controla o número de quadros sem proposição de janela, é zerada.

Se não houve uso da janela de contenção proposta no quadro corrente e se o número de células reservadas no quadro é maior do que o número de janelas de reserva do quadro, a janela de contenção é eliminada no próximo quadro e a variável *espera* é incrementada unitariamente. Este é um mecanismo de desestímulo que é implementado usando as variáveis *espera* e *propor*.

Quando a nova janela proposta é usada no quadro corrente, uma variável *propor* recebe o conteúdo da variável *espera*. Se no quadro corrente não houve proposição de janela nova, a variável *propor* é decrementada de um, e no quadro que o valor dessa variável chega a zero, propõe-se uma nova janela de contenção para o quadro seguinte.

Nunca se propõe mais que uma nova janela de contenção por quadro. Se é necessário propor uma nova janela de contenção no próximo quadro, mas esgotou-se o número de janelas do quadro, a variável *propor* recebe o valor 1 para que se tenha a oportunidade de propor uma nova janela de contenção no quadro seguinte.

Pseudocódigo do algoritmo de alocação dinâmica de janelas de contenção

tq = total de quadros;
tj = total de janelas no quadro;
r = número de janelas de reserva ;
c = número de janelas de contenção do quadro;
nt = número de terminais;
pilha = contém a ordem de transmissão das células; representa o número de quadros que o terminal tem de esperar antes de retransmitir sua solicitação (c x nt+1);
reserva = vetor reserva dos terminais (nt);
nbuf = vetor que registra a existência de células nos *buffers* dos terminais (nt);
ntjan = vetor que identifica qual janela de contenção o terminal está usando no processo de contenção (nt);
liberater = vetor que identifica o terminal que pode ser liberado para participar de outro processo de contenção (nt);
colisao = vetor que identifica colisão na janela de contenção (c);
jan = vetor que registra o uso da janela de contenção (c): 0=sem uso, 1=em contenção e 2=contenção resolvida;
indice = índice que percorre o vetor reserva dos terminais (agendamento cíclico);
temreserva = indica se existe ou não reserva para utilização da janela de reserva;
vazia = identifica existência de janela livre (de colisão);
uso = identifica o uso da janela de contenção;
fresh = nova janela de contenção proposta;
espera = controla o número de quadros sem proposição de janela de contenção;
ntr = número de terminais com reserva agendada no quadro;
ncr = número de células reservadas por quadro;
propor, i, j, k, l = variáveis auxiliares.

begin
input tq; {125000}
input tj; {5}
input nt; {10}
c ← 1; fresh ← c; r ← tj – c; indice ← 0; espera ← 0; propor ← espera; vazia ← 0;
pilha[1:c,1:nt+1] ← -1;
ntjan[1:nt] ← 0;
liberater[1:nt] ← 0;
jan[1:c] ← 0;
colisao[1:c] ← 0;
for i ← 1 **to** tq {para todos os quadros}

```
for j ← 1 to r {utiliza as janelas de reserva do quadro}
        temreserva ← 0;
        k ← 1;
        while k <= nt {verifica existência de reserva nos terminais}
        if indice = nt {atendimento cíclico: peso1}
                then indice ← 1
                else indice ← indice + 1;
        if reserva[indice] > 0
                then temreserva ← 1;
                     libera célula de reserva do terminal para utilizar a janela de
                     reserva;
                      reserva[indice] ← reserva[indice] – 1;
                      k ← nt;
        k ← k + 1;
        end;
        if temreserva = 0
                then libera célula vazia para ocupar a janela livre (de reserva);
end; {r}

for j ← 1 to c {utiliza as janelas de contenção do quadro}
        if jan[j] = 0 {a janela pode ser usada para contenção}
                then uso ← 0;
                        for k ← 1 to nt
                            if reserva[k] = 0 and ntjan[k] = 0 and nbuf[k] > 0
                             {o terminal não tem célula de reserva agendada, não
                              está participando de um processo de contenção e
                              possui célula para transmitir}
                                    then o terminal transmite e utiliza a janela de
                                         contenção;
                                          ntjan[k] ← j;  pilha[j,k] ← 0;  uso ← 1;
                                          vazia ← 0;
                     end;
                     jan[j] ← uso;
                     if uso = 1
                             then pilha[j,nt+1] ← 0 {a BS comporta-se co
                                                     mo um terminal}
                else {a janela está em processo de contenção}
                     vazia ← 1;
                     for k ← 1 to nt
                             if ntjan[k] = j and pilha[j,k] = 0
                             {o terminal está em contenção na janela e transmite
                              se pilha=0}
                                     then o terminal transmite na janela;
                                          vazia ← 0;
                     end;
        if jan[j] = 0
                then libera célula vazia para ocupar a janela livre (de tráfego)
                else if vazia = 1
                        then libera célula vazia para ocupar a janela livre (de
                             colisão);
```

```
        verifica colisão na janela;  {colisão(j): =0 ou =1}

        if colisao[j] = 1  {houve colisão}
                then for k ← 1 to nt  {determina quem aguarda (+2) e quem
                                       transmite no próximo quadro (0, 1 ou 2)}
                        if pilha[j,k] = 0
                                then pilha[j,k] ← random(0,1,2)
                                else if pilha[j,k] > 0
                                        then pilha[j,k] ← pilha[j,k] + 2;
                        end;
                        pilha[j,nt+1] ← pilha[j,nt+1] + 2
                else if jan[j] = 1  {houve sucesso e não é janela livre (de tráfego)}
                        then   faz agendamento;
                                for k ← 1 to nt
                                        if pilha[j,k] >= 0  {decrementa a pilha}
                                                then pilha[j,k] ← pilha[j,k] –1;
                                                     if pilha[j,k] < 0
                                                       then liberater[k] ← 1;
                                end;
                                if pilha[j,nt+1] >= 0
                                        then pilha[j,nt+1] ← pilha[j,nt+1] –1;
                                if pilha[j,nt+1] < 0 {verifica término do processo de
                                                      contenção}
                                        then jan[j] ← 2;
end;  {c}

{liberação do terminal após final de quadro para participar de outro processo}
for k ← 1 to nt
        if liberater[k] = 1  {se o terminal pode ser liberado}
                then ntjan[k] ← 0;
                     liberater[k] ← 0;
end;

{cômputo de ncr e ntr}
ntr ← 0;
ncr ← 0;
for k ← 1 to nt
        if reserva[k] > 0  {se há reserva para o terminal}
                then ntr ← ntr + 1;
                     ncr ← ncr + reserva[k];
end;

j ← 1;
while j <= c
        if jan[j] = 2 and j <> fresh
                {elimina uma janela quando se resolve uma contenção e não é nova
                 janela}
                then for k ← j to c-1
                        colisao[k] ← colisao[k+1];
```

```
                        jan[k] ← jan[k+1];
                        for l ← 1 to nt+1
                            pilha[k,l]← pilha[k+1,l];
                        end;
                    end;
                    for k ← 1 to nt
                            if ntjan[k] > j
                                    then ntjan[k] ← ntjan[k] –1;
                            pilha[c,k] ← -1;
                    end;
                    pilha[c,nt+1] ← -1;
                    jan[c] ← 0;  colisao(c) ← 0;  c ← c – 1;  r ← tj – c;
                    if fresh > 0
                            then fresh ← fresh - 1
            else j ← j + 1;
end;

if fresh > 0
    then if colisao[fresh] = 1  {houve colisão na janela proposta para
                                 contenção}
                then if c < tj and ntr < nt {propõe-se uma nova janela
                                 para contenção}
                                 then c ← c + 1;  r ← tj – c;  fresh ← c;
                                      jan[fresh] ← 0;
                                      colisao[fresh] ← 0;
                                      for k ← 1 to nt
                                        pilha[fresh,k] ← -1;
                                      end;
                                      espera ← 0;
                                      propor ← espera
                                 else fresh ← -1;
                                      propor ← 1
                     else  {não houve colisão na janela proposta}
                            if jan(fresh) = 2 and ntr < nt
                                 then {se houve uso com sucesso, mantenho a
                                       janela}
                                      espera ← 0;
                                      jan(fresh) ← 0
                                 else {se não houve uso e se ncr > r, elimino
                                       a janela}
                                      if  ncr > r or ntr = nt
                                        then jan(c) ← 0;  c ← c – 1; r ← tj – c;
                                             fresh ← 0;
                                             espera ← espera + 1; {espera
                                             espera quadros para criar uma
                                             nova janela de contenção}
                                 propor ← espera
    else propor ← propor – 1;
            if propor = 0  or ncr < r {propõe uma nova janela de contenção}
```

```
                    then if c < tj and ntr < nt
                                then c ← c + 1;  r ← tj – c;  fresh ← c;
                                     jan[fresh] ← 0;
                                     colisao[fresh] ← 0;
                                     for k ← 1 to nt
                                       pilha[fresh,k] ← -1;
                                     end;
                                else fresh ← -1;
                                     propor ← 1
end; {tq}
end. {begin}
```

Um exemplo do algoritmo de alocação dinâmica de janelas de contenção para a transmissão dos dez primeiros quadros é mostrado a seguir. Existem dez terminais (A,B,C,....H,I,J) e o quadro possui cinco janelas. O mecanismo de agendamento das janelas de reserva utilizado é o cíclico (peso 1).

R = janela de reserva; C = janela de contenção;

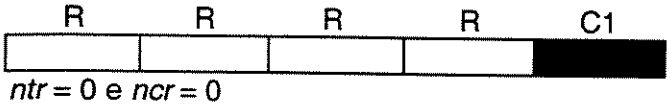
= janela livre; = janela com colisão;

A = o terminal A usa uma janela de reserva;

$A_{(2)}$ = janela com sucesso (o terminal A transmitiu com sucesso e reservou 2 janelas)

ntr = número de terminais com reserva agendada no quadro
ncr = número de células reservadas no quadro

1o. quadro:



ntr = 0 e *ncr* = 0

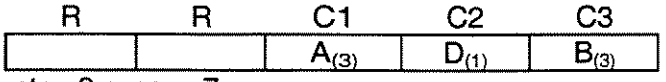
Os terminais A e C colidiram na nova janela de contenção 1.
Uma nova janela de contenção é proposta no próximo quadro, porque houve colisão na janela de contenção 1 e o *ntr* é menor que o número de terminais existentes (=10).

2o. quadro:

ntr = 0 e *ncr* = 0

Não ocorreu transmissão de célula na janela de contenção 1.
Os terminais D, H e J colidiram na nova janela de contenção 2.
Uma nova janela de contenção é proposta no próximo quadro, porque houve colisão na janela de contenção 2 e o *ntr* é menor que o número de terminais existentes.

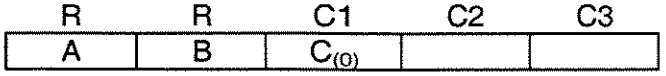
3o. quadro:



ntr = 3 e *ncr* = 7

O terminal A obteve sucesso na janela de contenção 1 e reservou três janelas.
O terminal D obteve sucesso na janela de contenção 2 e reservou uma janela.
O terminal B obteve sucesso na nova janela de contenção 3 e reservou três janelas.
Uma nova janela de contenção é proposta no próximo quadro, porque houve uso da nova janela de contenção 3 e o *ntr* é menor que o número de terminais existentes.
Como resolveu-se a contenção na janela 3, esta janela é eliminada do próximo quadro.

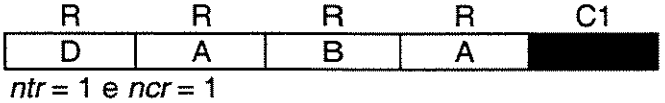
4o. quadro:



ntr = 3 e *ncr* = 5

Os terminais A e B são atendidos pelas duas janelas de reserva disponíveis.
O terminal C obteve sucesso na janela de contenção 1, mas não reservou janelas.
Não houve transmissão de célula nas janelas de contenção 2 e 3.
Como resolveu-se a contenção da janela 1, esta janela é eliminada do próximo quadro.
Como não houve uso da nova janela de contenção 3 e o *ncr* é maior do que o número de janelas de reserva do quadro (=3), a janela C3 é eliminada no próximo quadro e haverá um quadro sem a proposição de uma nova janela de contenção. Portanto, o algoritmo deverá propor uma nova janela de contenção no 6o. quadro.
No próximo quadro tem-se apenas uma janela de contenção: a C1 passa a ser a C2 do quadro anterior que se encontra em processo de contenção.

5o. quadro:



ntr = 1 e *ncr* = 1

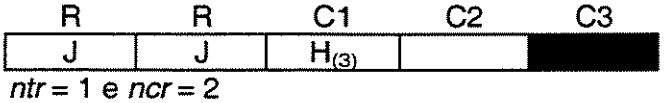
Os terminais A, B e D são atendidos pelas quatro janelas de reserva disponíveis.
Os terminais H e J colidiram na janela de contenção 1.
Como nesse quadro não houve proposição de nova janela de contenção, propõe-se uma nova janela de contenção no próximo quadro.

6o. quadro:

ntr = 1 e *ncr* = 2

O terminal B é atendido por uma janela de reserva.
O terminal J obteve sucesso na janela de contenção 1 e reservou duas janelas.
Os terminais D e I colidiram na nova janela de contenção 2.
Uma nova janela de contenção é proposta no próximo quadro, porque houve colisão na nova janela de contenção 2 e o *ntr* é menor que o número de terminais existentes.

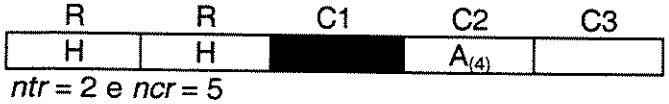
7o. quadro:



ntr = 1 e *ncr* = 2

O terminal J é atendido pelas duas janelas de reserva disponíveis.
O terminal H obteve sucesso na janela de contenção 1 e reservou três janelas.
Não ocorreu transmissão de célula na janela de contenção 2.
Os terminais A e E colidiram na nova janela de contenção 3.
Como resolveu-se a contenção na janela 1, esta janela é eliminada do próximo quadro.
Uma nova janela de contenção é proposta no próximo quadro, porque houve colisão na nova janela de contenção 3 e o *ntr* é menor que o número de terminais existentes.
No próximo quadro tem-se três janelas de contenção: a C1 passa a ser a C2 do quadro anterior, que ainda se encontra em processo de contenção, a C2 passa a ser a C3 do quadro anterior, que se encontra também em processo de contenção e a C3 é a nova janela de contenção proposta.

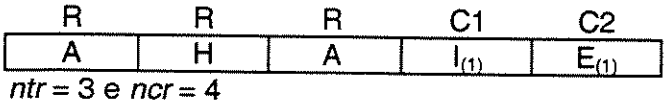
8o. quadro:



ntr = 2 e *ncr* = 5

O terminal H é atendido pelas duas janelas de reserva disponíveis.
Os terminais D e I colidiram na janela de contenção 1.
O terminal A obteve sucesso na janela de contenção 2 e reservou quatro janelas.
Não ocorreu transmissão de célula na nova janela de contenção 3.
Como não houve uso da nova janela de contenção 3 e o *ncr* é maior do que o número de janelas de reserva do quadro (=2), a janela C3 é eliminada no próximo quadro e haverá um quadro sem proposição de uma nova janela de contenção. Portanto, o algoritmo deverá propor uma nova janela de contenção no 10o. quadro.

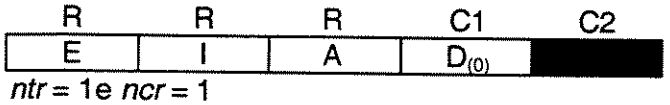
9o. quadro:



ntr = 3 e *ncr* = 4

Os terminais A e H são atendidos pelas três janelas de reserva disponíveis.
O terminal I obteve sucesso na janela de contenção 1 e reservou uma janela.
O terminal E obteve sucesso na janela de contenção 2 e reservou uma janela.
Como resolveu-se a contenção na janela 2, esta janela é eliminada do próximo quadro.
Uma nova janela de contenção é proposta no próximo quadro.

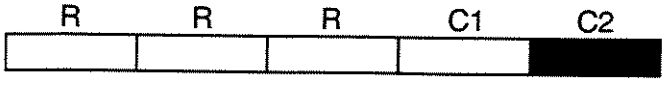
10o. quadro:



ntr = 1e *ncr* = 1

Os terminais A, E e I são atendidos pelas três janelas de reserva disponíveis.
O terminal D obteve sucesso na janela de contenção 1, mas não reservou janelas.
Os terminais J e C colidiram na nova janela de contenção 2.
Como resolveu-se a contenção na janela 1, esta janela é eliminada do próximo quadro.
Uma nova janela de contenção é proposta no próximo quadro, porque houve colisão na nova janela de contenção 2 e o *ntr* é menor que o número de terminais existentes.

11o. quadro:



II.4 – ADAPTAÇAO DO TAMANHO DO QUADRO

O algoritmo de alocação dinâmica de janelas de contenção e de reserva possui as seguintes alterações em relação ao descrito em II.3:

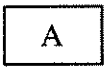
- o número de janelas do quadro é variável;
- o número de janelas de reserva (r) do quadro é igual ao número de terminais com reserva agendada (*ntr*) no quadro;
- se não houve uso da janela de contenção proposta no quadro corrente, ela é eliminada, via mecanismo de desestímulo, que passa a atuar se todos os terminais têm reserva agendada ou se existe pelo menos uma célula reservada (*ncr* > 0);
- se a janela proposta no quadro corrente é eliminada, a variável *espera* (que controla o número de quadros sem proposição de janela de contenção) é incrementada unitariamente;
- não há limite para o número de janelas de contenção no quadro;
- o limite do número de janelas de reserva no quadro é igual ao número de terminais existentes.

No início de cada quadro, o algoritmo fornece o número de janelas de reserva e o número de janelas de contenção do quadro corrente. As janelas de reserva são alocadas no início do quadro e as janelas de contenção, no final do quadro.

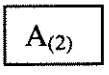
Um exemplo do algoritmo de alocação dinâmica de janelas para a transmissão dos dez primeiros quadros é mostrado a seguir. Existem dez terminais (A,B,C,....H,I,J) e o quadro é de tamanho variável. O mecanismo de agendamento das janelas de reserva utilizado é o cíclico (peso 1) com *piggybacking*.

R = janela de reserva; C = janela de contenção;

 = janela livre;  = janela com colisão;



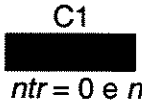
= o terminal A usa uma janela de reserva;



= janela com sucesso (o terminal A transmitiu com sucesso e reservou 2 janelas)

ntr = número de terminais com reserva agendada no quadro
ncr = número de células reservadas no quadro

1o. quadro:



ntr = 0 e *ncr* = 0

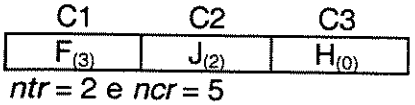
Os terminais A, D e F colidiram na nova janela de contenção 1.
Uma nova janela de contenção é proposta no próximo quadro, porque houve colisão na janela de contenção 1 e o *ntr* é menor que o número de terminais existentes (=10).
Não há janela de reserva no próximo quadro (r=*ntr*=0).

2o. quadro:

ntr = 0 e *ncr* = 0

Não ocorreu transmissão de célula na janela de contenção 1.
Os terminais B e J colidiram na nova janela de contenção 2.
Uma nova janela de contenção é proposta no próximo quadro, porque houve colisão na janela de contenção 2 e o *ntr* é menor que o número de terminais existentes.
Não há janela de reserva no próximo quadro.

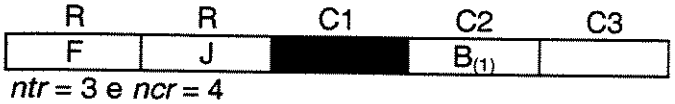
3o. quadro:



ntr = 2 e *ncr* = 5

O terminal F obteve sucesso na janela de contenção 1 e reservou três janelas.
O terminal J obteve sucesso na janela de contenção 2 e reservou duas janelas.
O terminal H obteve sucesso na janela de contenção 3 mas não reservou janelas.
Como resolveu-se a contenção na janela 3, esta janela é eliminada no próximo quadro.
Uma nova janela de contenção é proposta para o próximo quadro, porque houve uso da nova janela de contenção 3 e o *ntr* é menor que o número de terminais existentes.
No próximo quadro há duas janelas de reserva (r=*ntr*=2).

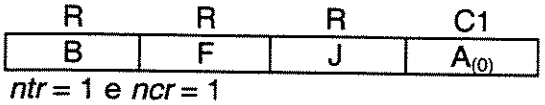
4o. quadro:



ntr = 3 e *ncr* = 4

Os terminais F e J são atendidos pelas duas janelas de reserva disponíveis.
Os terminais A e D colidiram na janela de contenção 1.
O terminal B obteve sucesso na janela de contenção 2 e reservou uma janela.
Como não houve uso da nova janela de contenção 3 e o *ncr* é maior do que zero, a janela C3 é eliminada no próximo quadro e haverá um quadro sem a proposição de uma nova janela de contenção. Portanto, o algoritmo deverá propor uma nova janela de contenção no 6o. quadro.
Como resolveu-se a contenção da janela 2, esta janela é eliminada no próximo quadro.
No próximo quadro tem-se apenas uma janela de contenção: a C1 do quadro anterior que se encontra em processo de contenção.
No próximo quadro há três janelas de reserva (r=*ntr*=3).

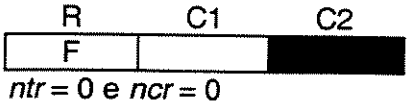
5o. quadro:



ntr = 1 e *ncr* = 1

Os terminais B, F e J são atendidos pelas três janelas de reserva disponíveis.
O terminal A obteve sucesso na janela de contenção 1, mas não reservou janelas.
Como nesse quadro não houve proposição de nova janela de contenção, propõe-se uma nova janela de contenção no próximo quadro.
No próximo quadro há uma janela de reserva (r=*ntr*=1).

6o. quadro:

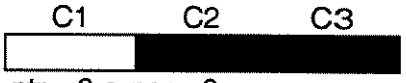


ntr = 0 e *ncr* = 0

O terminal F é atendido pela janela de reserva disponível.

Não houve transmissão de célula na janela de contenção 1.
Os terminais F e C colidiram na nova janela de contenção 2.
Uma nova janela de contenção é proposta no próximo quadro, porque houve colisão na nova janela de contenção 2.
Não há janela de reserva no próximo quadro.

7o. quadro:



ntr = 0 e *ncr* = 0

Não ocorreu transmissão de célula na janela de contenção 1.
Os terminais F e C colidiram na janela de contenção 2.
Os terminais A e I colidiram na nova janela de contenção 3.
Uma nova janela de contenção é proposta no próximo quadro, porque houve colisão na nova janela de contenção 3.
Não há janela de reserva no próximo quadro.

8o. quadro:

C1	C2	C3	C4
$D_{(5)}$	$C_{(2)}$	$I_{(4)}$	

ntr = 3 e *ncr* = 11

O terminal D obteve sucesso na janela de contenção 1 e reservou cinco janelas.
O terminal C obteve sucesso na janela de contenção 2 e reservou duas janelas.
O terminal I obteve sucesso na janela de contenção 3 e reservou quatro janelas.
Como resolveu-se a contenção na janela 1, esta janela é eliminada no próximo quadro.
Como não houve uso da nova janela de contenção 4 e o *ncr* é maior do que zero, a janela C4 é eliminada no próximo quadro e haverá um quadro sem a proposição de uma nova janela de contenção. Portanto, o algoritmo deverá propor uma nova janela de contenção no 10o. quadro.
No próximo quadro tem-se duas janelas de contenção: a C1 passa a ser a C2 do quadro anterior, que ainda se encontra em processo de contenção e a C2 passa a ser a C3 do quadro anterior, que se encontra também em processo de contenção.
No próximo quadro há três janelas de reserva (r=*ntr*=3).

9o. quadro:

R	R	R	C1	C2
I	C	D	$F_{(1)}$	$A_{(1)}$

ntr = 5 e *ncr* = 10

Os terminais C, D e I são atendidos pelas três janelas de reserva disponíveis.
O terminal F obteve sucesso na janela de contenção 1 e reservou uma janela.
O terminal A obteve sucesso na janela de contenção 2 e reservou uma janela.
Como resolveu-se a contenção nas janelas 1 e 2, estas janelas são eliminadas no próximo quadro.
Como nesse quadro não houve proposição de nova janela de contenção, propõe-se uma nova janela de contenção no próximo quadro.
No próximo quadro há cinco janelas de reserva (r=*ntr*=5).

10o. quadro:

R	R	R	R	R	C1
F	I	A	C	D	

ntr = 2 e *ncr* = 5

O terminais A, C, D, F e I são atendidos pelas cinco janelas de reserva disponíveis.
Como não houve uso da nova janela de contenção 1 e o *ncr* é maior do que zero, a janela C1 é eliminada no próximo quadro e haverá dois quadros (mecanismo de desestímulo) sem a proposição de uma nova janela de contenção. Portanto, o algoritmo deverá propor uma nova janela de contenção no 13o. quadro.
No próximo quadro há duas janelas de reserva (r=*ntr*=2).

11o. quadro:

R	R
I	D

II.5 – MINI-JANELAS

A transmissão de mensagens em janelas de contenção é um mecanismo eficiente quando o número de terminais envolvidos é muito grande e cada um com uma carga de tráfego muito baixa. Nas demais situações é quase sempre preferível o uso de janelas de contenção apenas para solicitação de reserva para posterior transmissão. As janelas de contenção devem conter a identificação do terminal e parâmetros que caracterizam o pedido. No contexto de ATM sem fio é típico o uso de janelas de contenção com um quarto do tamanho das janelas que transmitem uma célula ATM [Pe95], [AGSF00].

II.6 – TRÁFEGO EM TEMPO REAL

O algoritmo de alocação dinâmica de janelas de contenção e de reserva, para atender os tráfegos em tempo real e não em tempo real, possui as seguintes alterações em relação ao descrito em II.4:

- na resolução de contenção, é utilizado o algoritmo *binary tree* para os terminais com tráfego em tempo real e o *ternary tree* para os terminais com tráfego não em tempo real;
- é proposta uma nova janela de contenção em todos os quadros salvo no caso de todos os terminais possuírem reserva agendada, ou que o mecanismo de desestímulo esteja atuante;
- para terminais com tráfego CBR, atualiza-se a variável *reserva* (+1) no tempo previsto para a chegada da célula no *buffer* (*timestamp* + 1/PCR);
- quando a janela é disponibilizada para tráfego CBR e o terminal não faz uso da janela o algoritmo ignora este fato e agenda uma nova janela como se a janela disponibilizada tivesse sido usada; quando a janela é usada agenda-se uma nova janela para *timestamp* + 1/PCR;
- o mecanismo de agendamento das janelas de reserva utilizado é o cíclico (peso 1) com prioridade na ordem CBR, rtVBR, nrtVBR:
  - se existe pelo menos um terminal CBR com reserva agendada, o número de janelas de reserva do quadro é igual ao número de terminais CBR com reserva agendada;
    - se nenhum terminal CBR possui reserva agendada, mas existe pelo menos um terminal rtVBR com reserva agendada, o número de janelas de reserva do quadro é igual ao número de terminais rtVBR com reserva agendada;
      - se nenhum terminal rtVBR possui reserva agendada, mas existe pelo menos um terminal nrtVBR com reserva agendada, o número de janelas de reserva do quadro é igual ao número de terminais nrtVBR com reserva agendada;

- se nenhum terminal nrtVBR possui reserva agendada, o número de janelas de reserva do quadro é zero.

Pseudocódigo da definição do número de janelas de reserva do quadro

r = número de janelas de reserva no quadro;
ntcbr = número de terminais com tráfego CBR;
ntrt = número de terminais com tráfego rtVBR;
ntnrt = número de terminais com tráfego VBRnrt;
reserva = vetor reserva dos terminais (nt);
tem = indica que existe tráfego CBR (=0), rtVBR (=1) ou nrtVBR (=2);
i = variável auxiliar.

```
begin
        input ntcbr; {número de terminais com tráfego CBR}
        input ntrt; {número de terminais com tráfego rtVBR}
        input ntnrt; {número de terminais com tráfego nrtVBR}
        input tem; {indica o tipo de tráfego existente}
        r ← 0;

        for i ← 1 to ntcbr {verifica se existe reserva agendada para terminais
                            com tráfego CBR}
                if reserva(i) > 0
                        then r ← r + 1;
        end;
        if r > 0
                then tem ← 0
                else for i ← (ntcbr + 1) to (ntcbr + ntrt)
                        {verifica se existe reserva agendada para terminais
                        com tráfego rtVBR}
                        if reserva(i) > 0
                                then r ← r + 1;
                    end;
                    if r > 0
                      then tem ← 1
                      else for i ← (ntcbr + ntrt + 1) to (ntcbr + ntrt+ ntnrt)
                              {verifica se existe reserva agendada para terminais
                               com tráfego nrtVBR}
                               if reserva(i) > 0
                                       then r ← r + 1;
                           end;
                           if r > 0
                             then tem ← 2;
end;
```

CAPÍTULO III – MODELOS DE SIMULAÇÃO

III.1 – TRANSMISSÃO POR CONTENÇÃO

A transmissão por contenção é modelada pelo diagrama de blocos (Arena [ARENA5.0]) mostrado na Figura 3.1, no qual estão representados cinco terminais.

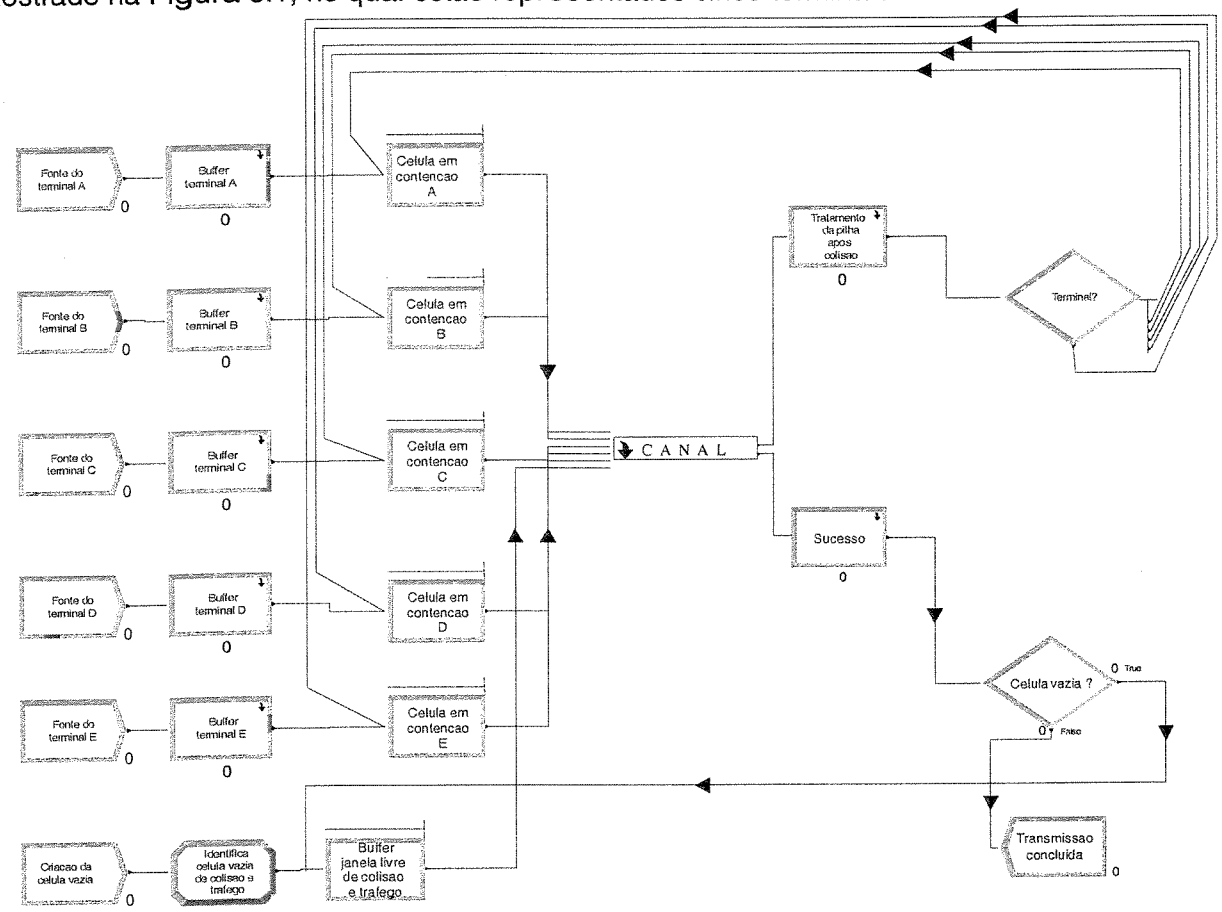


Figura 3.1 - Modelo de transmissão por contenção.

O terminal é modelado por uma fonte (*create*), por um buffer (*hold = Buffer terminal*) que armazena as células em espera para transmissão e por um buffer (*hold = Célula em contenção*) que controla a célula que participa do processo de contenção. O primeiro buffer, *Buffer terminal*, libera a célula quando todas as células do processo de contenção anterior foram transmitidas com sucesso e existe, pelo menos, uma célula no buffer desse terminal para participar de uma nova tentativa de transmissão. O segundo buffer, *Célula em contenção*, libera a célula quando é início de quadro (a primeira janela do quadro é de contenção) e a variável pilha do terminal é zero. No caso de colisão, o segundo buffer recebe de volta a célula que colidiu.

O modelo de janela livre (de colisão e de tráfego) possui uma fonte (*create*) que gera uma única célula no instante zero e um buffer (*hold = Buffer janela livre de colisão e tráfego*) que libera a célula quando é início de quadro e a variável pilha é zero. A célula criada pela fonte é identificada como uma célula vazia que ocupa uma janela livre (de colisão ou de tráfego). Se o valor da variável pilha para cada terminal é superior a zero, tem-se a ocorrência de uma janela livre (de colisão). Se o processo de colisão foi resolvido e não existem células nos buffers dos

terminais, para serem liberadas, tem-se a ocorrência de uma janela livre (de tráfego). Em ambos os casos, livre (de colisão) e livre (de tráfego), a célula vazia é liberada do buffer para ocupar a janela de contenção (primeira do quadro) e volta sempre ao buffer.

O canal (vide Figura 3.2) é modelado pelo recurso *process* do Arena (= *Utilização da janela*) e por uma lógica que controla a utilização desse recurso, identificando a ocorrência ou não de colisão no canal (primeira janela do quadro). A primeira célula que participa do processo de contenção ocupa o recurso *Utilização da janela* durante o tempo de janela. Se durante esse tempo, outra célula chega para ocupar o recurso, a colisão é identificada e essa célula é submetida a um atraso igual ao tempo de janela. Se não chega nenhuma célula durante o tempo de janela (tempo de utilização do recurso), não há colisão, e, portanto, a célula é transmitida com sucesso. Em ambos os casos, colisão e não colisão, as células que participam do processo de contenção são submetidas a um atraso que corresponde ao complemento do tempo de quadro (tempo de três janelas, no exemplo).

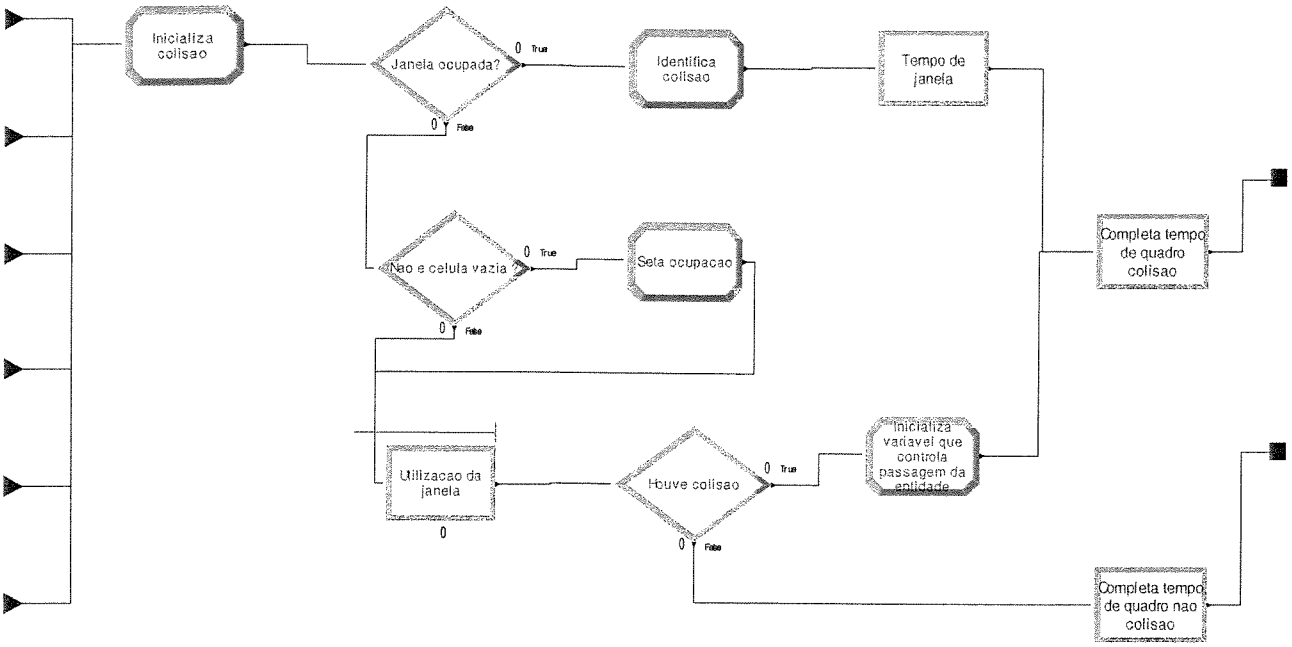


Figura 3.2 - Modelo de canal.

Quando há colisão (vide Figura 3.3), os terminais que participam do processo de contenção, mas não participam da colisão, têm a sua variável pilha incrementada de dois (*ternary tree*) ou incrementada de um (*binary tree*). Os terminais que têm suas células colididas na primeira janela do quadro escolhem o quadro da sua próxima transmissão (0 = primeiro quadro; 1 = segundo quadro; 2 = terceiro quadro no *ternary* ou 0, 1 no *binary*). Após o incremento e a escolha do quadro de transmissão, verifica-se a existência de janela livre (de colisão). Se o valor da variável pilha para cada terminal é superior a zero, tem-se a ocorrência de uma janela livre (de colisão), e, portanto, uma célula vazia é liberada do buffer *Buffer janela livre de colisão e tráfego* para ser transmitida pelo canal. As células que participaram do processo de colisão são identificadas e encaminhadas aos terminais de origem, para, dependendo do valor da variável pilha, participarem de uma nova tentativa de transmissão (pilha = zero).

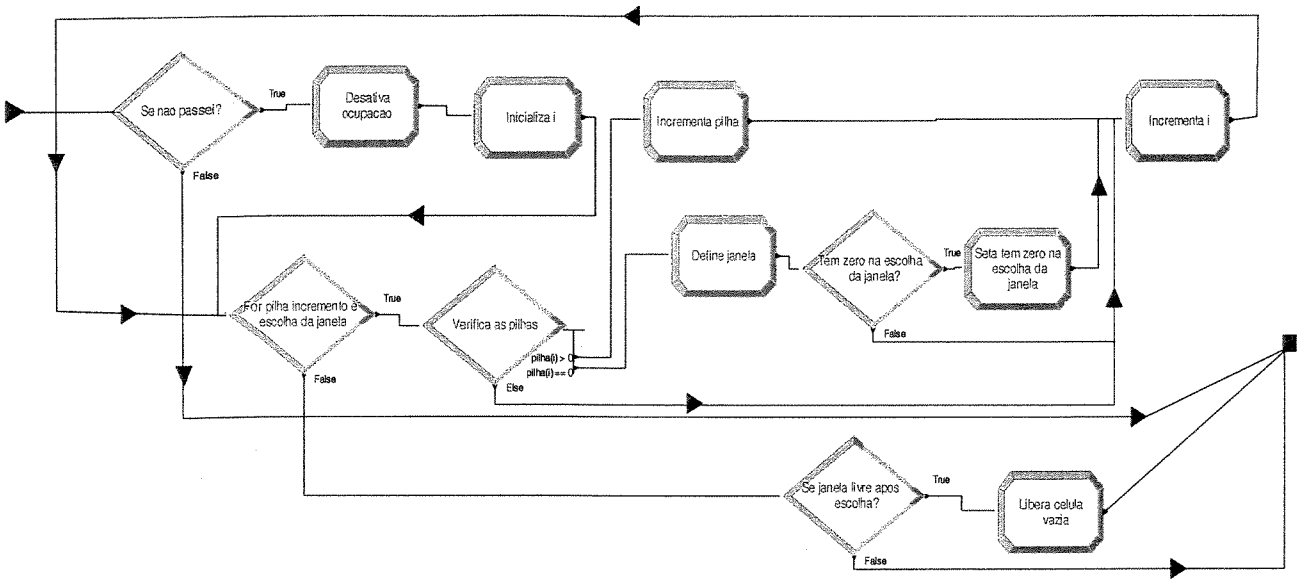


Figura 3.3 – Tratamento da pilha após colisão.

Quando não há colisão, é computado o tempo de transferência da célula; é feito um tratamento da pilha e verifica-se a existência de janela livre (de colisão) e de janela livre (de tráfego) (vide Figura 3.4).

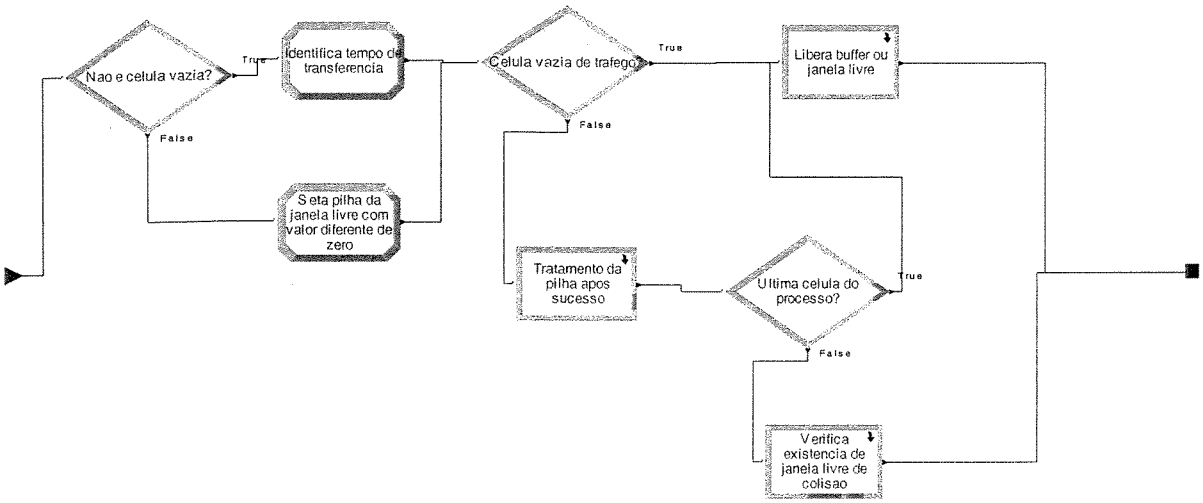


Figura 3.4 - Sucesso.

Se a célula que obteve sucesso não é uma célula vazia, que identifica uma janela livre (de tráfego), o valor da variável pilha de cada terminal, que é superior ou igual a zero, é decrementado unitariamente (vide Figura 3.5).

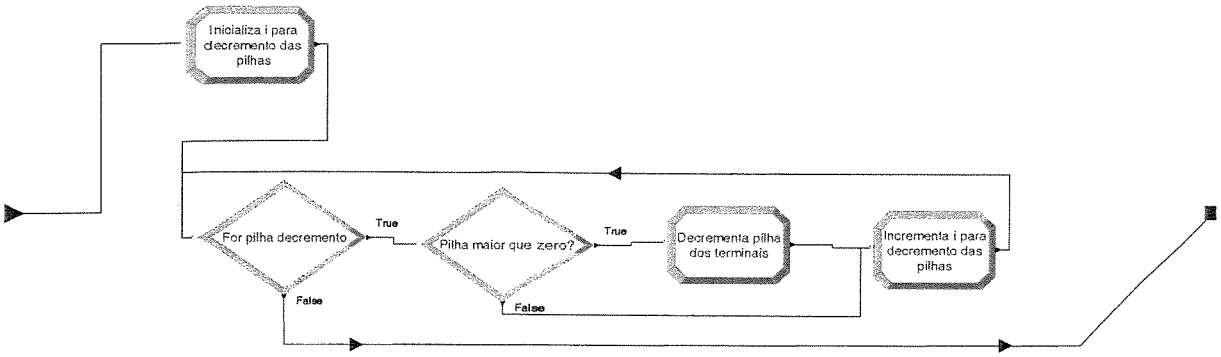


Figura 3.5 – Tratamento da pilha após sucesso.

Após o decremento da variável pilha, pode ocorrer uma janela livre (de colisão), quando o valor da variável pilha de cada terminal, que participa do processo de contenção, é superior a zero. Se isso ocorre, uma célula vazia é liberada do buffer *Buffer janela livre de colisão e tráfego* para ser transmitida pelo canal (vide Figura 3.6).

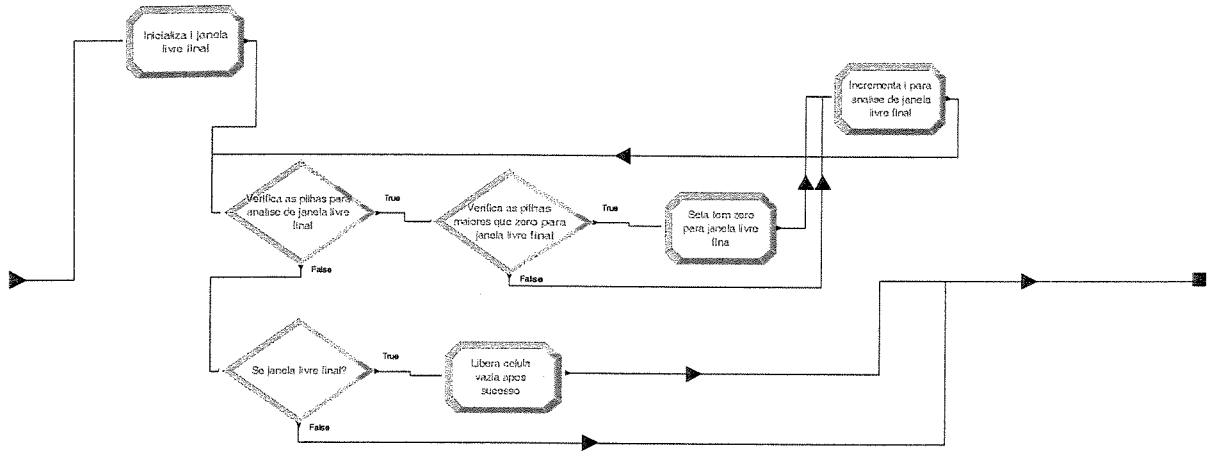


Figura 3.6 – Verifica existência de janela livre (de colisão).

Quando o processo de contenção é finalizado, ou seja, quando todas as células que participaram do processo de contenção foram transmitidas com sucesso, o *processo Libera buffer ou janela livre* (vide Figura 3.7) verifica a existência de células nos buffers dos terminais. Caso haja pelo menos uma célula, em um dos terminais, essa célula é enviada ao buffer de contenção do terminal *Célula em contenção*. Caso contrário, uma célula vazia é liberada do buffer *Buffer janela livre de colisão e tráfego* para ser transmitida pelo canal para identificar uma janela livre (de tráfego). Ao término da transmissão do quadro, ela é encaminhada de volta ao buffer *Buffer janela livre de colisão e tráfego*.

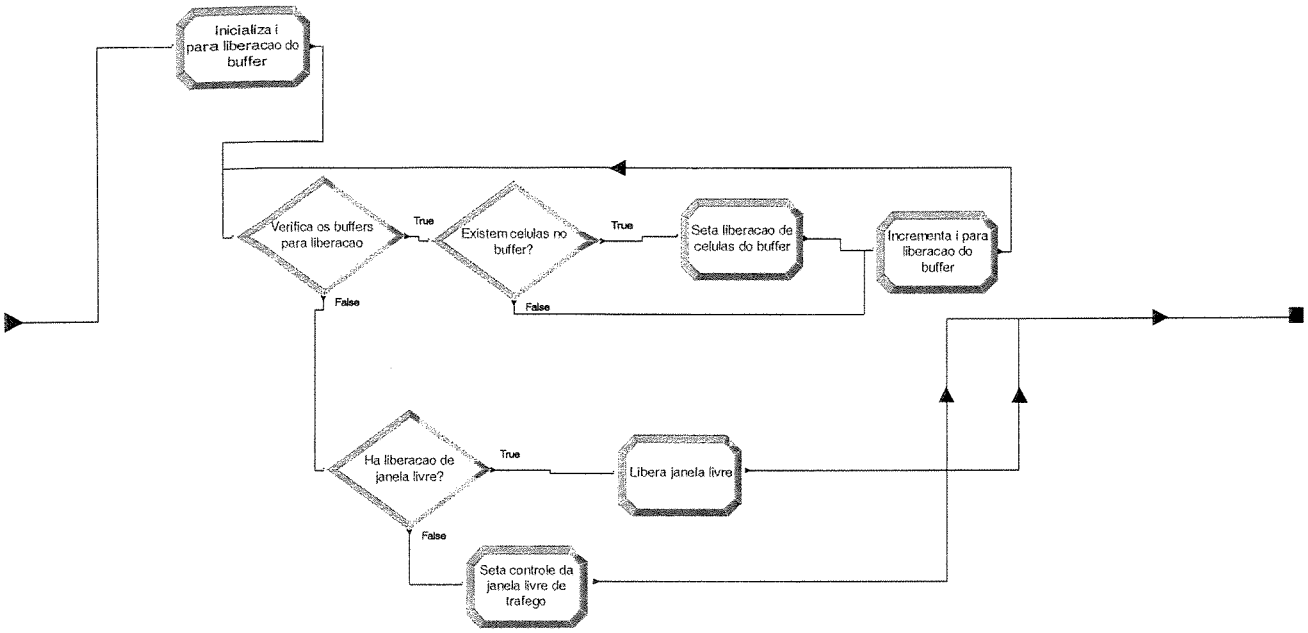


Figura 3.7 – Libera buffer ou janela livre.

Para validar o modelo de simulação, as Figuras 3.8 e 3.9 mostram a ocupação do canal em função da carga com os algoritmos *binary* e *ternary trees,* respectivamente, usando simulação de evento discreto com 125.000 quadros. Uma carga de 100% significa a ocupação de todas as janelas disponíveis do quadro. No modelo de transmissão por contenção, o quadro possui quatro janelas sendo a primeira de contenção. Desta forma, 100% é uma célula por janela de contenção, ou seja, uma célula por quadro. A curva identificada *simulado* apresenta os resultados obtidos na simulação e a outra curva apresenta os resultados obtidos na literatura [Kl76], onde para um número M de terminais, a fórmula para cálculo da vazão é

$$S = G\left(1 - \frac{G}{M}\right)^{M-1} \tag{1}$$

onde S = vazão e G = taxa de ocupação.

Em ambas as figuras, verifica-se uma região não estacionária de simulação, onde o número de ocupação já saturou, ou seja, a ocupação não cresce mais (carga próxima de 40% da capacidade do canal). A ocupação máxima no *binary tree* é de 85%, o que comprova o percentual de 15% de janelas livres. A ocupação máxima no *ternary tree* é de aproximadamente 70%, o que comprova também o percentual de 30% de janelas livres.
Pode-se observar que o *ternary* é levemente superior ao *binary*, pois para uma mesma carga há uma menor ocupação do canal.

As Figuras 3.10 e 3.11 mostram o tempo de contenção e transmissão e o tempo de espera no buffer, respectivamente, com a utilização dos algoritmos *binary e ternary trees*. Observa-se na Figura 3.10 um crescimento exponencial do tempo de contenção e transmissão com o aumento da carga. Na Figura 3.11, observa-se um crescimento muito acentuado do tempo de espera no buffer.

A Figura 3.12 mostra o número médio de células no buffer com a utilização dos algoritmos *binary e ternary trees*. Observa-se que a partir da carga de 35%, o número de células no buffer tem um crescimento muito acentuado.

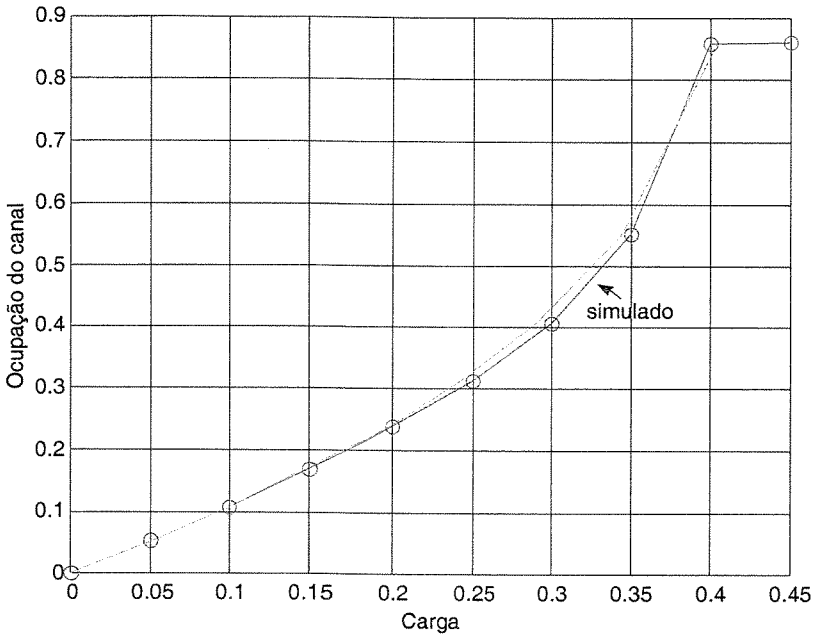


Figura 3.8 – Ocupação versus carga na transmissão por contenção utilizando *binary tree* com chegadas poissonianas. Simulação e modelo analítico.

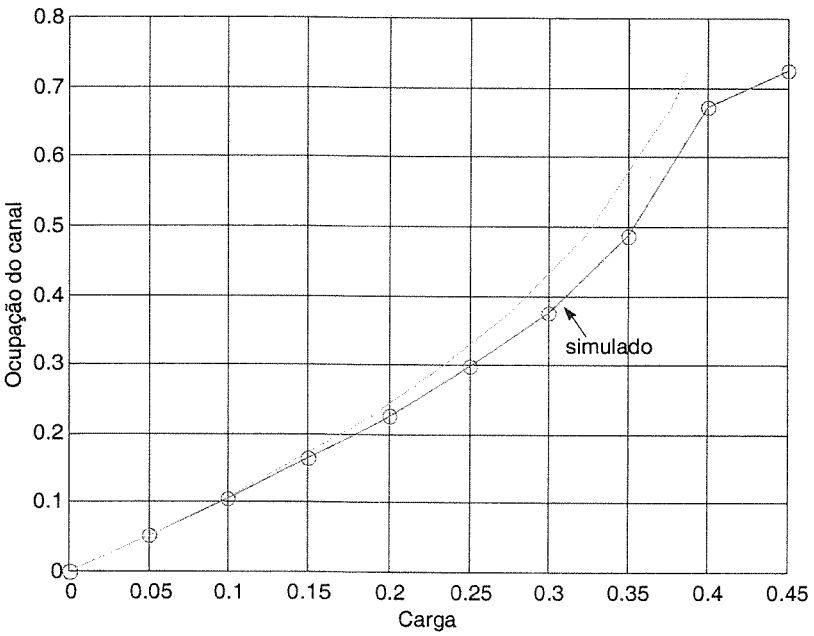


Figura 3.9 – Ocupação versus carga na transmissão por contenção utilizando *ternary Tree* com chegadas poissonianas. Simulação e modelo analítico.

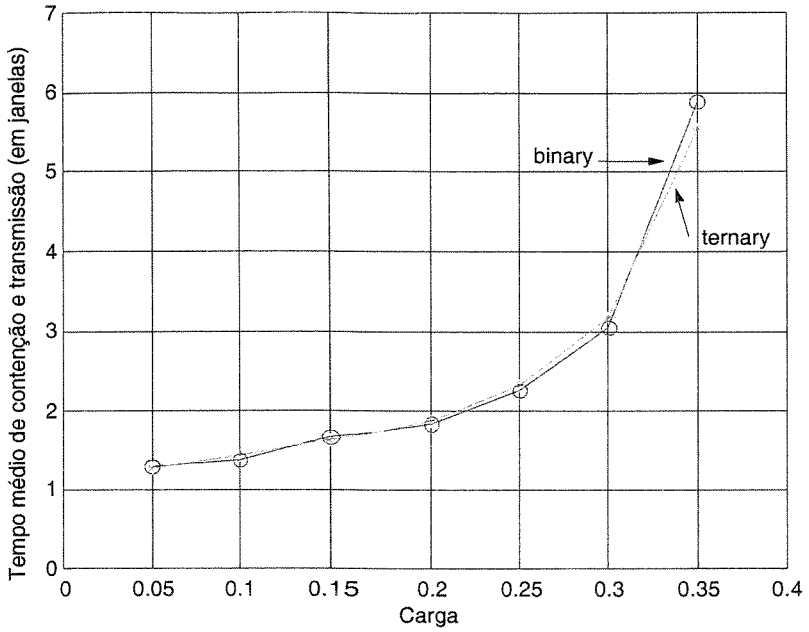


Figura 3.10 – Tempo de contenção e transmissão usando transmissão por contenção com chegadas poissonianas.

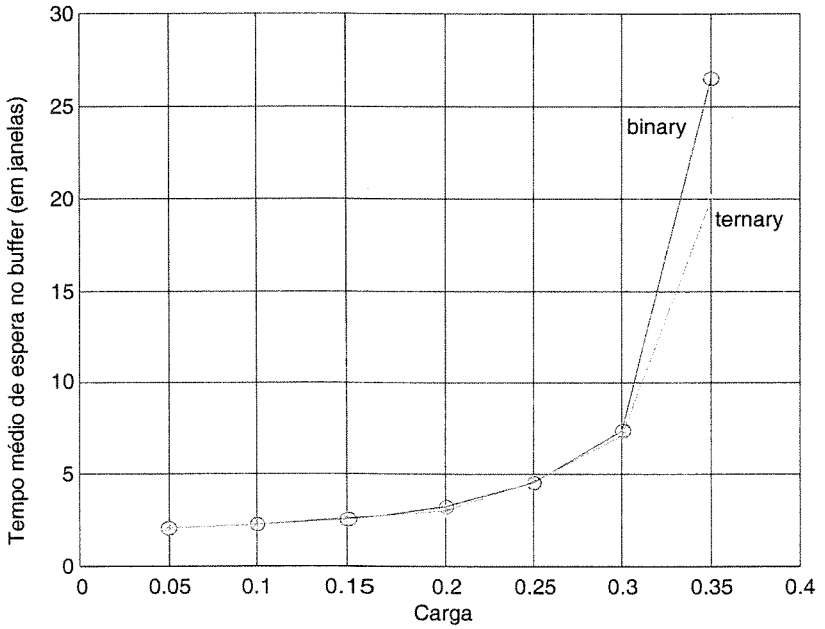


Figura 3.11 – Tempo de espera no buffer usando transmissão por contenção com chegadas poissonianas.

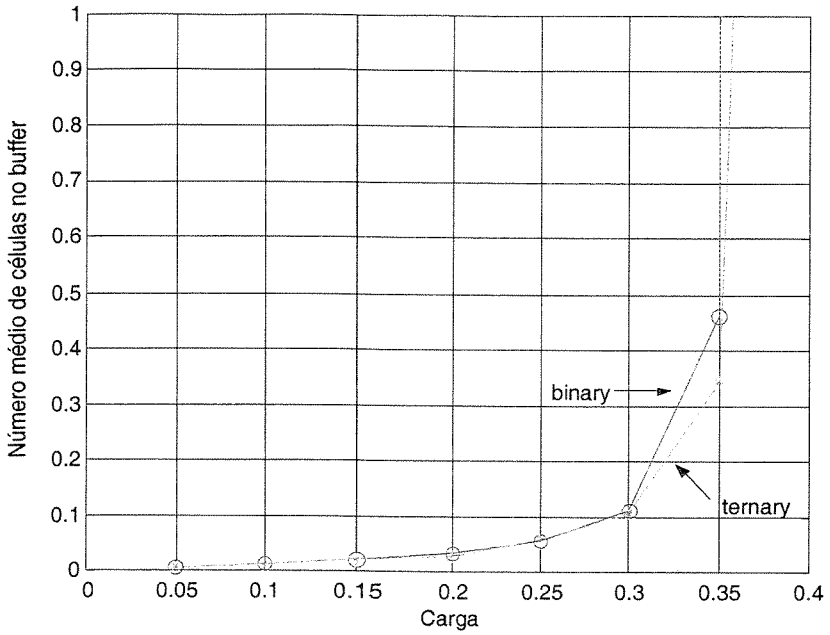


Figura 3.12 – Número médio de células no buffer usando transmissão por contenção com chegadas poissonianas.

III.2 – AGENDAMENTO CÍCLICO

O agendamento cíclico é modelado pelo diagrama mostrado na Figura 3.13, no qual estão representados cinco terminais (A, B, C, D e E). Os terminais utilizam a janela de contenção (primeira janela do quadro) para requisição das janelas de reserva. O algoritmo usado para resolução da contenção é o *ternary tree*.

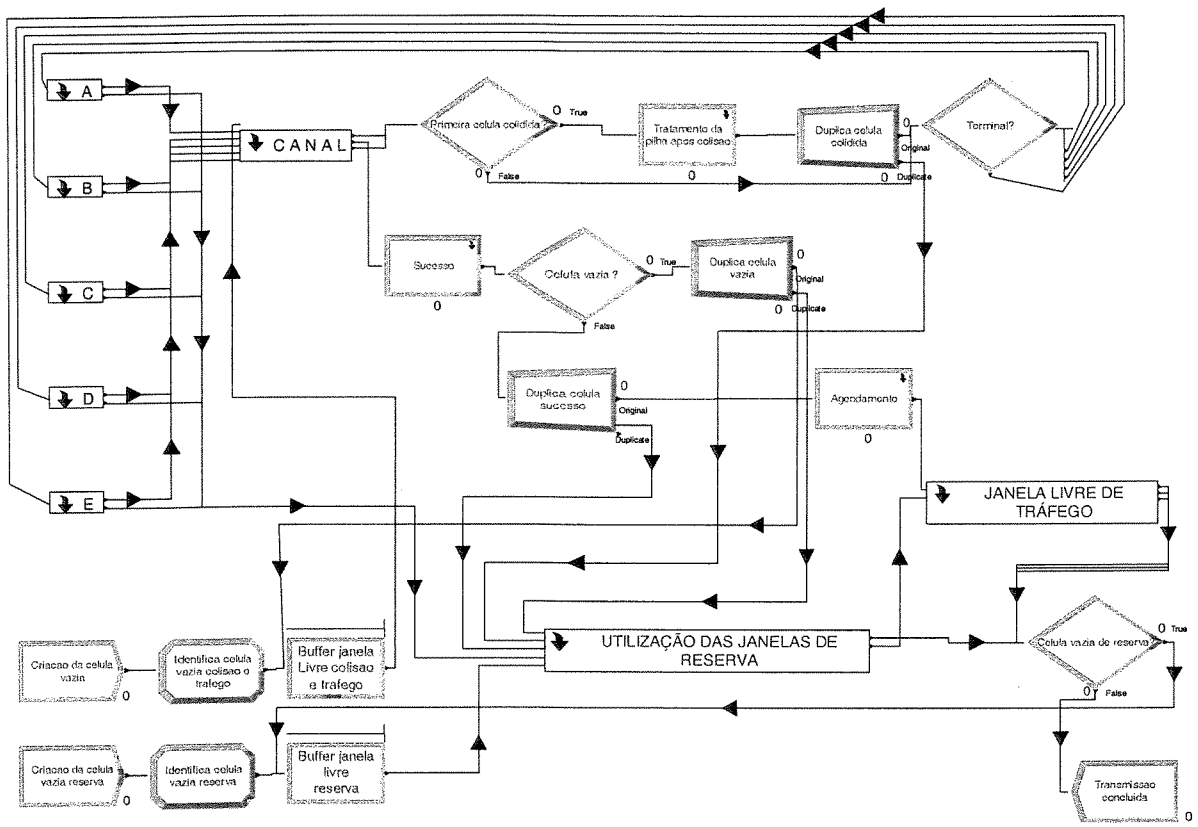


Figura 3.13 - Modelo de agendamento cíclico.

O terminal (vide Figura 3.14) é modelado por uma fonte (*create*), por um buffer (*hold = Buffer A*) que armazena as células em espera (de contenção e de reserva) e por um buffer (*hold = Célula em contenção A*) que controla a célula que participa do processo de contenção. O primeiro buffer, *Buffer A*, libera dois tipos de célula: de contenção e de reserva. A célula de contenção é liberada quando não existe reserva agendada de células para esse terminal e todas as células, do processo de contenção anterior, foram transmitidas com sucesso. A célula de reserva é liberada pelo processo *Utilização das janelas de reserva* (vide Figuras 3.19 e 3.20). Se a célula liberada do primeiro buffer é uma célula de reserva, ela é encaminhada ao processo de *Utilização das janelas de reserva* e se é uma célula de contenção, ela é encaminhada ao *Canal* para participação do processo de contenção. O segundo buffer libera a célula quando é início de quadro (a primeira janela do quadro é de contenção) e a variável pilha do terminal é zero. Quando a célula é liberada do segundo buffer, registra-se, na variável agenda, o número de células (de reserva) existentes no primeiro buffer para posterior agendamento. No caso de colisão no canal (na primeira janela do quadro), o segundo buffer recebe de volta a célula que colidiu.

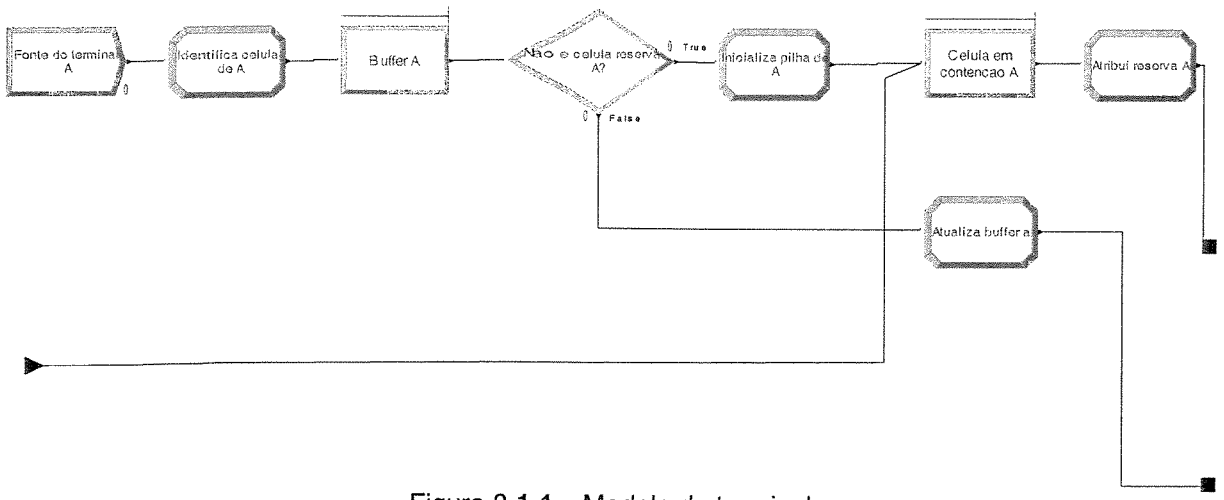


Figura 3.14 - Modelo de terminal.

O modelo de janela livre (de colisão e de tráfego) foi descrito em III.1 (vide Figura 3.1).

O modelo de janela livre (de reserva) possui uma fonte (*create*), que gera uma única célula no instante zero, e um buffer (*hold = Buffer janela livre reserva*), que libera a célula quando a variável pilha é zero. A célula criada pela fonte é identificada como uma célula vazia, que ocupa uma janela livre (de reserva). Se não existe reserva de células para nenhum dos terminais, a célula vazia é liberada do buffer para ocupar uma janela de reserva. A célula vazia volta ao buffer, após o término do tempo de janela ocupada por ela.

O canal (vide Figura 3.15) é modelado por um recurso (*process = Utilização da janela de contenção*) e por uma lógica que controla a utilização desse recurso, identificando a ocorrência, ou não, de colisão no canal (primeira janela do quadro). A primeira célula que participa do processo de contenção ocupa o recurso *Utilização da janela de contenção* durante o tempo de janela. Se durante esse tempo, uma outra célula tenta ocupar o recurso, a colisão é identificada e essa célula é submetida a um atraso igual ao tempo de janela. Se não chega nenhuma célula durante o tempo de janela (tempo de utilização do recurso), não há colisão, e, portanto, a célula é transmitida com sucesso.

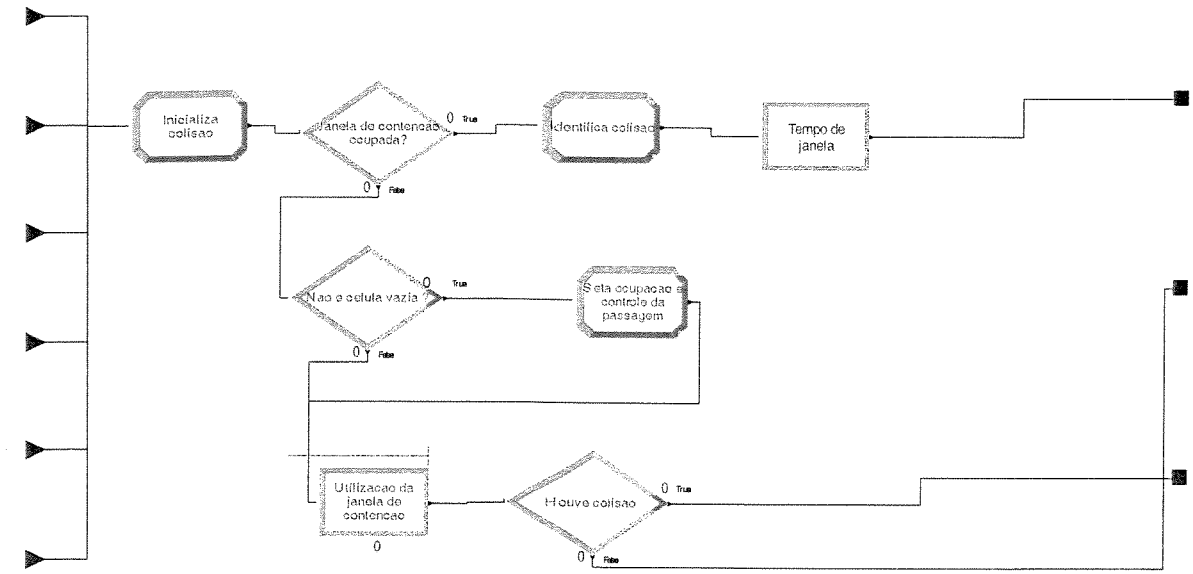


Figura 3.15 - Modelo de canal.

Quando há colisão, os terminais que participam do processo de contenção, mas não participam da colisão, têm a sua variável pilha incrementada de dois (vide Figura 3.3). Os terminais que têm suas células colididas na primeira janela do quadro escolhem o quadro da sua próxima transmissão (0 = primeiro quadro; 1 = segundo quadro; 2 = terceiro quadro). Se o valor da pilha para cada terminal é superior a zero, a próxima janela de contenção é livre (de colisão), e, portanto, uma célula vazia é liberada do *hold Buffer janela livre de colisão e tráfego* para ocupar a janela de contenção (primeira) no próximo quadro.

Na ocorrência de colisão, após o tratamento da pilha, uma das células colididas é duplicada para dar início ao processo *Utilização das janelas de reserva* (vide Figura 3.16). As células que participaram do processo de colisão são identificadas e encaminhadas aos terminais de origem (buffer de contenção), para, dependendo do valor da pilha, participarem de uma nova tentativa de transmissão, no próximo quadro.

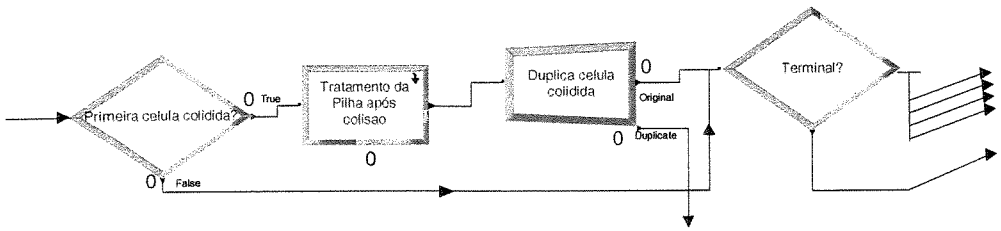


Figura 3.16 – Fluxo após colisão.

Quando não há colisão, é computado o tempo de transferência (contenção e transmissão) da célula; é feito um decremento da pilha e verifica-se a existência de janela livre (de colisão) (vide Figura 3.17).

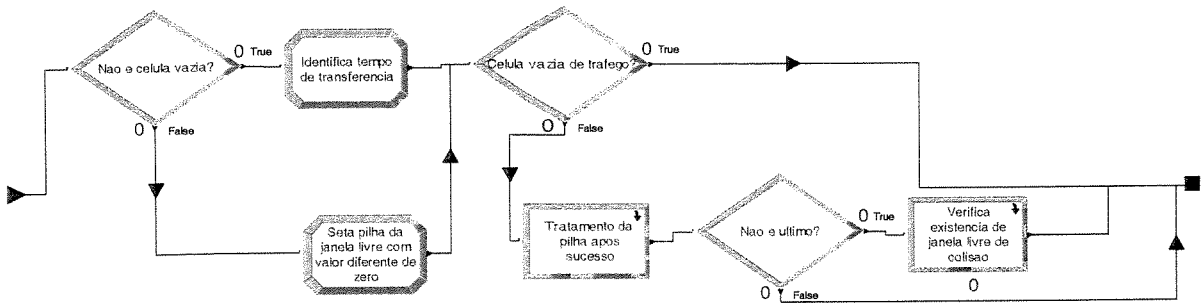


Figura 3.17 – Sucesso.

Na ocorrência de sucesso, após o tratamento da pilha e da verificação de existência de janela livre (de colisão), a célula é duplicada. Quando a célula duplicada é de sucesso, a original é encaminhada ao *Agendamento* e a duplicata inicia o processo de *Utilização das janelas de reserva*. Quando a célula duplicada é vazia (que identifica janela livre), a original volta ao buffer *Buffer janela livre colisão e tráfego* e a duplicata inicia o processo de *Utilização das janelas de reserva*.

No *Agendamento* (vide Figura 3.18), a célula é submetida a um atraso que corresponde ao complemento do tempo de quadro (tempo de três janelas). Após esse atraso, é feito o

agendamento, atualizando-se a variável reserva com o conteúdo da variável agenda. Caso o conteúdo da variável agenda seja zero, ou seja, não existe nenhuma reserva para o terminal, a variável reserva se mantém em zero.
Após o processo de *Agendamento*, a célula é encaminhada ao processo *Janela livre de tráfego* (vide Figura 3.22).

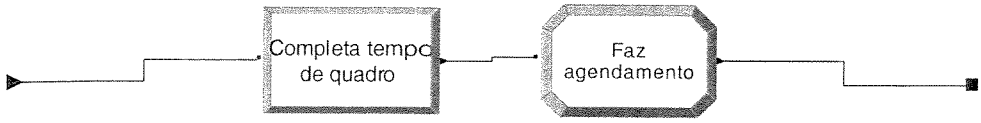


Figura 3.18 – Agendamento.

O processo de *Utilização das janelas de reserva* é mostrado na Figura 3.19. O quadro possui quatro janelas sendo a primeira de contenção e as demais de reserva.

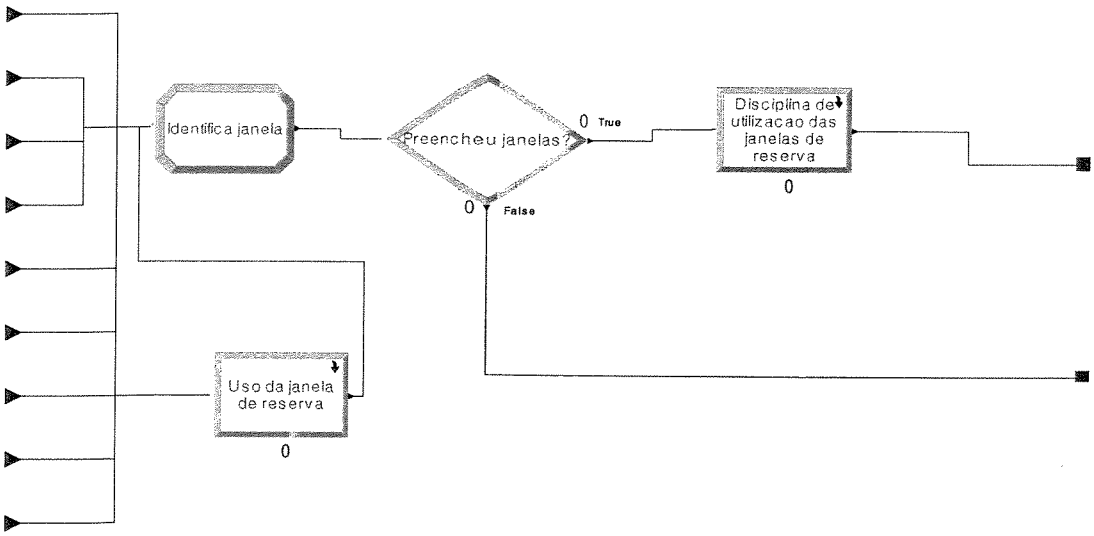


Figura 3.19 – Utilização das janelas de reserva.

A disciplina (agendamento cíclico) de utilização das janelas de reserva (vide Figura 3.20) consiste em atender uma célula de cada terminal por ciclo. A variável reserva é percorrida e se existir reserva para um dos terminais, uma célula é liberada pelo primeiro buffer desse terminal para ocupar a janela de reserva (vide Figura 3.21). Caso não exista nenhuma reserva, uma célula vazia é liberada pelo buffer *Buffer janela livre de reserva* para ocupar a janela de reserva (vide Figura 3.21). Decorrido o tempo de janela, o processo é repetido até esgotar-se o número total de janelas de reserva. O índice utilizado para percorrer a variável reserva determina o próximo terminal a ser atendido.

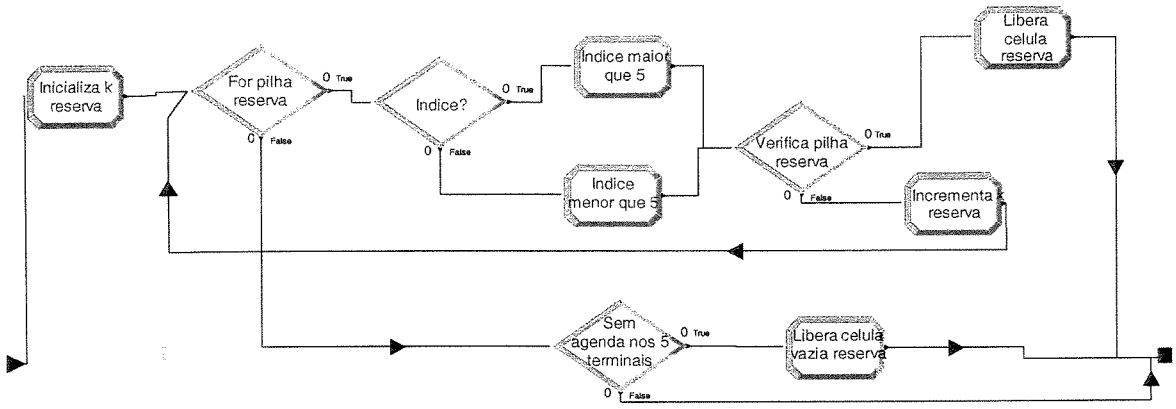


Figura 3.20 – Disciplina de utilização das janelas de reserva.

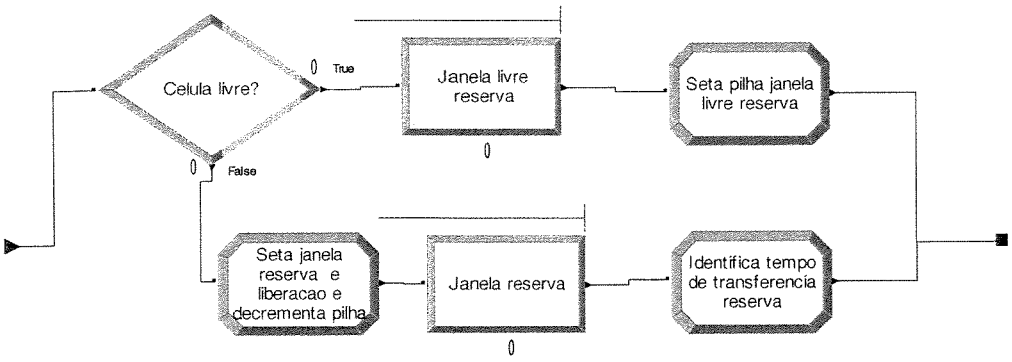


Figura 3.21 – Uso da janela de reserva.

Após a utilização das janelas de reserva ou após o agendamento, o processo *Janela livre de tráfego* (vide Figura 3.22) é executado para verificar a existência de células nos buffers dos terminais para os quais não há reserva. Caso haja pelo menos uma célula, em um dos terminais, e não exista reserva para esse terminal, a célula é liberada e enviada ao buffer *Célula em contenção*. Caso contrário, uma célula vazia é liberada do *Buffer janela livre de colisão e de tráfego* para ocupar a janela de contenção do próximo quadro e identificar uma janela livre (de tráfego) (vide Figura 3.7).

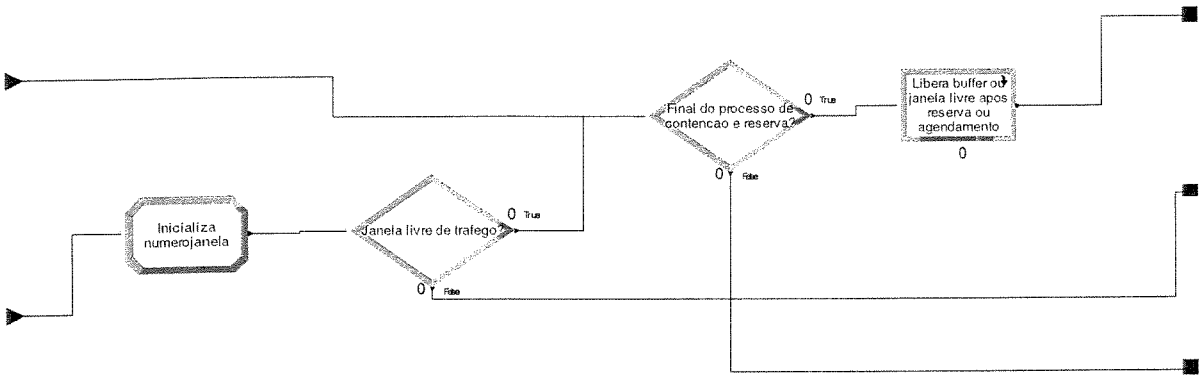


Figura 3.22 – Janela livre de tráfego.

III.3 – ALOCAÇÃO ADAPTATIVA DE JANELAS DE CONTENÇÃO

A alocação adaptativa de janelas de contenção é modelada pelo diagrama mostrado na Figura 3.23, no qual estão representados dez terminais (A, B, C, D, E, F, G, H, I e J). O número de janelas no quadro é fixo e igual a cinco e o objetivo do modelo é alocar dinamicamente as janelas de reserva e de contenção dentro do quadro de *uplink*. As janelas de reserva são alocadas no início do quadro e as janelas de contenção, no final do quadro. Os terminais utilizam as janelas de contenção para requisição das janelas de reserva. O algoritmo usado para resolução da contenção é o *ternary tree*.

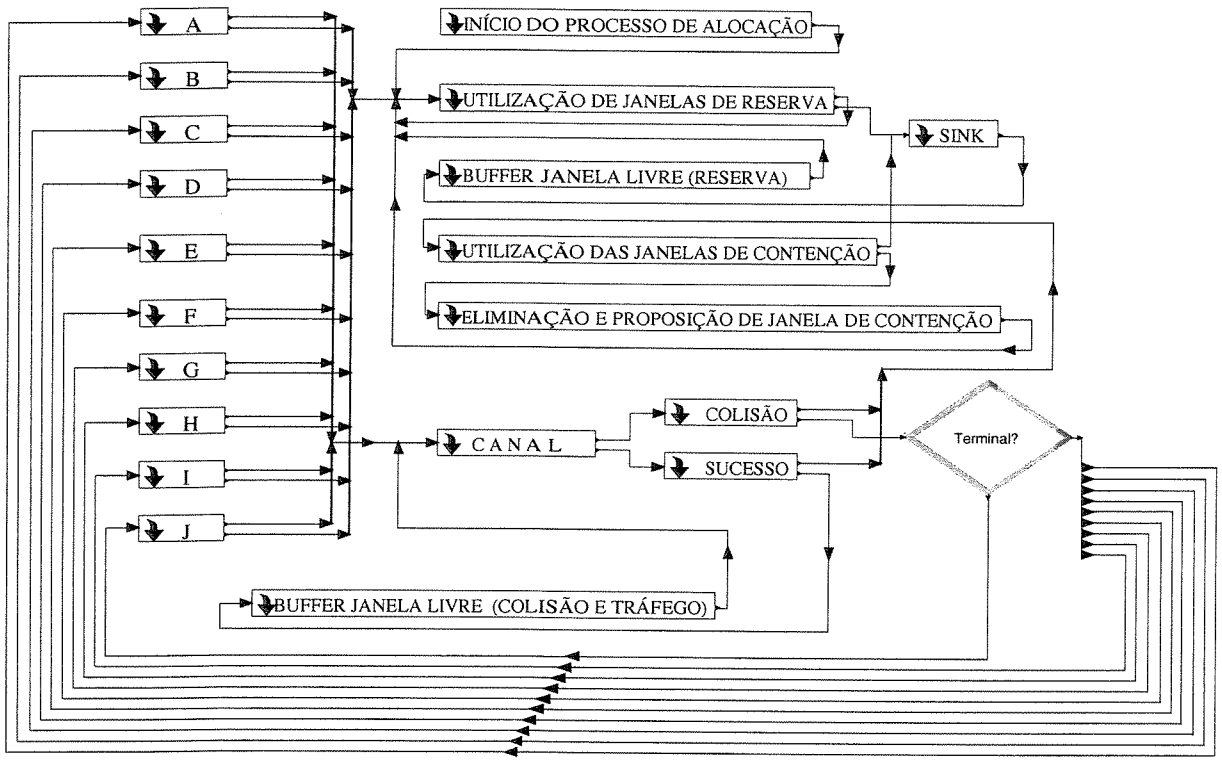


Figura 3.23 - Modelo de alocação adaptativa de janelas de contenção.

O modelo de *Terminal* (A, B, C,...,J) possui apenas uma alteração em relação ao descrito em III.2 (vide Figura 3.14): a célula de contenção só é liberada pelo primeiro buffer, *Buffer A*, quando não existe reserva de células agendada para esse terminal e a célula anterior de contenção desse terminal, tenha sido transmitida com sucesso, mesmo que o processo de contenção, no qual o terminal estava inserido, ainda não tenha terminado.

O modelo *Buffer janela livre (de colisão e de tráfego)* foi descrito em III.1 (vide Figura 3.1), o modelo *Buffer janela livre (de reserva)* foi descrito em III.2 (vide Figura 3.13) e o modelo de *Canal* foi descrito em III.2 (vide Figura 3.15).

O modelo *Início do processo de alocação* (vide Figura 3.24), gera uma única célula no instante zero para dar início à utilização das janelas de reserva. Após este instante, esse modelo não é mais executado.

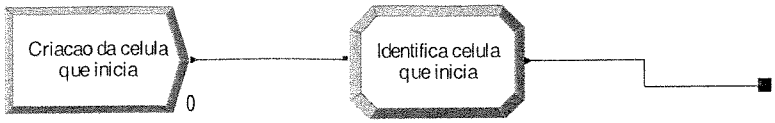


Figura 3.24 – Início do processo de alocação.

Na ocorrência de colisão, após o tratamento da pilha descrito em III.1 (vide Figura 3.3), uma das células colididas é duplicada para dar continuidade ao processo de *Utilização das janelas de contenção* (vide Figura 3.25), caso não tenha se esgotado o número total de janelas de contenção do quadro.

As células que participaram do processo de colisão são identificadas e encaminhadas aos terminais de origem (buffer de contenção), para, dependendo do valor da pilha e do grupo de contenção ao qual pertencem, participarem de uma nova tentativa no próximo quadro.

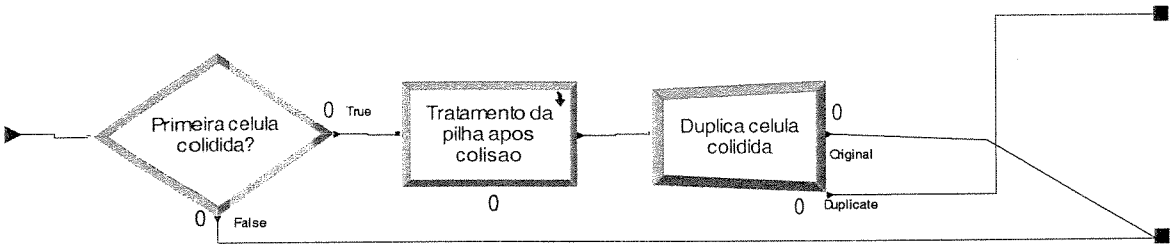


Figura 3.25 – Modelo de colisão.

Se a célula que obteve sucesso é uma célula vazia, ela é duplicada. A original volta ao buffer *Buffer janela livre (colisão e tráfego)* e a duplicata dá continuidade ao processo de *Utilização das janelas de contenção*. Se a célula de sucesso não é uma célula vazia, ela é encaminhada ao processo de *Utilização das janelas de contenção* (vide Figura 3.26).

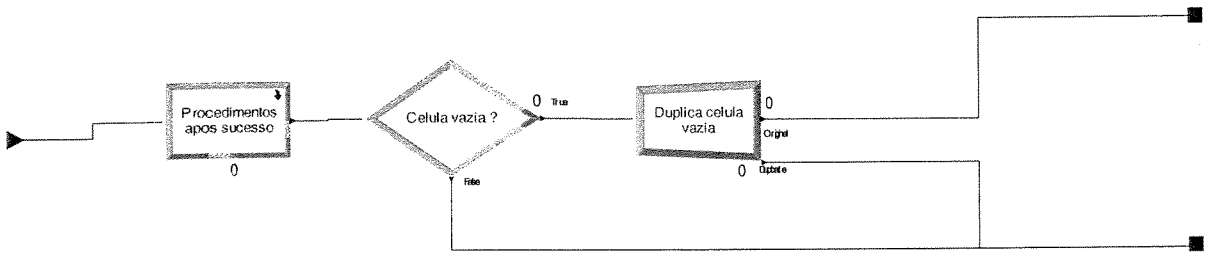


Figura 3.26 – Modelo de sucesso.

Quando não há colisão, no modelo *Procedimentos após sucesso* é computado o tempo de transferência da célula; é feito o agendamento, atualizando-se a variável reserva com o conteúdo da variável agenda; é incrementado o contador de células transmitidas por janela de contenção; é feito um decremento da pilha para os terminais, que se mantêm no grupo de contenção e que ainda não obtiveram sucesso; é indicada a liberação do terminal para participar de outro grupo de contenção no final do quadro, se o terminal teve sucesso; e é verificado o término do processo de contenção do grupo (vide Figura 3.27).

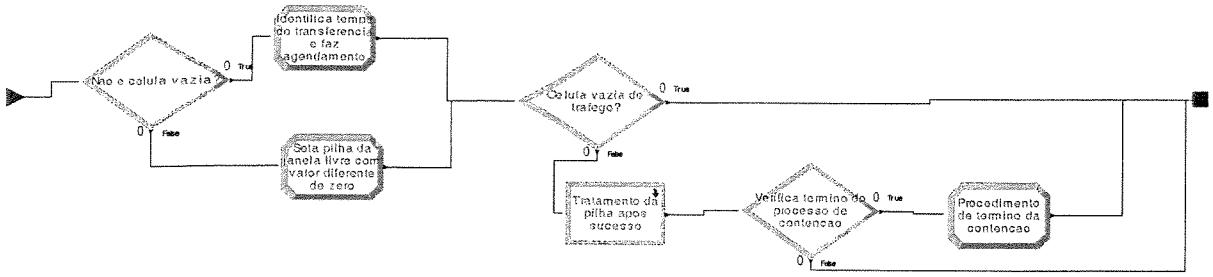


Figura 3.27 – Modelo de procedimentos após sucesso.

O processo de *Utilização das janelas de reserva* é mostrado na Figura 3.28. A disciplina de uso das janelas de reserva disponíveis no quadro foi descrito em III.2 (vide Figura 3.20). Quando há o uso da janela de reserva, é computado o tempo de transferência da célula; é incrementado o contador de células transmitidas por janela de reserva e é executado o procedimento de *piggybacking*, atualizando-se a variável reserva com a informação do número de células existentes no primeiro buffer.

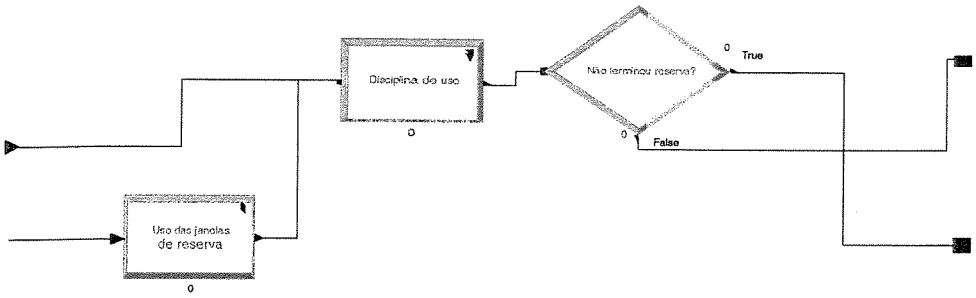


Figura 3.28 – Utilização das janelas de reserva.

O processo de *Utilização das janelas de contenção* é mostrado na Figura 3.29 e verifica para cada janela de contenção disponível no quadro, se ela está em processo de contenção ou se ela está liberada para participar de um novo processo de contenção. Uma célula é liberada pelo terminal para participar de um novo processo de contenção, quando o terminal possui célula para transmitir, não possui reserva agendada e não está participando de outro processo de contenção. Uma célula é liberada pelo terminal, que já participa de um grupo de contenção, quando sua variável pilha é igual zero.

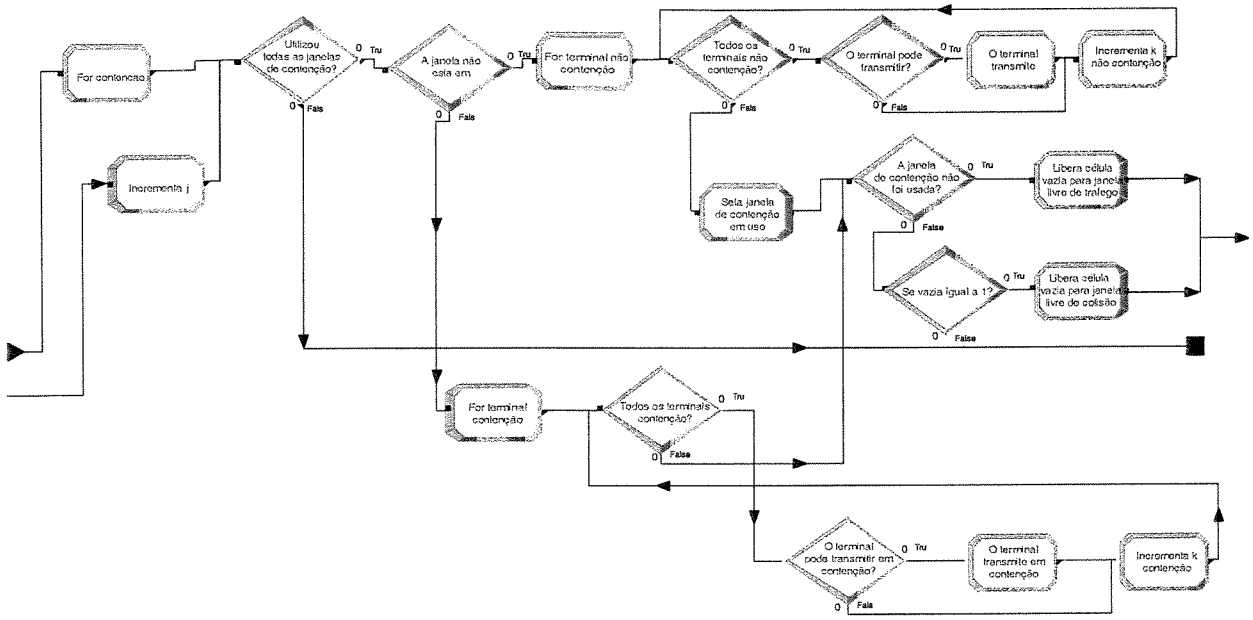


Figura 3.29 – Utilização das janelas de contenção.

O processo de Eliminação e proposição de janela de contenção possui os modelos de: Libera terminal, Computo de ncr e ntr, Eliminação e Proposição de uma janela de contenção (vide Figura 3.30).

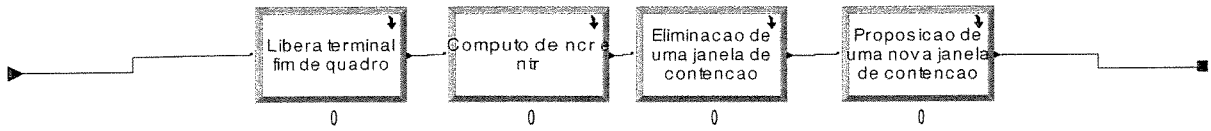


Figura 3.30 – Eliminação e proposição de janela de contenção.

O modelo *Libera terminal* é executado a cada final de quadro. Se existe indicação de liberação, os terminais são liberados para participarem de um novo processo de contenção no próximo quadro (vide Figura 3.31).

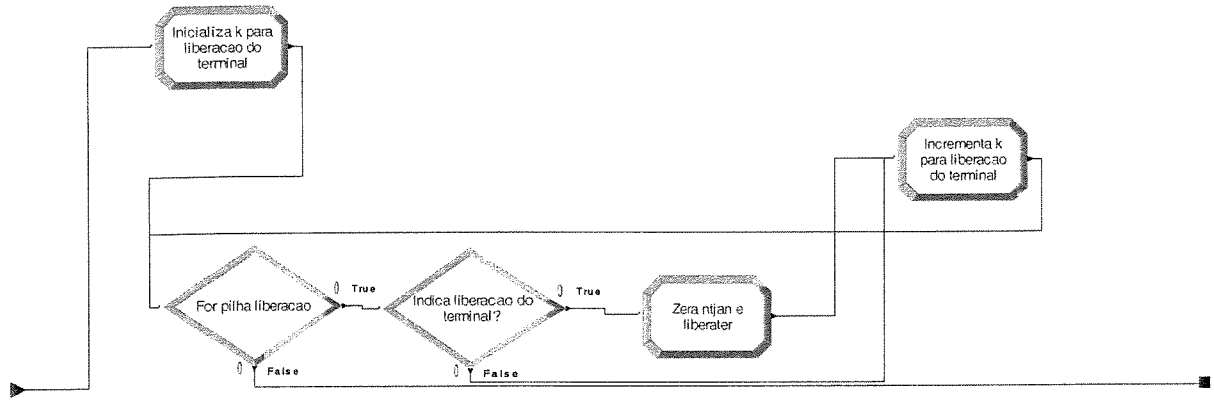


Figura 3.31 – Libera terminal.

No modelo de *Computo de ncr e ntr*, a variável reserva é percorrida e são computados o número de células reservadas (ncr) e o número de terminais com reserva agendada (ntr) a cada quadro (vide Figura 3.32).

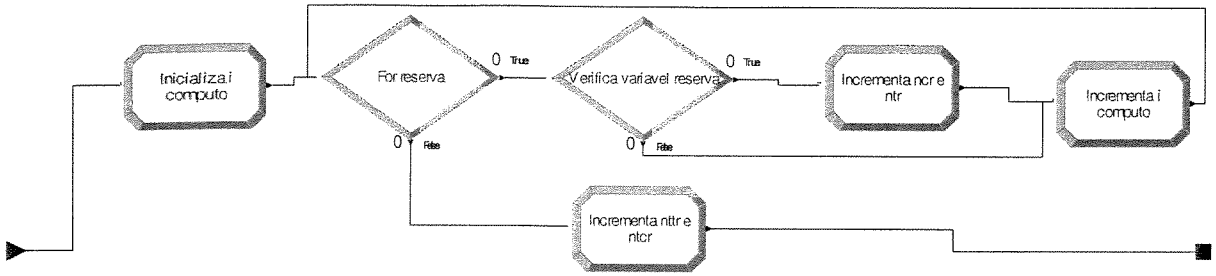


Figura 3.32 – Cômputo de ncr e ntr.

No modelo de *Eliminação* (vide Figura 3.33), para cada janela de contenção do quadro corrente, verifica-se o término do processo de contenção. Se o processo se encerrou, elimina-se a janela no próximo quadro. Se a janela eliminada não é a última do quadro, o mecanismo de aglutinação das janelas de contenção é executado, porque elas sempre ocupam a parte final do quadro (vide Figura 3.34).

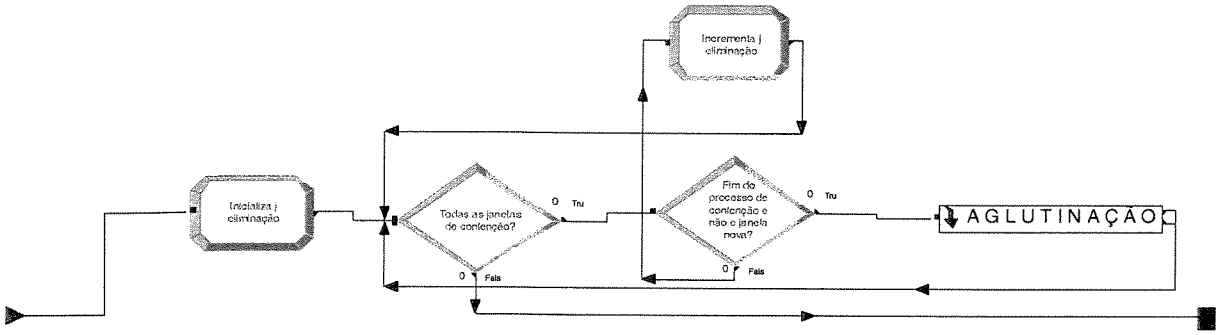


Figura 3.33 – Eliminação.

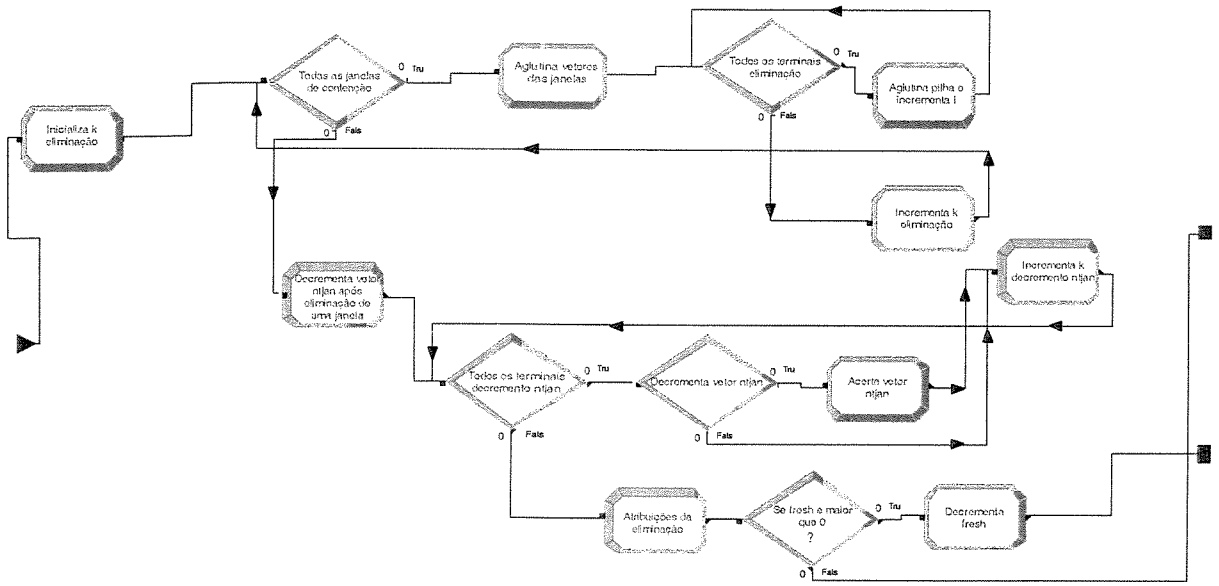


Figura 3.34 – Aglutinação das janelas de contenção.

No modelo *Proposição* (vide Figura 3.35), uma nova janela de contenção é proposta no próximo quadro, se no quadro corrente houve proposição de nova janela, houve uso dessa janela e o número de terminais com reserva agendada no quadro é menor que o número de terminais existentes. Nesta situação, a variável *espera*, que controla o número de quadros sem proposição de janela, é zerada. Se houve uso da nova janela de contenção proposta, a variável auxiliar *propor* recebe o conteúdo da variável *espera*.

A nova janela proposta é eliminada em duas situações: quando não foi usada e o número de células reservadas é maior que o número de janelas de reserva do quadro, ou, quando não foi usada e o número de terminais com reserva agendada é igual ao número de terminais existentes. Se a nova janela proposta é eliminada, a variável *espera* é incrementada unitariamente.

A variável *propor* é decrementada unitariamente toda vez que no quadro não há proposição de nova janela de contenção. Se o número de células reservadas é menor que o número de janelas de reserva do quadro ou se o conteúdo da variável *propor* chega a zero, propõe-se uma nova janela de contenção no quadro seguinte. Esta proposição só acontece se o número de terminais com reserva agendada é menor que o número de terminais existentes. Se o número de janelas do quadro se esgotou, portanto não é possível propor uma nova janela, a variável *propor* recebe o valor um para que se tenha a oportunidade de propor uma nova janela de contenção no quadro seguinte ou logo que possível. Lembre-se que não é possível propor uma janela de contenção, se no quadro seguinte todas as janelas forem de contenção, ou se todos os terminais têm reserva agendada.

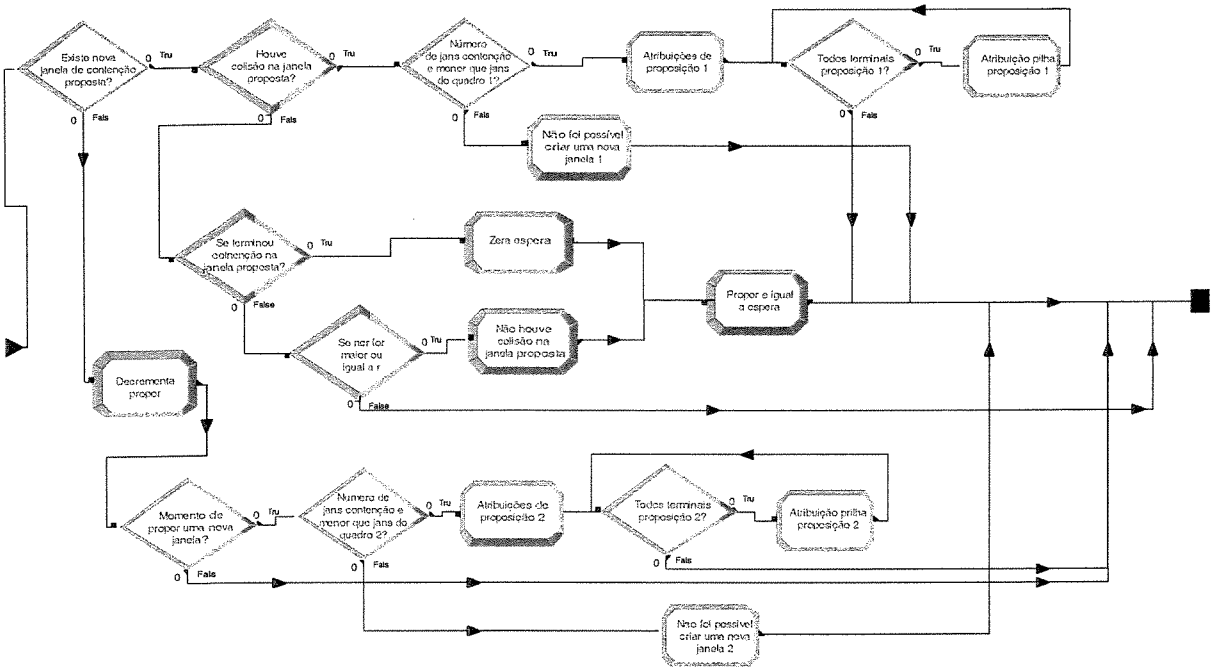


Figura 3.35 – Proposição.

III.4 – QUADRO DE TAMANHO VARIÁVEL

A alocação adaptativa de janelas de contenção e de reserva no quadro é modelada por um diagrama igual ao mostrado na Figura 3.23, no qual estão representados dez terminais (A, B, C, D, E, F, G, H, I e J). O número de janelas no quadro é variável:

- número de janelas de contenção do quadro é ilimitado e definido pelo algoritmo de alocação adaptativa de janelas de contenção, modelado no item III.3, no qual é proposta uma nova janela de contenção em todos os quadros salvo no caso de todos os terminais possuírem reserva agendada, ou se o mecanismo de desestímulo estiver atuante. O mecanismo de desestímulo passa a atuar quando se propõe uma nova janela de contenção, não há uso desta nova janela e o número de células reservadas é maior que zero.
- o número de janelas de reserva do quadro é igual ao número de terminais com reserva agendada no quadro;
  - ✓ no modelo anterior, III.3, o número de janelas de reserva era igual ao número de janelas do quadro menos o número de janelas de contenção;
  - ✓ o limite do número de janelas de reserva é igual ao número de terminais existentes.

As alterações foram implementadas no modelo *Proposicao*, vide Figura 3.36, nos blocos que estão em negrito.

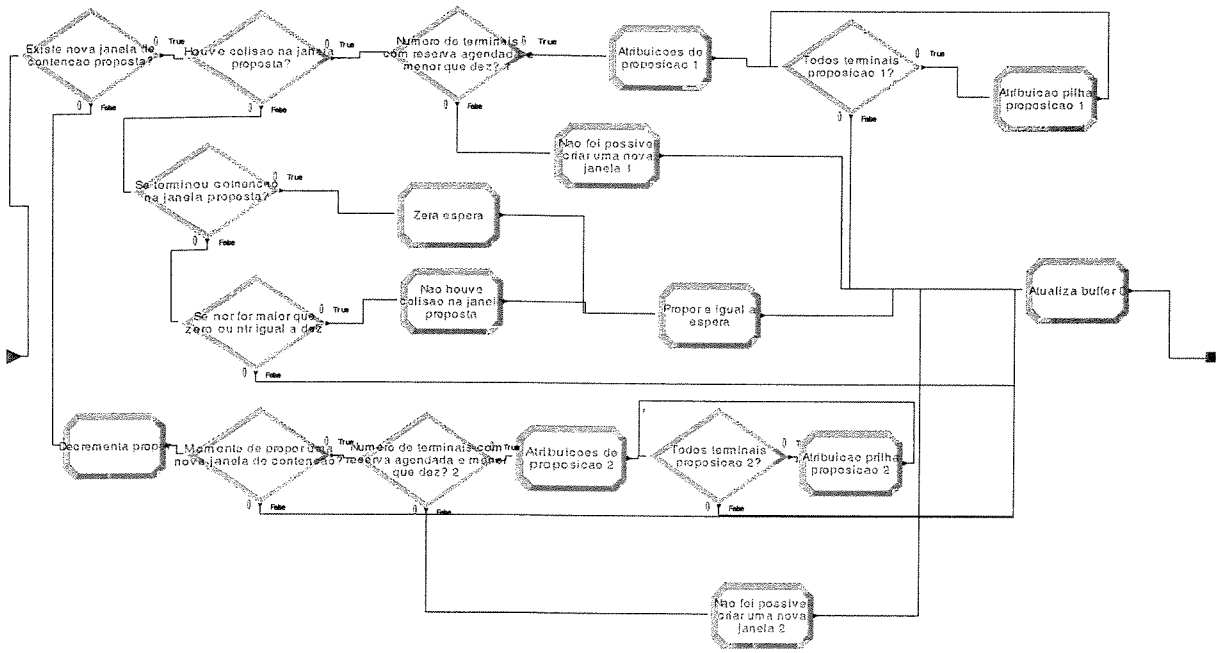


Figura 3.36 – Proposição.

Na resolução de contenção, a estação rádio-base comporta-se, para estimar o fim do processo, como um terminal que participa do processo de contenção, mas não participa da colisão, isto é, ela soma 2 na posição da pilha virtual em cada colisão. Esta consideração foi abordada no modelo de *Tratamento da pilha após colisão*, vide Figura 3.37, que está contido no modelo de *Colisão* (vide Figura 3.25) e no modelo de *Procedimentos após sucesso*, vide Figura 3.38, que está contido no modelo de *Sucesso* (vide Figura 3.26).

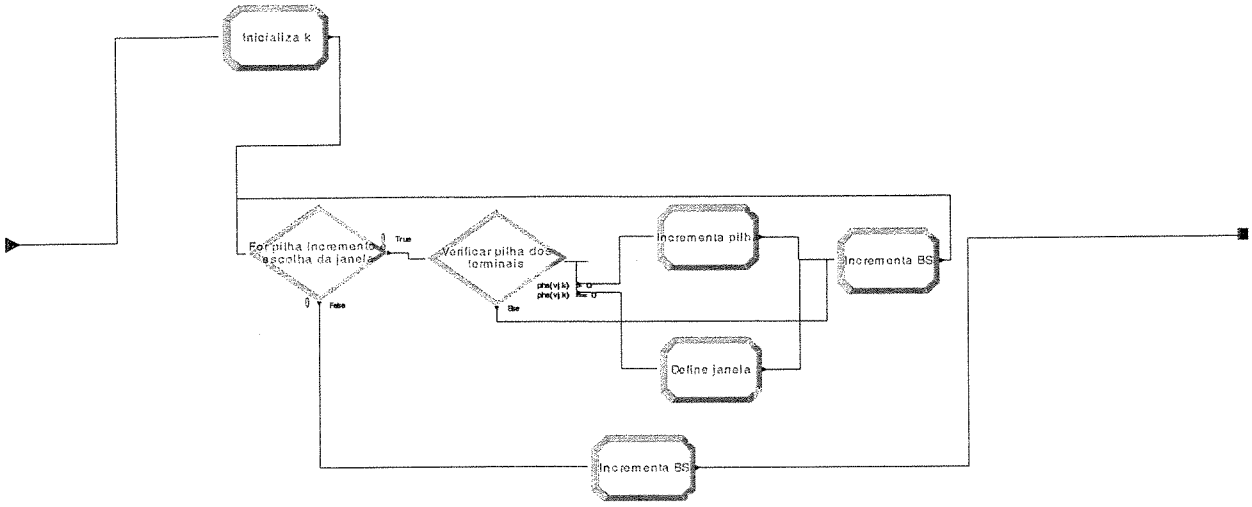


Figura 3.37 – Tratamento da pilha após colisão.

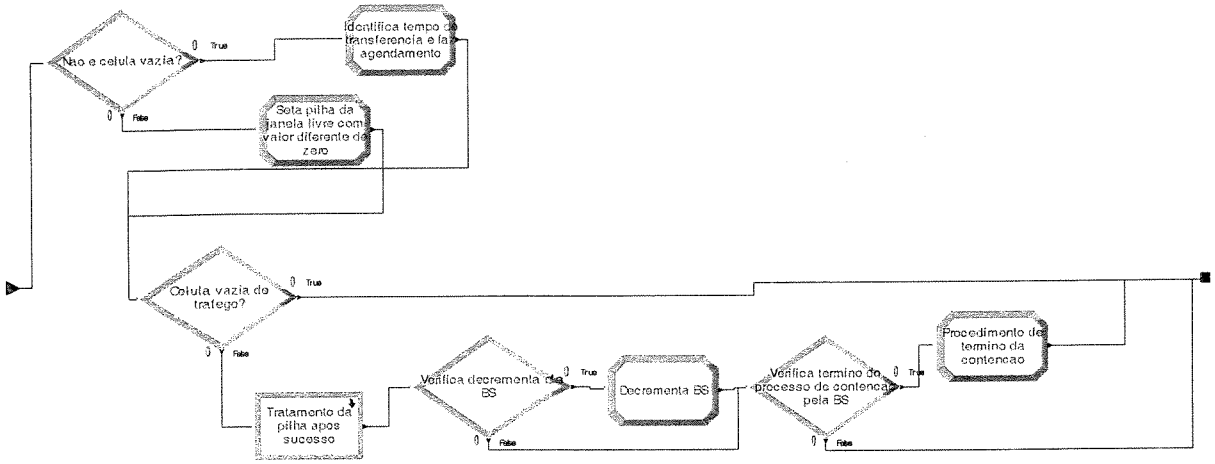


Figura 3.38 – Procedimentos após sucesso.

III.5 – MINI-JANELA

Nos modelos em que foi implementada a utilização da mini-janela para solicitação de reserva, houve as seguintes alterações:

- no modelo de *Terminal* (*Fonte do terminal A, B, C, D,...., J*), vide Figura 3.39. O terminal é modelado por uma fonte (*create*); por um *buffer* (*hold*) que armazena as células em espera; por um duplicador (*separate*) que duplica a célula que sai do *Buffer* e por um *buffer* (*hold*) que controla a célula que participa do processo de contenção. A célula é liberada pelo *Buffer* em duas situações: quando participa de um processo de contenção ou quando usa as janelas de reserva. Após a célula ser liberada pelo *Buffer* decide-se se a célula será encaminhada ao processo de *Utilização das janelas de reserva* (vide Figura 3.28), caso ela tenha sido liberada pelo mesmo processo ou, ela será duplicada, caso tenha que participar da contenção para solicitação de reserva. A original é enviada de volta ao *Buffer* para posterior transmissão e a duplicata é encaminhada ao *Canal* para participar do processo de contenção. A célula de contenção leva a identificação do terminal e o número de células existentes no *Buffer* para posterior agendamento.

- no modelo de *Canal*, a primeira célula que participa de um processo de contenção ocupa o recurso *Utilização da janela* durante o tempo de janela que é de um quarto do tempo da janela de reserva. Se durante esse tempo, outra célula chega para ocupar o recurso, a colisão é identificada e essa célula é submetida a um atraso *(Tempo de janela)* igual a um quarto do tempo da janela de reserva (vide Figura 3.40). As alterações no modelo de *Canal* foram feitas nos blocos em negrito.

- no modelo de *Procedimentos após sucesso*, não é computado o tempo de transferência da célula que participa do processo de contenção. O tempo de transferência é computado quando há transmissão de célula por janela de reserva. Esta alteração foi feita no bloco em negrito da Figura 3.41.

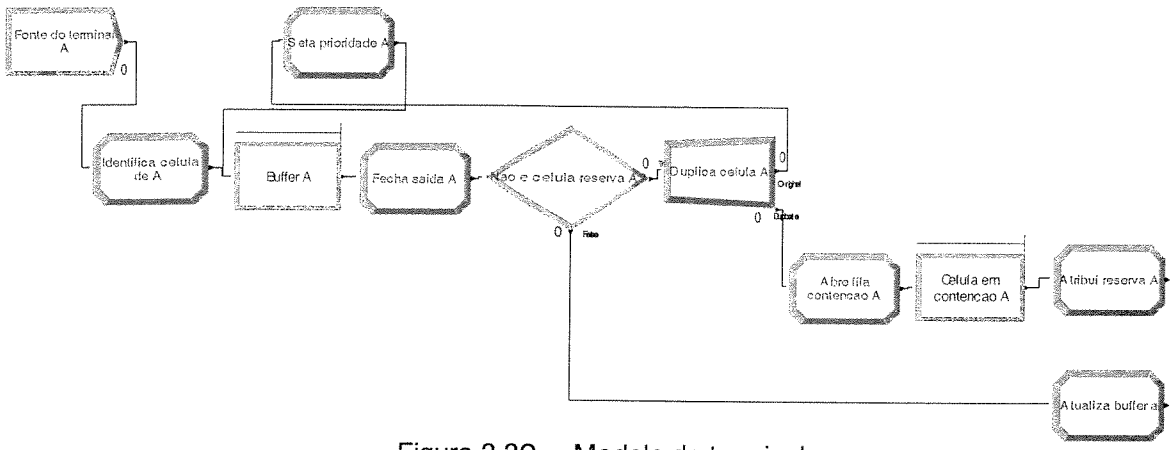


Figura 3.39 – Modelo de terminal.

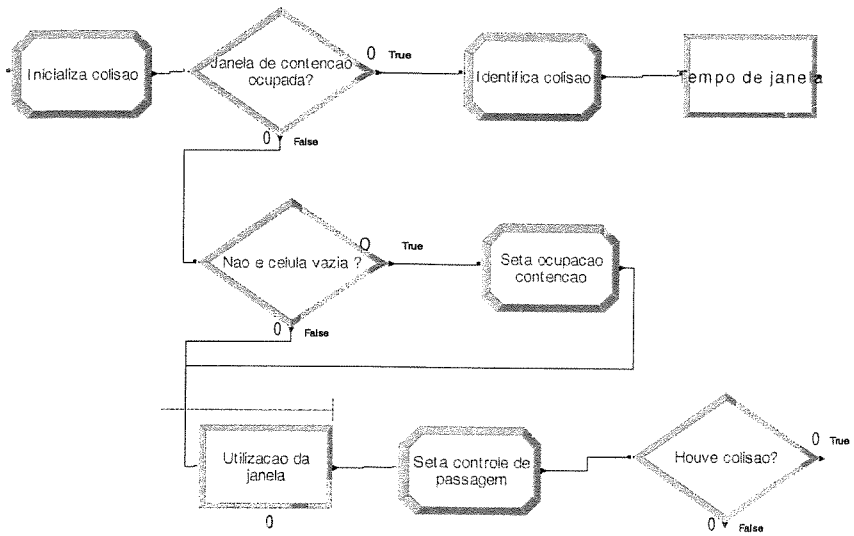


Figura 3.40 – Modelo de canal.

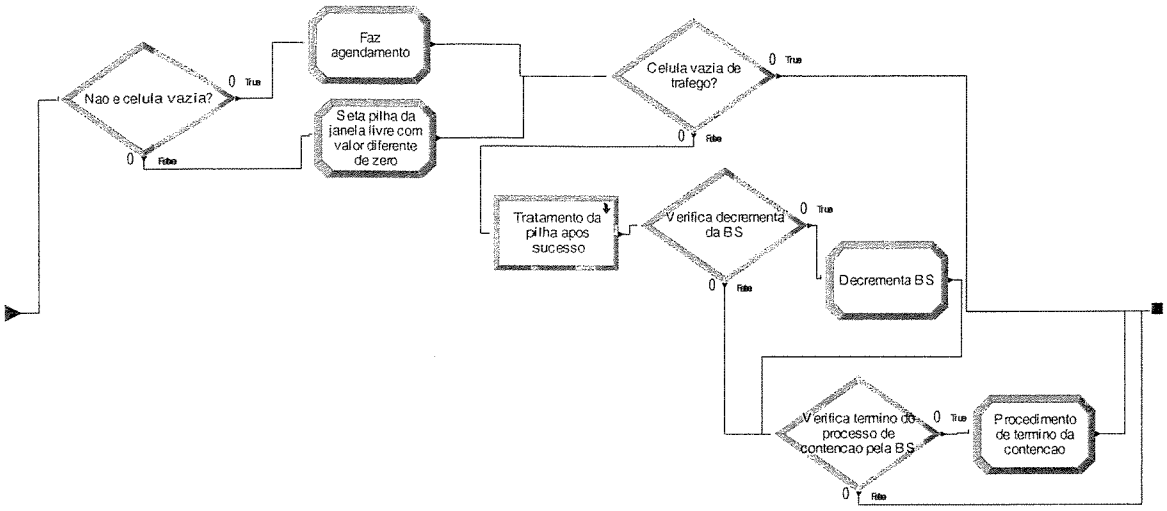


Figura 3.41 – Modelo de procedimentos após sucesso.

III.6 – PRIORIDADE

Nos modelos em que foi implementada a prioridade para o tráfego em tempo real, houve as seguintes alterações:

- no modelo de *Terminal* (*Fonte do terminal A, B, C, D,...., J*), 50% dos terminais foram identificados como sendo de tráfego em tempo real e os demais com tráfego não em tempo real. Foi introduzido um limitador de *buffer:* os terminais *rt* possuem um *buffer* de tamanho 10 e os terminais *nrt*, de tamanho 100 (vide Figura 3.42).

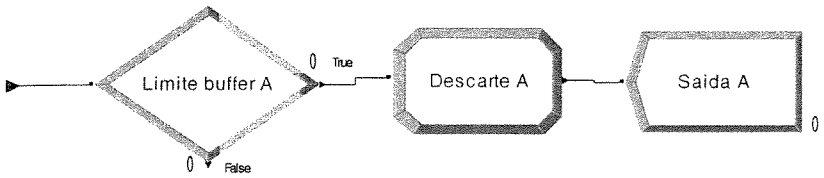


Figura 3.42 – Limitador de buffer.

- no modelo Tratamento da pilha após colisão, vide Figura 3.43, na resolução de contenção, é utilizado o algoritmo *binary tree* para os terminais com tráfego em tempo real e o *ternary tree* para os terminais com tráfego não em tempo real.

- no modelo de *Disciplina de utilização das janelas de reserva*, atende-se primeiro os terminais com tráfego em tempo real, vide Figura 3.44:
  - ✓ se existe pelo menos um terminal rtVBR com reserva agendada, o número de janelas de reserva do quadro é igual ao número de terminais rtVBR com reserva agendada;
    - ✓ se nenhum terminal rtVBR possui reserva agendada, mas existe pelo menos um terminal nrtVBR com reserva agendada, o número de janelas de reserva do quadro é igual ao número de terminais nrtVBR com reserva agendada;
      - ✓ se nenhum terminal nrtVBR possui reserva agendada, o número de janelas de reserva do quadro é zero.

- no modelo de *Utilização das janelas de reserva*, vide Figura 3.45, é computado o tempo de transferência da célula (*rt* ou *nrt*); é executado o mecanismo de *piggybacking*, atualizando-se a variável reserva com o conteúdo do *buffer* do terminal e é incrementado o contador de células transmitidas (*rt* ou *nrt*).

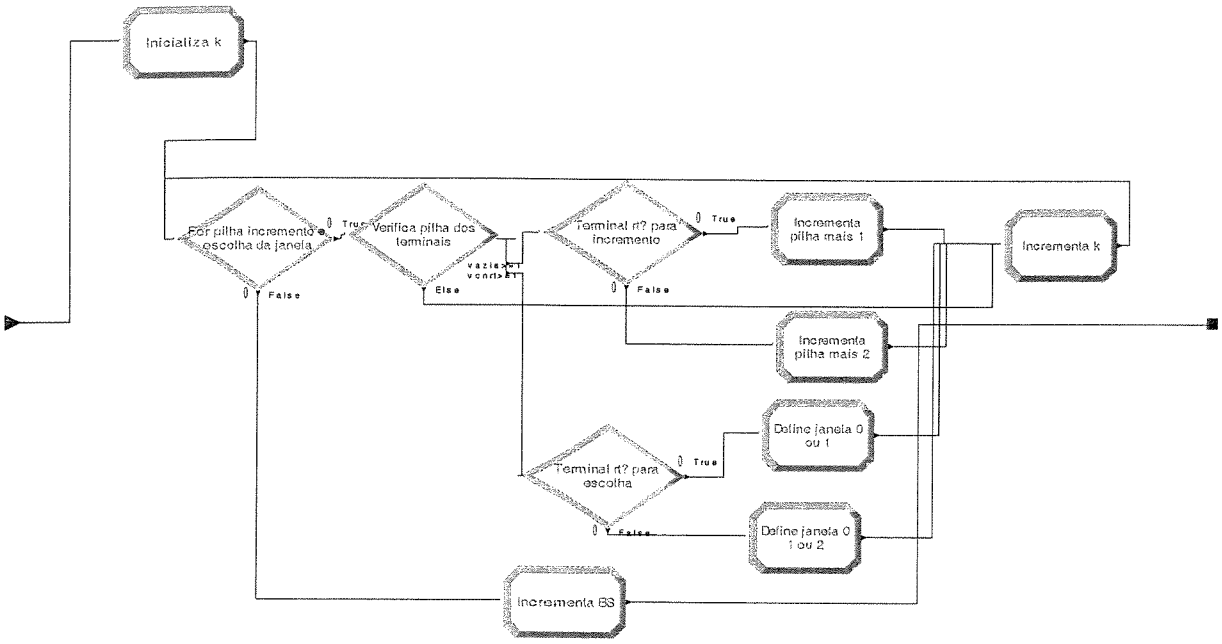


Figura 3.43 – Tratamento da pilha após colisão.

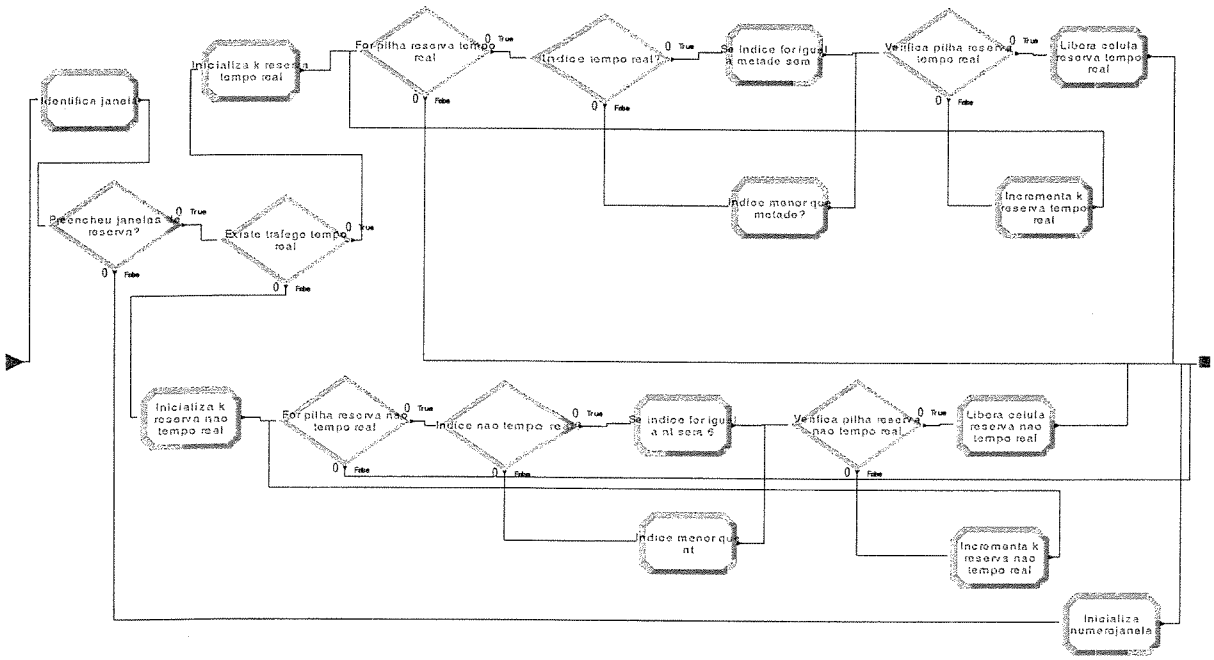


Figura 3.44 – Disciplina de utilização das janelas de reserva.

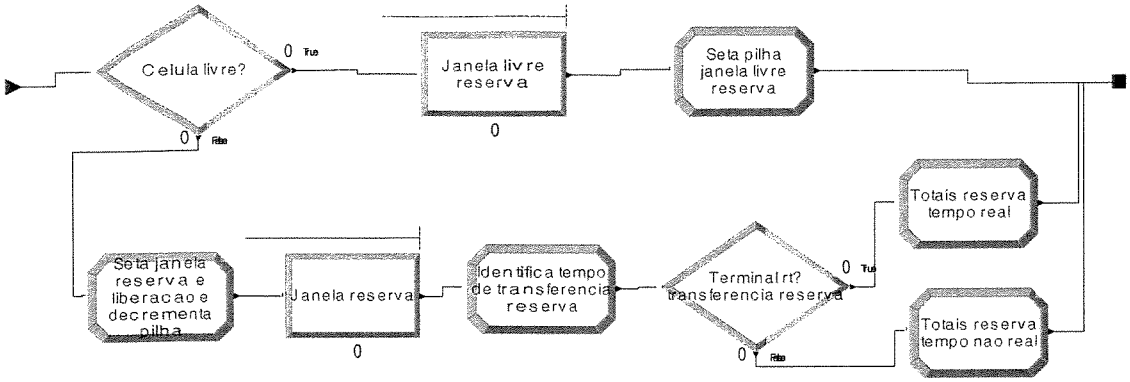


Figura 3.45– Utilização das janelas de reserva.

CAPÍTULO IV – ESTUDO DE CASOS

IV.1 – QUADRO FIXO COM TRÁFEGO POISSONIANO

IV.1.1 – Alocação estática

Na simulação do modelo de agendamento cíclico existem cinco terminais e utilizou-se a alocação estática de janelas. O quadro possui quatro janelas, onde a primeira é de contenção e as demais de reserva. A Figura 4.1 mostra a ocupação da janela de contenção e das janelas de reserva em função da carga, usando simulação de evento discreto com 125.000 quadros. Uma carga de 100% significa a geração, em média, de uma célula por janela.

As Figuras 4.2, 4.3, 4.4 apresentam, respectivamente, o número médio de células no *buffer*, o tempo médio de espera e de transferência e o tempo médio de contenção.
O número de eventos das simulações foi determinado para garantir intervalos de confiança da ordem de 1% do valor médio estimado com 95% de confiabilidade.

Observa-se na Figura 4.1 que, para cargas menores que 0.2, a transmissão das células ocorre, praticamente, apenas por contenção, isto é, quase não são usadas as janelas de reserva. Observa-se na Figura 4.2 que o número médio de células no *buffer*, para a carga de 0.2, é menor que 1. Ainda na Figura 4.1, observa-se que a ocupação da janela de contenção cresce até um percentual máximo, de aproximadamente 74% quando a carga é de 0.4. Após a carga de 0.4, a ocupação decresce lentamente, até a carga de 0.65, enquanto a ocupação da janela de reserva está crescendo linearmente. Após a carga de 0.65, a ocupação da janela de contenção decresce com uma maior ocupação das janelas de reserva, e a partir da carga de 0.8, a ocupação da janela de contenção chega muito próximo de zero, onde praticamente todas as células são transmitidas através das janelas de reserva.

A Figura 4.4 mostra que, a partir de uma carga de 0.5, o número de colisões no processo de contenção decresce com o aumento da carga. Os terminais envolvidos na contenção, quando conseguem sucesso, passam por uma fase de utilização da reserva, não participando imediatamente de novas contenções. Observa-se que esse efeito provoca uma mudança na taxa de crescimento das curvas para o tempo de espera no *buffer* e para o tempo de transferência (vide Figura 4.3).

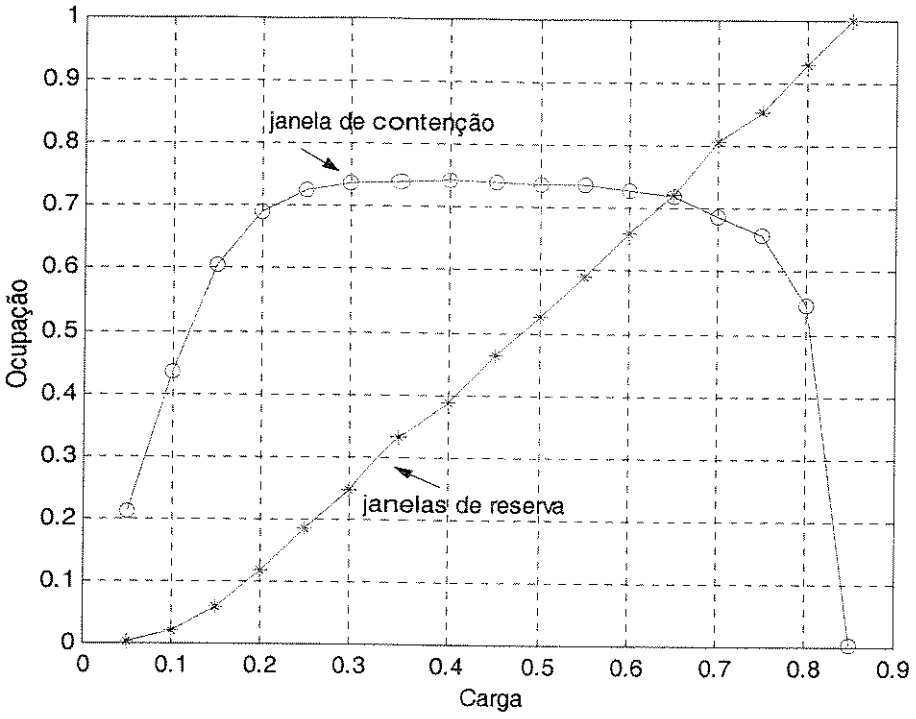


Figura 4.1 – Ocupação da janela de contenção e das janelas de reserva no agendamento cíclico com chegadas poissonianas.

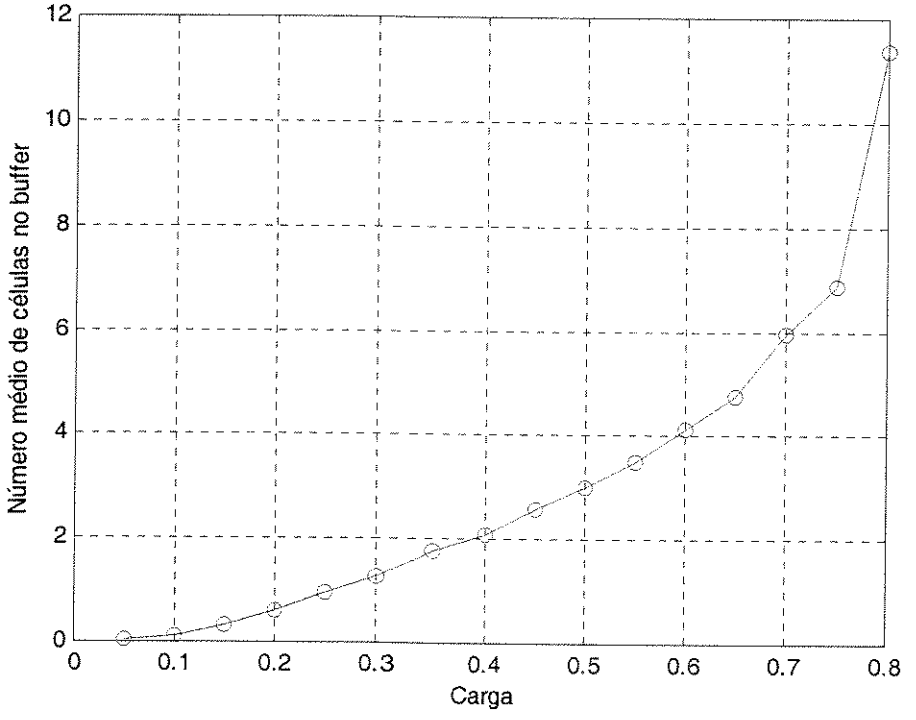


Figura 4.2 – Número médio de células no *buffer* usando agendamento cíclico com chegadas poissonianas.

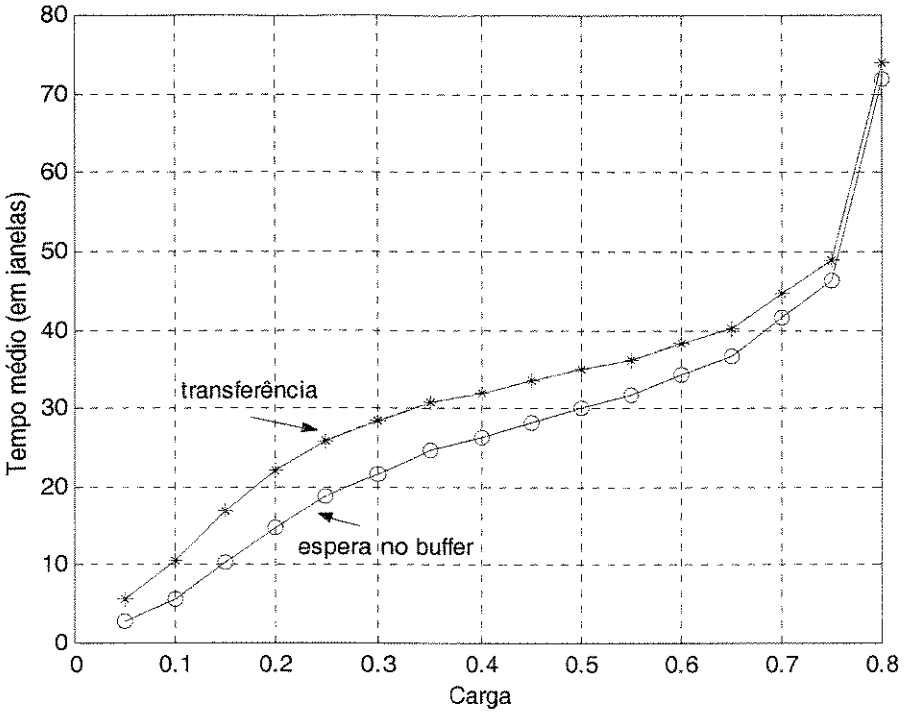


Figura 4.3 – Tempo médio de espera no *buffer* e tempo médio de transferência usando agendamento cíclico com chegadas poissonianas.

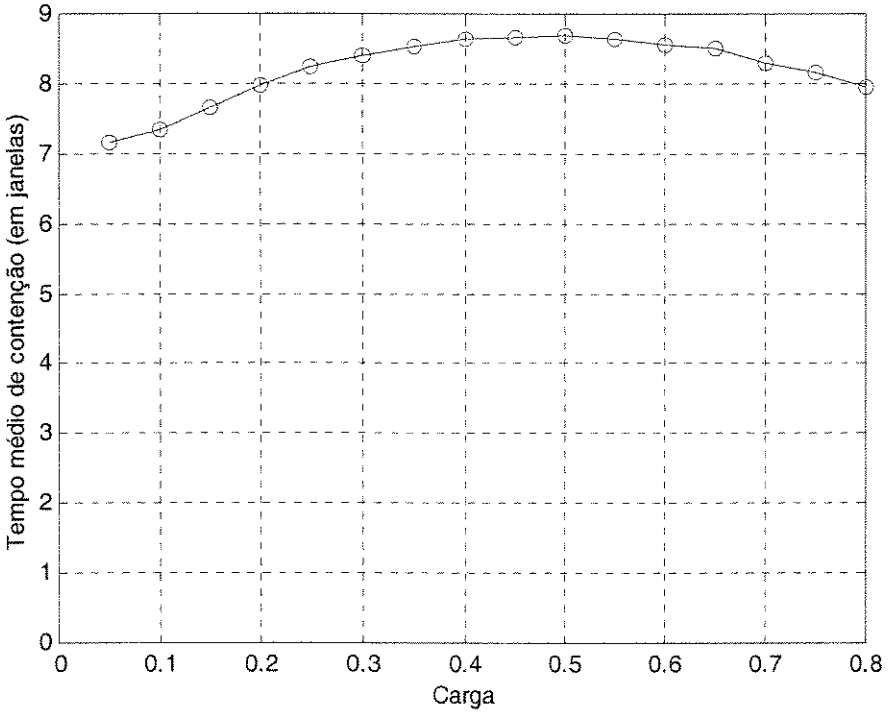


Figura 4.4 – Tempo médio de contenção usando agendamento cíclico com chegadas poissonianas.

IV.1.2 – Alocação estática versus alocação dinâmica com e sem *piggybacking*

As Figuras 4.5 a 4.11 apresentam os resultados obtidos na simulação, de evento discreto com 125.000 quadros, do mecanismo de alocação dinâmica de janelas de contenção, do mecanismo de alocação estática de uma janela de contenção e de ambos, com e sem *piggybacking* para dez terminais. Na simulação do modelo de alocação dinâmica de janelas de contenção, o quadro é fixo e possui cinco janelas. As janelas de contenção e de reserva são alocadas dinamicamente dentro do quadro, conforme o algoritmo descrito em II.3. Na simulação do modelo de uma janela fixa de contenção, o quadro é fixo com cinco janelas, sendo a última de contenção.

As Figuras 4.5, 4.6, 4.7, 4.8, 4.9, 4.10, e 4.11 apresentam, respectivamente, o número de janelas de contenção e de reserva no quadro, o número de terminais por contenção, o tempo de contenção, o número de células no *buffer*, o tempo de espera no *buffer*, o tempo de transferência e a vazão por reserva e contenção.
Em todas as figuras estão apresentados os valores médios. O número de eventos das simulações foi determinado para garantir intervalos de confiança menores que 3% do valor médio estimado com 95% de confiabilidade.

A Figura 4.5 mostra a adaptação do número de janelas de contenção em função da carga de tráfego. Para cargas altas (maior que 0.5), o número de janelas de contenção no algoritmo adaptativo diminui, devido à maior utilização das janelas de reserva, que causa a diminuição do número médio de terminais que disputam as janelas de contenção.
Observa-se na Figura 4.5 que o *piggybacking* reduz o número médio de janelas de contenção no acesso adaptativo. Esta diminuição decorre do aumento efetivo do número de células reservadas por terminal devido ao *piggybacking*.

A Figura 4.6 ilustra a eficiência do algoritmo adaptativo quanto à capacidade de manter reduzido o número de terminais por janela de contenção (menor que 2). A sensível queda no número médio de terminais por janela de contenção no algoritmo sem adaptação, para cargas altas (maior que 0.5), decorre do uso efetivo das quatro janelas de reserva quando a carga é alta.

A Figura 4.7 ilustra também a eficiência do algoritmo adaptativo em manter reduzido o tempo médio de contenção.

As Figuras 4.8, 4.9 e 4.10 mostram o significativo ganho de desempenho do algoritmo de adaptação em relação à alocação fixa do número de janelas de contenção. A importância do uso do mecanismo de *piggybacking* para cargas altas (maior que 0.5) é evidente nestas figuras.

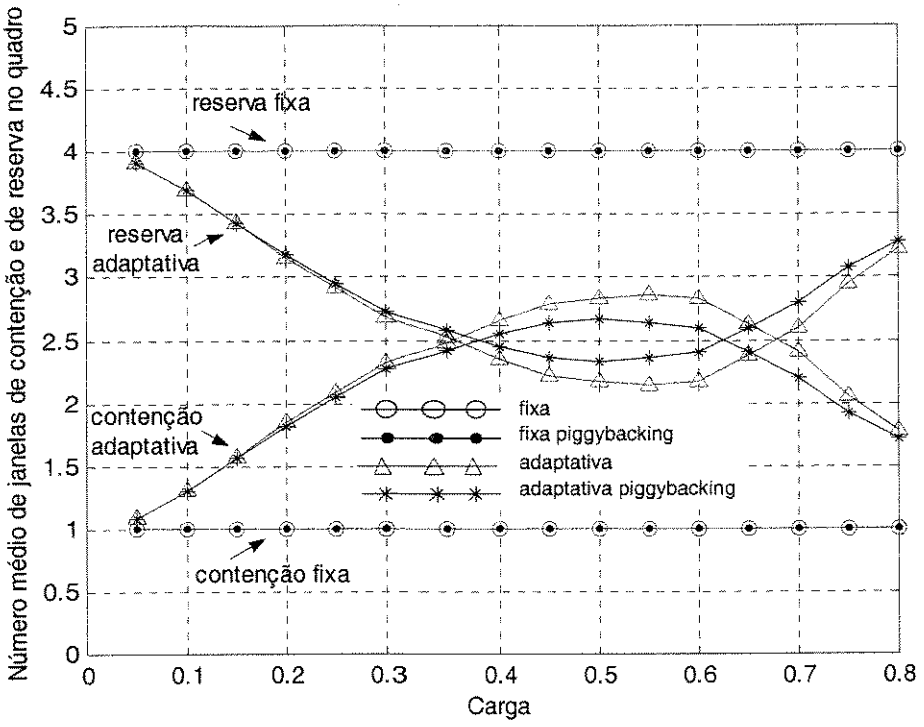


Figura 4.5 – Número médio de janelas de contenção e de reserva no quadro com chegadas poissonianas.

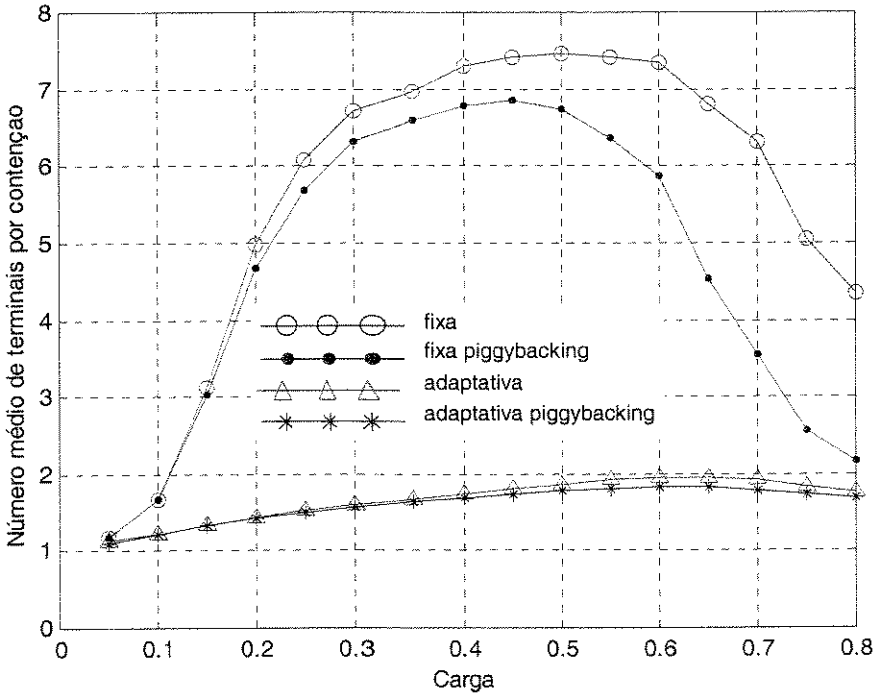


Figura 4.6 – Número médio de terminais por contenção com chegadas poissonianas.

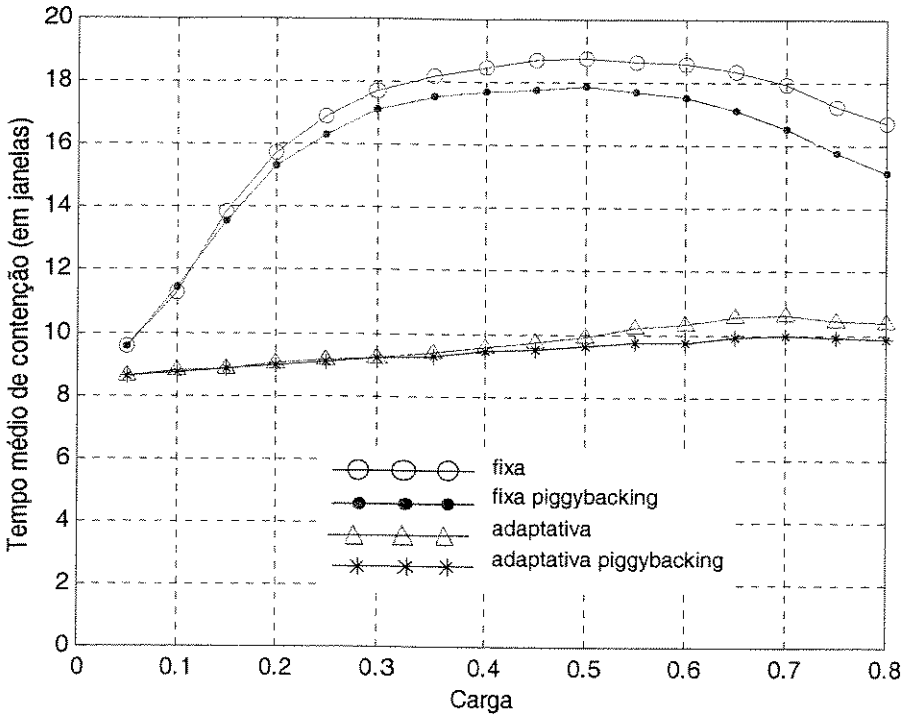


Figura 4.7 – Tempo médio de contenção com chegadas poissonianas.

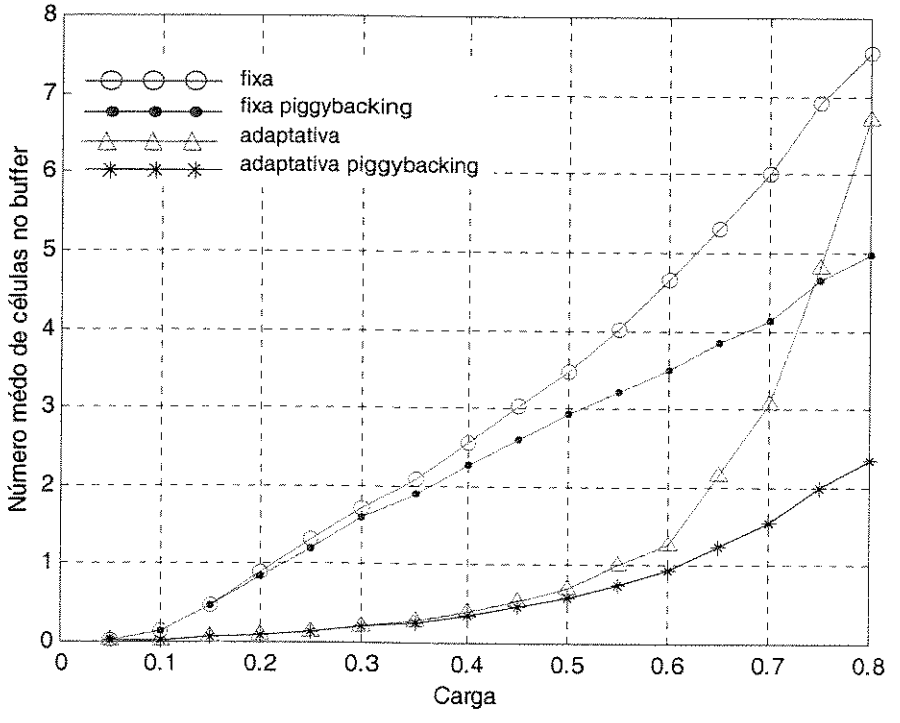


Figura 4.8 – Número médio de células no *buffer* com chegadas poissonianas.

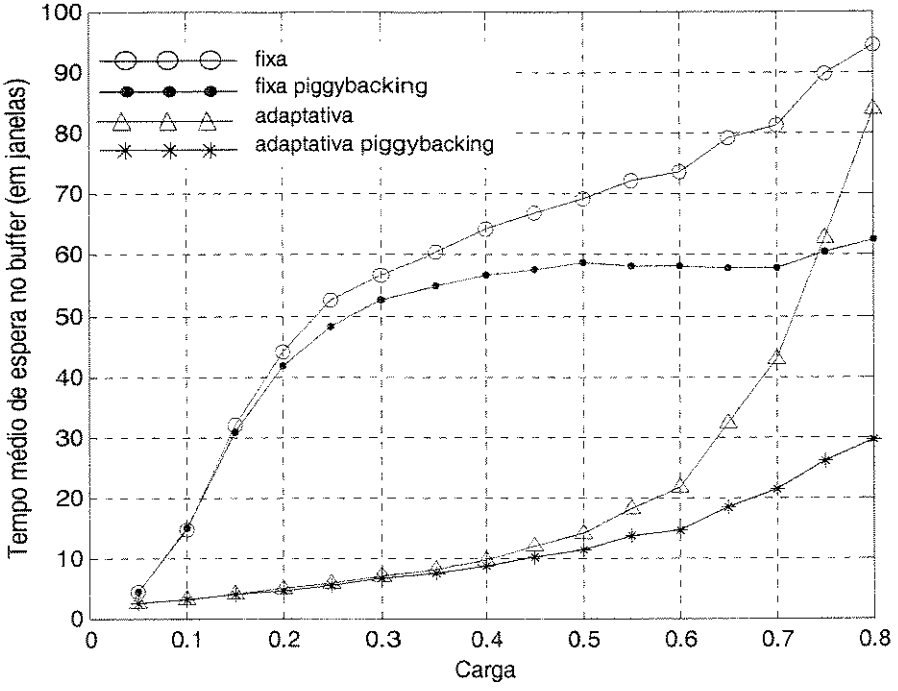


Figura 4.9 – Tempo médio de espera no *buffer* com chegadas poissonianas.

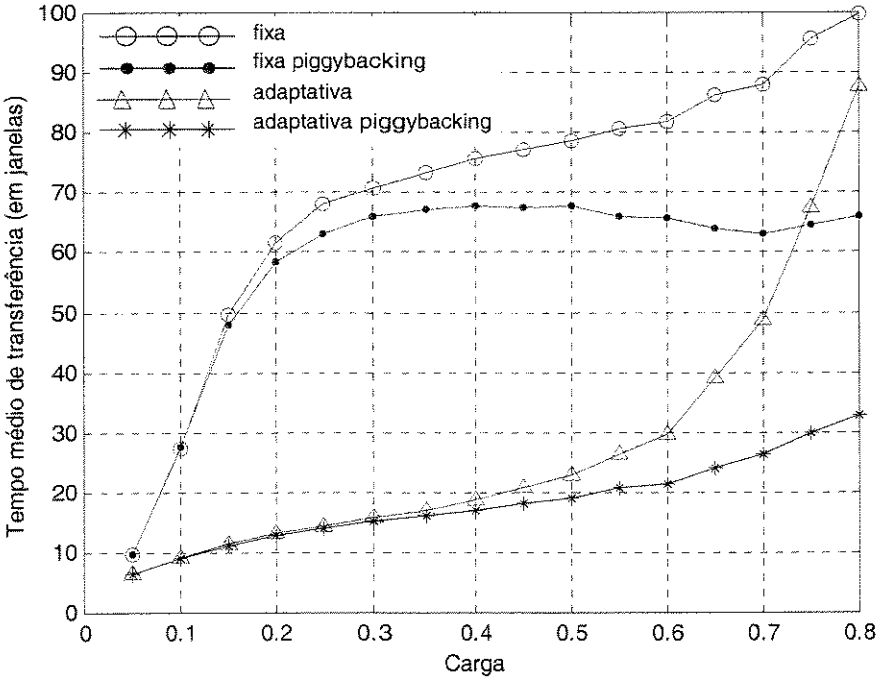


Figura 4.10 – Tempo médio de transferência com chegadas poissonianas.

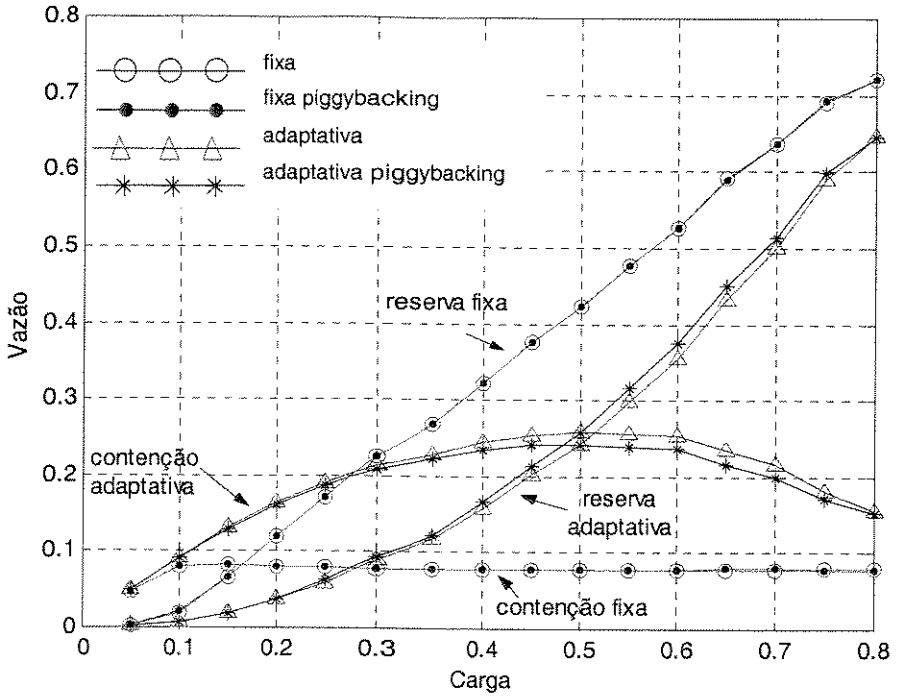


Figura 4.11 – Vazão por reserva e por contenção com chegadas poissonianas.

A alocação dinâmica das janelas de contenção produz uma melhora significativa de desempenho (menor tempo de transferência) em relação à alocação estática.
A importância do mecanismo de *piggybacking* para cargas superiores a 0.5 indica que este mecanismo deve também ser mantido.

A Figura 4.11 mostra que no caso adaptativo uma parcela significativa do tráfego é escoada pelas janelas de contenção.

Uma questão que aparece como conseqüência dos resultados desta seção é quanto ao tamanho do quadro, isto é, a determinação do número de janelas de reserva no quadro também deveria ser dependente da carga no sistema?

IV.2 – QUADRO VARIÁVEL COM TRÁFEGO POISSONIANO

No quadro variável, o número de janelas de reserva no quadro é igual ao número de terminais com reserva agendada.

As Figuras 4.12 a 4.18 apresentam os resultados obtidos através da simulação de evento discreto do mecanismo de alocação dinâmica de janelas com *piggybacking,* utilizando-se quadro de tamanho fixo com 1 janela (1) e 5 janelas (5) , e quadro de tamanho variável (V), para dez terminais.

Em todas as figuras estão apresentados os valores médios. O número de eventos das simulações (625.000 janelas) foi determinado para garantir intervalos de confiança menores que 3% do valor médio estimado com 95% de confiabilidade.

A Figura 4.12, número total de janelas no quadro (contenção e reserva) e número de janelas de contenção no quadro, mostra a adaptação do número de janelas de contenção e de reserva em função da carga de tráfego para quadro fixo (com 1 e 5 janelas) e para quadro de tamanho variável. A adaptação concomitante do número de janelas de contenção e de reserva permite que o tamanho médio do quadro adapte-se à carga de tráfego. Assim, para uma carga pequena, o tamanho do quadro é próximo de 1 e para carga de 0.7 o tamanho médio do quadro é 5. Observa-se que para cargas menores que 0.3, o algoritmo de quadro variável praticamente comporta-se como sendo de quadro fixo de tamanho 1. À medida que a carga cresce (acima de 0.5), o número de janelas cresce rapidamente.

A Figura 4.13, número médio de terminais por janela de contenção, mostra que esse número mantém-se bastante baixo (em média, menor que 2), ilustrando a eficiência dos algoritmos de adaptação.

A Figura 4.14, número médio de células no *buffer*, ilustra a eficiência do algoritmo adaptativo com quadro variável quanto à capacidade de manter reduzido o número de células no *buffer*. Note que para cargas baixas (menor que 0.5), o algoritmo adaptativo com quadro fixo de 1 janela é superior ao de 5 janelas.

As Figuras 4.15, tempo médio de espera no *buffer*, e 4.16, tempo médio de transferência, mostram o ganho de desempenho da utilização de quadro variável, em relação ao quadro fixo. Para cargas inferiores a 0.5, o quadro fixo com 1 janela é superior ao quadro fixo com 5 janelas.

As Figuras 4.17, ocupação das janelas de contenção, e 4.18, vazão por reserva e por contenção, confirmam o desempenho similar que têm o algoritmo de quadro variável e o de quadro fixo com 1 janela para cargas pequenas (inferiores a 0.4). Ilustram, também, que para cargas grandes é importante variar o número de janelas do quadro.

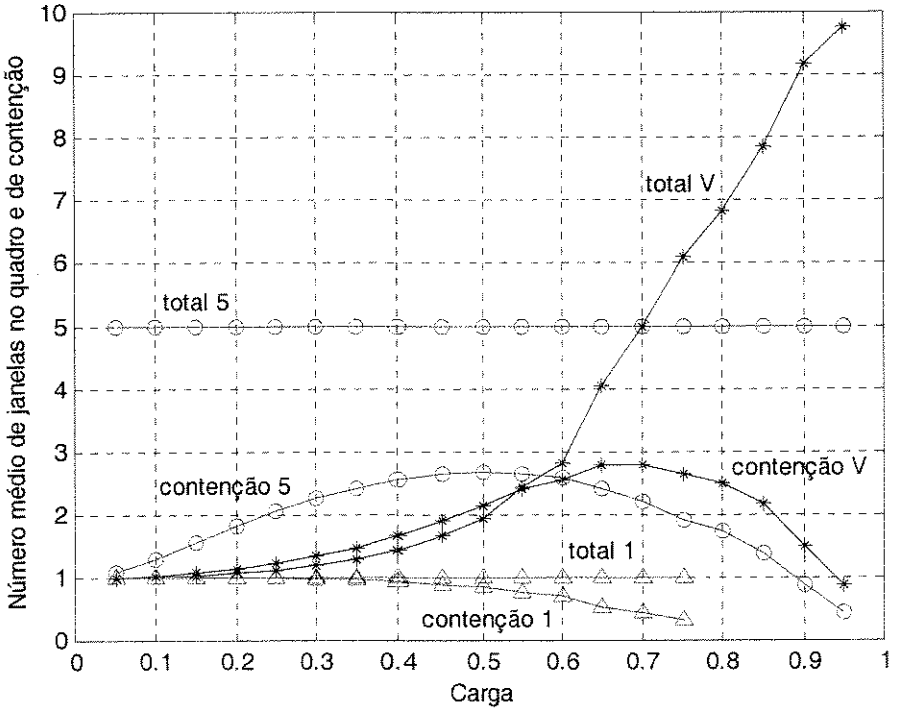


Figura 4.12 – Número médio de janelas no quadro e de contenção com chegadas poissonianas.

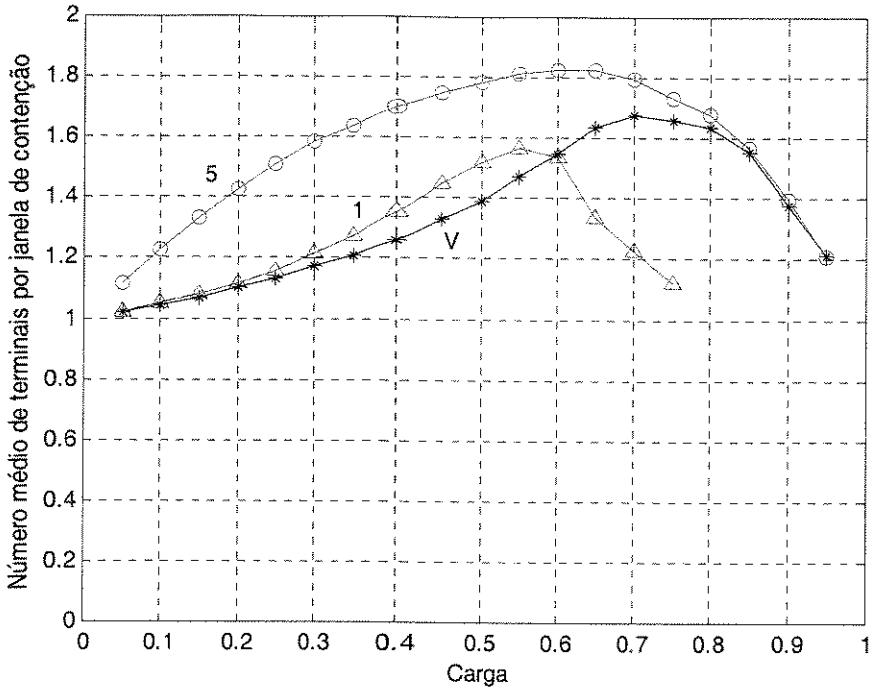


Figura 4.13 – Número médio de terminais por janela de contenção com chegadas poissonianas.

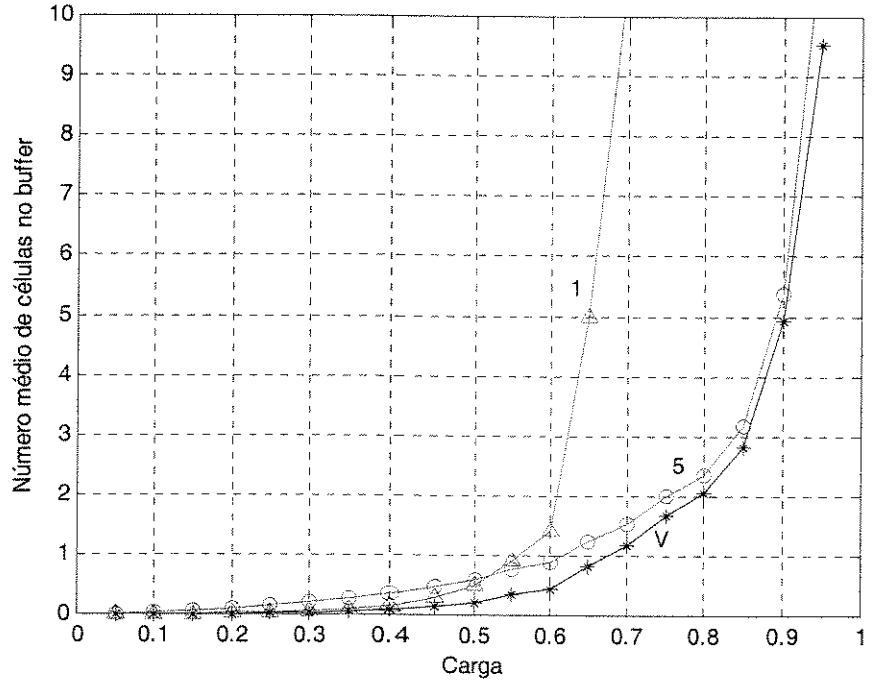


Figura 4.14 – Número médio de células no *buffer* com chegadas poissonianas.

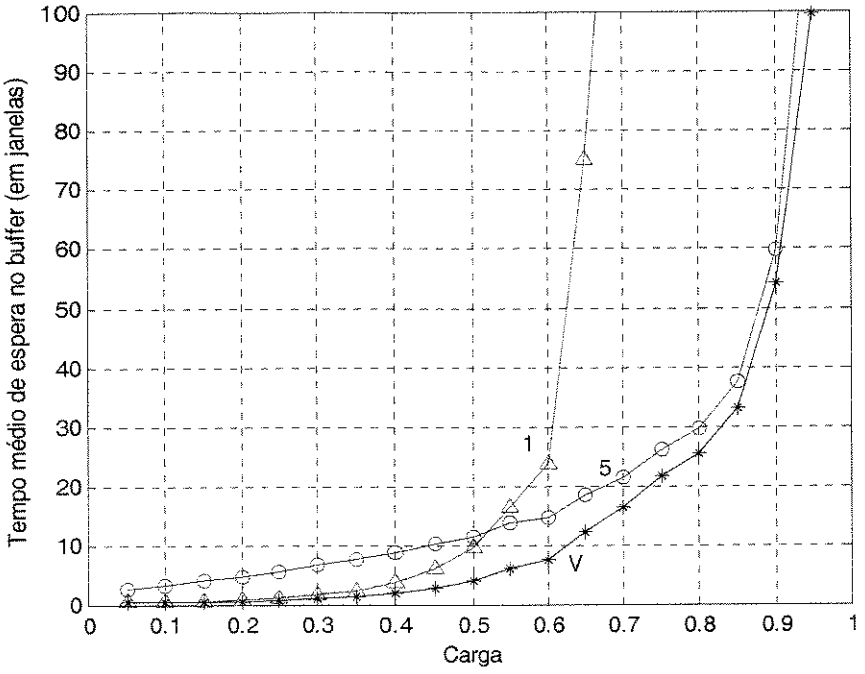


Figura 4.15 – Tempo médio de espera no *buffer* com chegadas poissonianas.

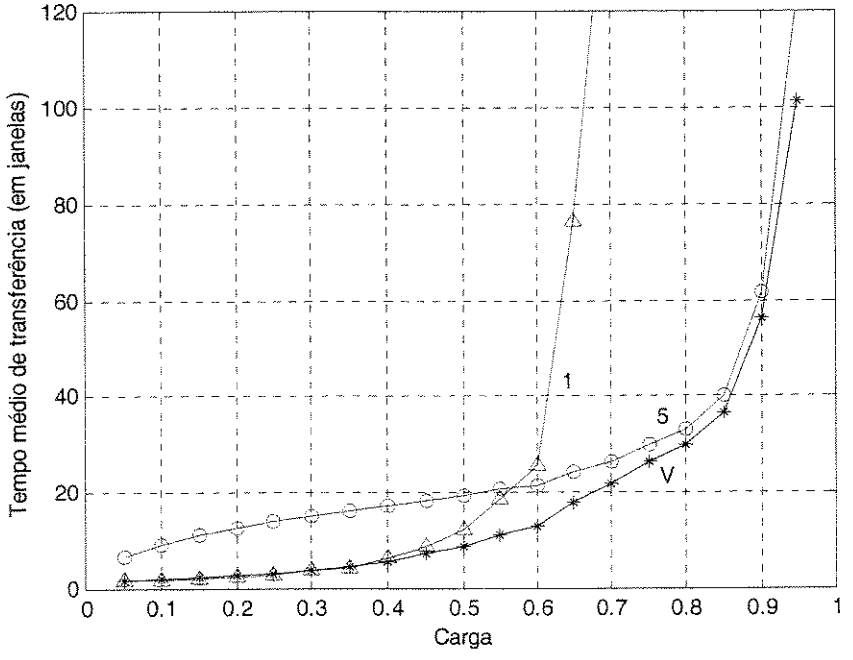


Figura 4.16 – Tempo médio de transferência com chegadas poissonianas.

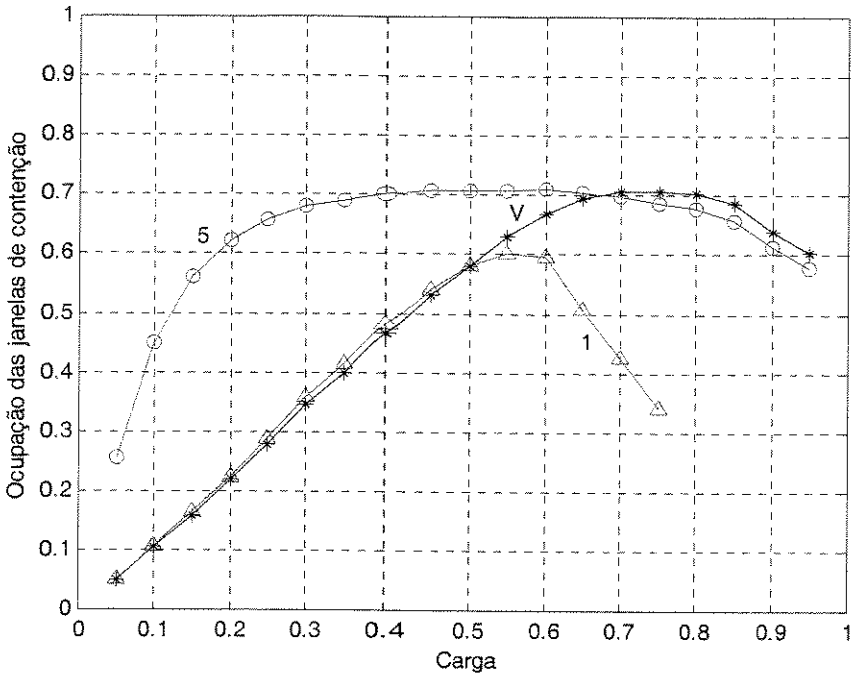


Figura 4.17 – Ocupação das janelas de contenção com chegadas poissonianas.

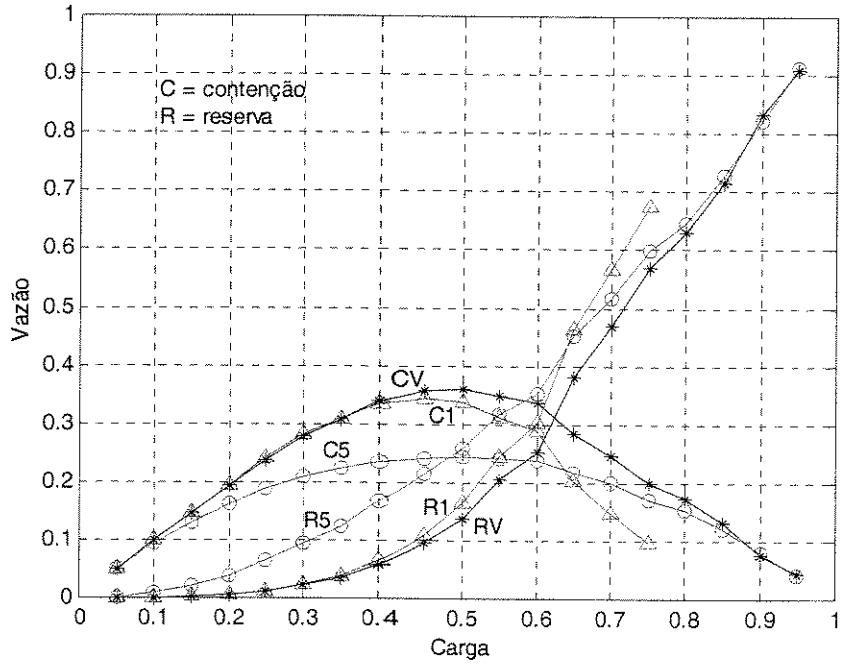


Figura 4.18 – Vazão por reserva e por contenção com chegadas poissonianas.

Tendo em vista os resultados apresentados, deve-se variar dinamicamente o número de janelas de contenção e de reserva do quadro. Portanto, só há quadro sem janelas de contenção se todos os terminais têm janelas agendadas ou se o mecanismo de desestímulo implica na não oferta de nova janela de contenção e ainda não há processo de contenção em andamento.

Testou-se o uso do número de janelas de reserva do quadro igual a 1, quando o número de células reservadas for maior que zero. Os resultados da simulação foram idênticos ao caso de número de janelas de reserva igual ao número de terminais com reserva agendada, até a carga de 35% e após essa carga o desempenho se degradou. Para a carga de 65%, o número de células no *buffer* cresceu anormalmente, aproximadamente 1.300 células.

Na próxima seção discute-se a influência do tamanho das janelas de contenção no desempenho do múltiplo acesso. Na resolução de contenção, a estação rádio-base comporta-se, para estimar o fim do processo, como um terminal que participa do processo de contenção, mas não participa da colisão, isto é, ela soma 2 na posição da pilha virtual em cada colisão.

IV.3 – MINI-JANELAS DE CONTENÇÃO

O número de eventos das simulações (625.000 janelas) foi determinado para garantir intervalos de confiança menores que 3% do valor médio estimado com 95% de confiabilidade.

A Figura 4.19 apresenta o tempo médio de transferência em função da carga para os protocolos usando mini-janelas de contenção (um quarto da janela de reserva) e janelas de contenção de mesmo tamanho da janela de reserva. O ganho de desempenho, no caso de mini-janelas, é significativo para todo o intervalo de carga considerado. A Figura 4.20, número de janelas de contenção e reserva no quadro e a Figura 4.21, vazão, permitem explicar qualitativamente esse ganho de desempenho.

A estratégia de manter o tamanho do quadro o menor possível (uma única janela de reserva por terminal em cada quadro) é melhor aproveitada com a diminuição das janelas de contenção. Parcela significativa do tráfego escoa via janela de contenção no caso sem mini-janelas, o que aumenta o tamanho médio do quadro resultando em um pior desempenho.

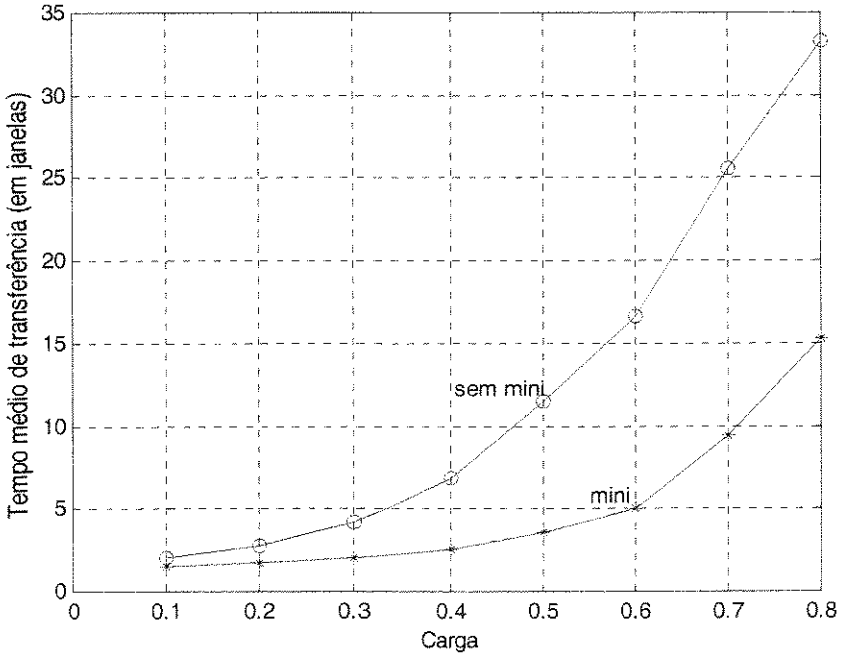


Figura 4.19 – Tempo médio de transferência com tráfego poissoniano uniforme.

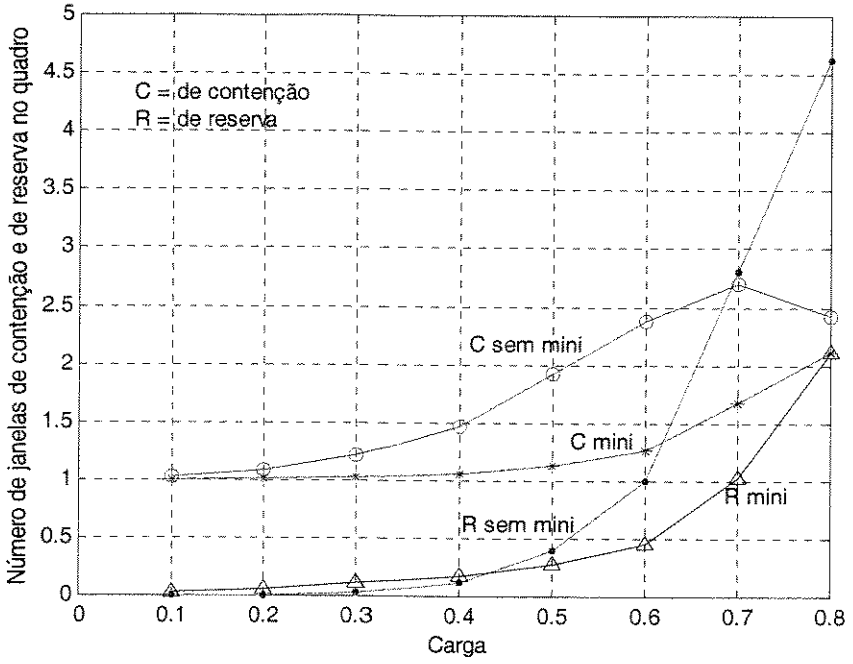


Figura 4.20 – Número médio de janelas de contenção e de reserva no quadro com tráfego poissoniano uniforme.

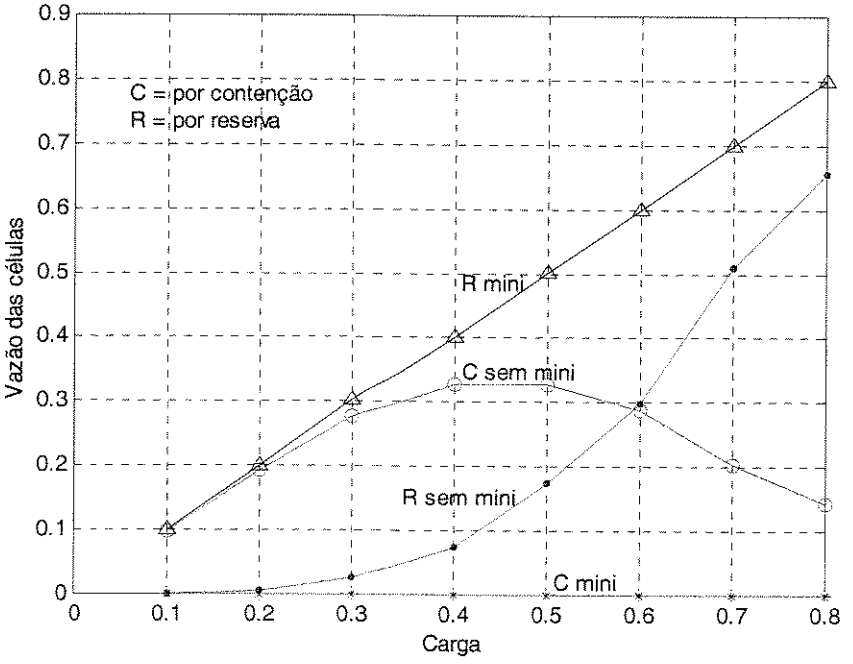


Figura 4.21 – Vazão das células com tráfego poissoniano uniforme.

IV.4 – TRÁFEGOS EM *BATCH*

A avaliação dos mecanismos do protocolo de múltiplo acesso é dependente das características de tráfego gerado pelos terminais. Modelos realistas de tráfego devem ser usados para

previsão de desempenho de um dado protocolo. Neste trabalho apenas tráfego poissoniano e *batch* poissoniano foram considerados com o intuito de avaliar comparativamente os mecanismos componentes dos protocolos. O modelo de *batch* adotado é baseado na referência [AGSF00] no qual as chegadas são poissonianas e os *batches* têm em média 11 células com tamanhos distribuídos aleatoriamente como mostrados na Tabela 4.1. Observa-se que o tráfego poissoniano estimula bastante o mecanismo de *piggybacking*, enquanto o de *batch* faz uso intenso do mecanismo de reserva. A hipótese de chegadas em *batch*, apesar de não realista (as células do *batch* chegam todas simultaneamente), serve para avaliar o mecanismo de agendamento.

Tamanho (células)	2	4	8	16	32
Probabilidade	0.5	0.1	0.1	0.05	0.25

Tabela 4.1 – Distribuição do tamanho das mensagens em *batch*.

O número de eventos das simulações (625.000 janelas) foi determinado para garantir intervalos de confiança menores que 3% do valor médio estimado com 95% de confiabilidade.

A Figura 4.22, tempo médio de transferência, mostra a degradação do desempenho com as chegadas em *batch* quando comparado com as chegadas poissonianas. O tempo necessário para esvaziar os *buffers*, vide Figura 4.23, justifica a diferença de desempenho.

A Figura 4.24, número de janelas de contenção e de reserva no quadro, ilustra o mecanismo de desestímulo usado na alocação de janelas de contenção, resultando, para o tráfego em *batch*, em um número médio de janelas de contenção por quadro menor que um. Para uma mesma carga de tráfego, no caso poissoniano, há mais terminais com demandas que no caso em *batch*, o que justifica um número maior de janelas de reserva no quadro.

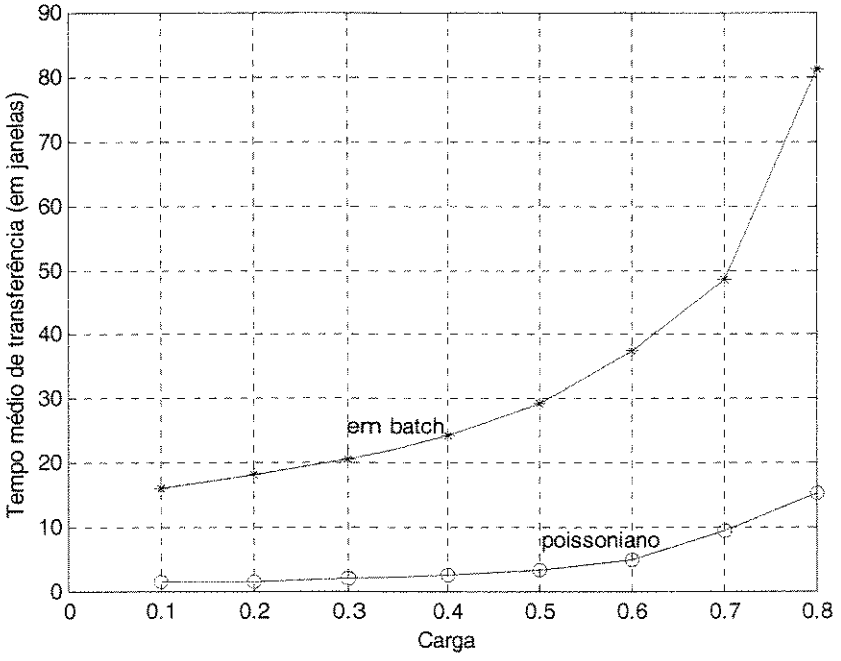


Figura 4.22 – Tempo médio de transferência com tráfegos poissoniano e em *batch* uniformes.

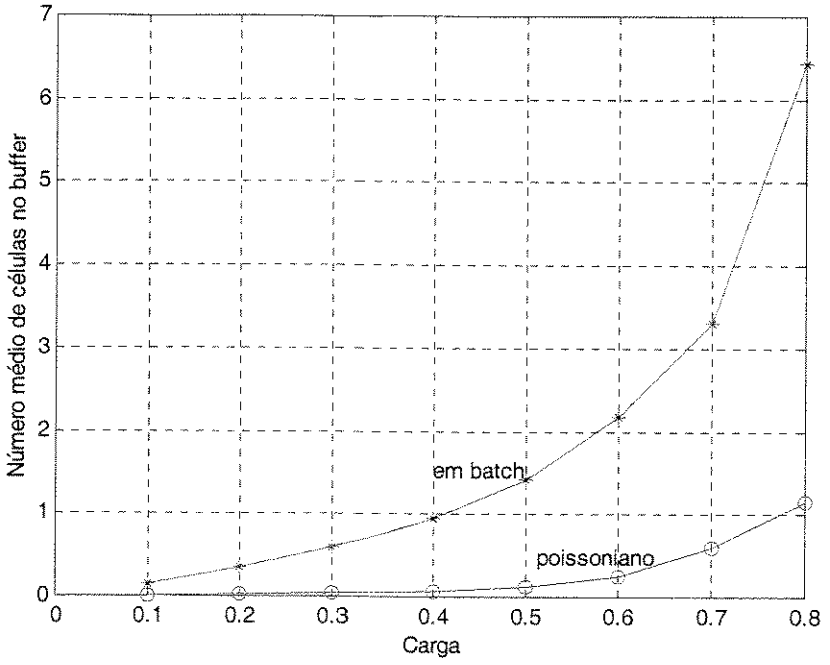


Figura 4.23 – Número médio de células no *buffer* com tráfegos poissoniano e em *batch* uniformes.

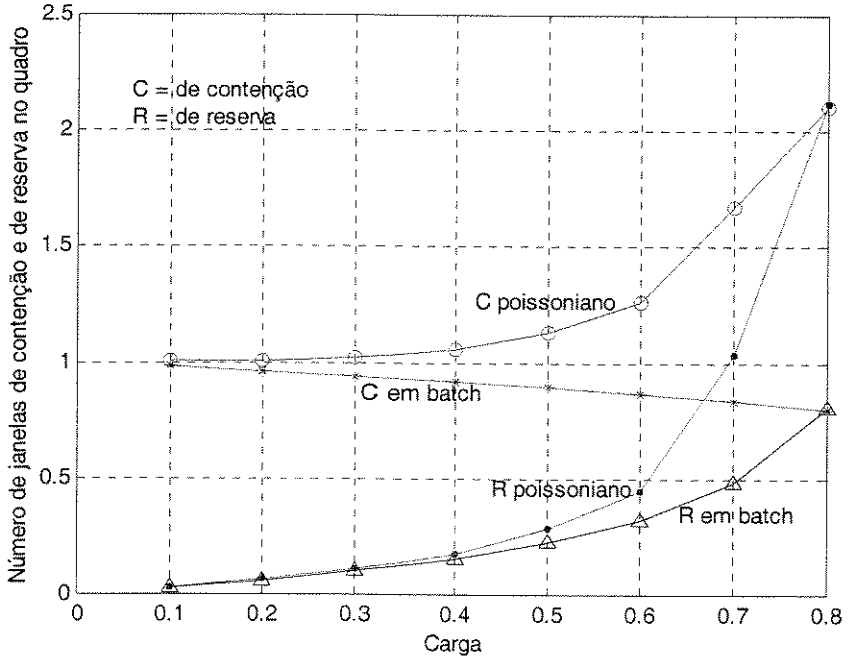


Figura 4.24 – Número médio de janelas de contenção e de reserva no quadro com tráfegos poissoniano e em *batch* uniformes.

Na próxima seção trata-se os tráfegos em tempo real (*rt*) e não em tempo real (*nrt*), considerando que as janelas de contenção são mini-janelas (um quarto da janela de reserva).

IV.5 – TRÁFEGOS EM TEMPO REAL

A Figura 4.25, tempo médio de transferência, mostra o ganho de desempenho dos terminais *rt* (prioritário) em relaçao aos terminais *nrt* para tráfego poissoniano uniforme. Cinco terminais têm tráfego em tempo real, portanto, tendo prioridade sobre os demais cinco. A prioridade ocorre em dois níveis: no acesso às mini-janelas de contenção (*binary e ternary*) e no agendamento das janelas de reserva (atende-se primeiro os terminais com tráfego em tempo real). Os terminais *rt* possuem um *buffer* de tamanho 10 e os terminais *nrt*, de tamanho 100. Para comparação, é mostrada também na figura, a curva de desempenho, quando todos os terminais são considerados sem prioridade.

O número de eventos das simulações (1.000.000 janelas) foi determinado para garantir intervalos de confiança menores que 3% do valor médio estimado com 95% de confiabilidade.

A Figura 4.26, número médio de células no *buffer*, confirma o comportamento do tempo médio de transferência. A Figura 4.27, número de janelas de contenção e de reserva no quadro, indica que a prioridade, praticamente, não afeta esses parâmetros.

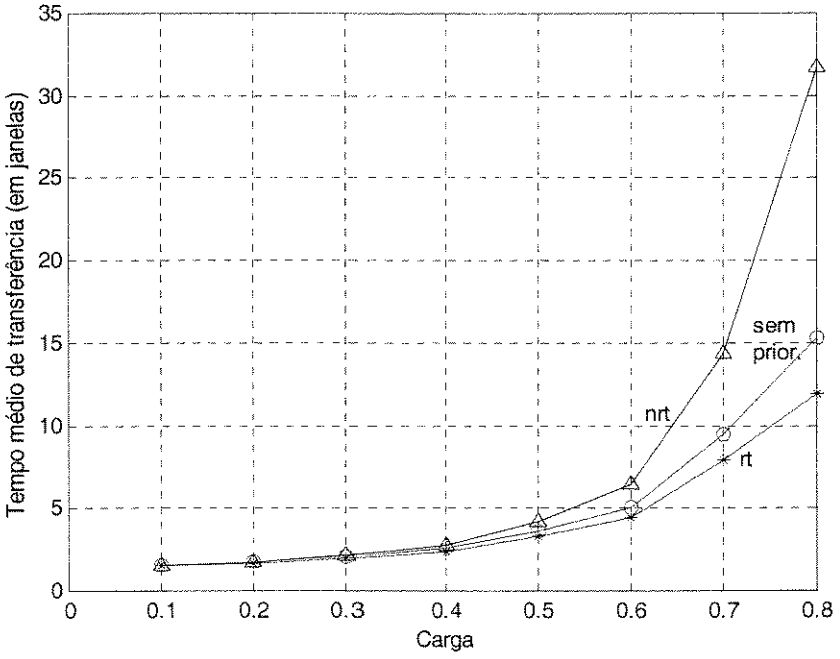


Figura 4.25 – Tempo médio de transferência para os terminais *rt* e *nrt* com tráfego poissoniano uniforme.

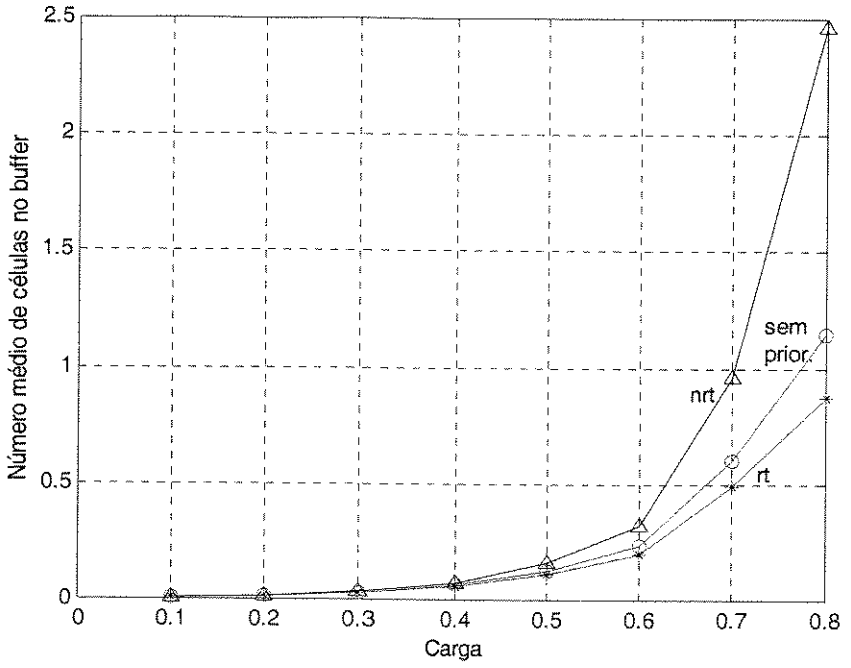


Figura 4.26 – Número médio de células no *buffer* para os terminais *rt* e *nrt* com tráfego poissoniano uniforme.

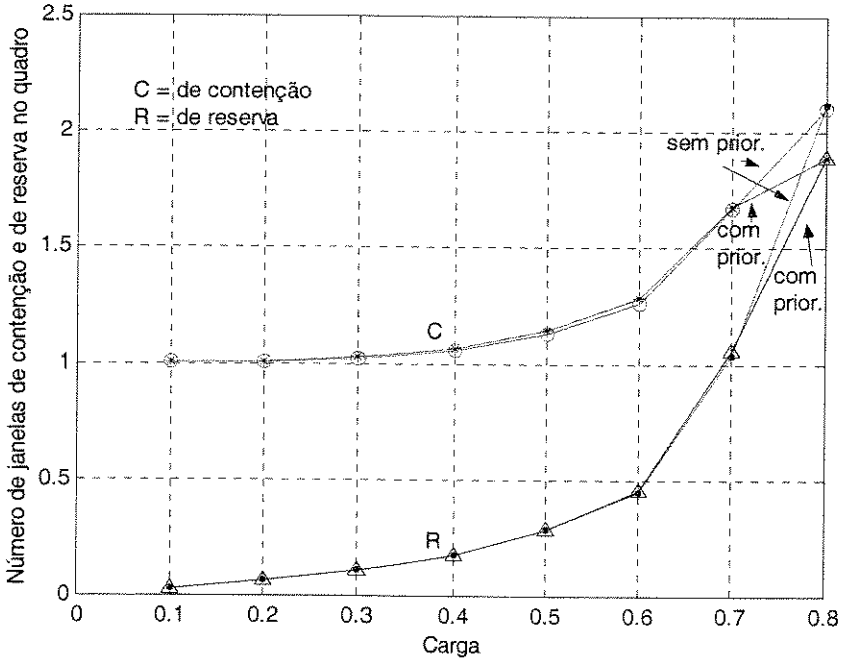


Figura 4.27 – Número médio de janelas de contenção e de reserva no quadro com tráfego poissoniano uniforme.

As Figuras 4.28, 4.29 e 4.30, mostram o desempenho do protocolo para tráfego em *batch*. Observa-se que o tempo médio de transferência e o número médio de células no *buffer* indicam um ganho de desempenho significativo para os terminais prioritários. Para efeito de comparação, a Figura 4.28 mostra também o tempo médio de transferência para o tráfego

poissoniano. A Figura 4.30 indica que o número de janelas no quadro não é sensível à existência de prioridade. Os *buffers* foram limitados em 100, o que acarretou perdas significativas para o tráfego não em tempo real para cargas superiores a 0.5, como mostra a Tabela 4.2.

Carga		0.5	0.6	0.7	0.8
Perda	***rt*** (10^{-5})	0	0	0	3.2
	nrt (10^{-4})	2.3	5.0	18	68

Tabela 4.2 – Taxa de perda de células.

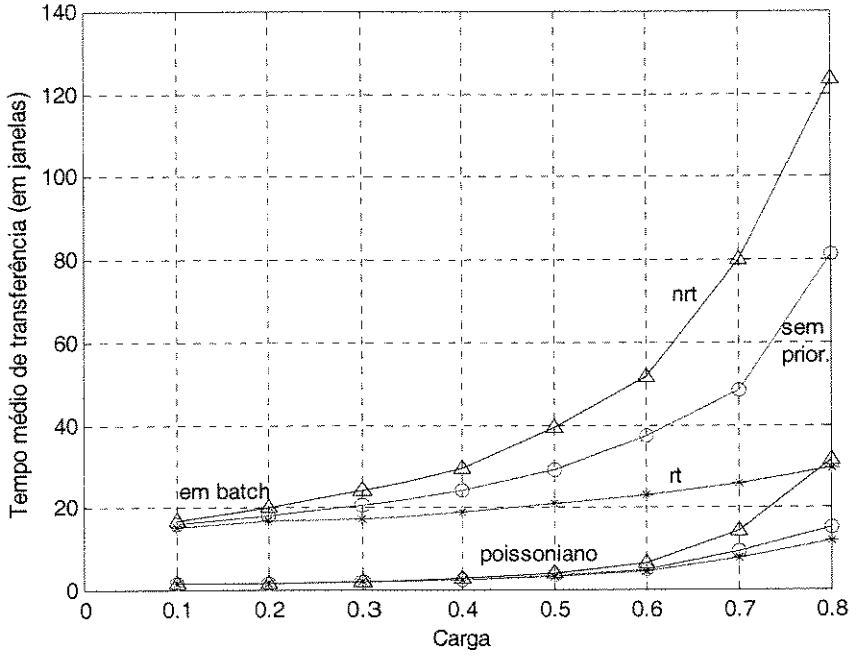


Figura 4.28 – Tempo médio de transferência para os terminais *rt* e *nrt* com tráfegos poissoniano e em *batch* uniformes.

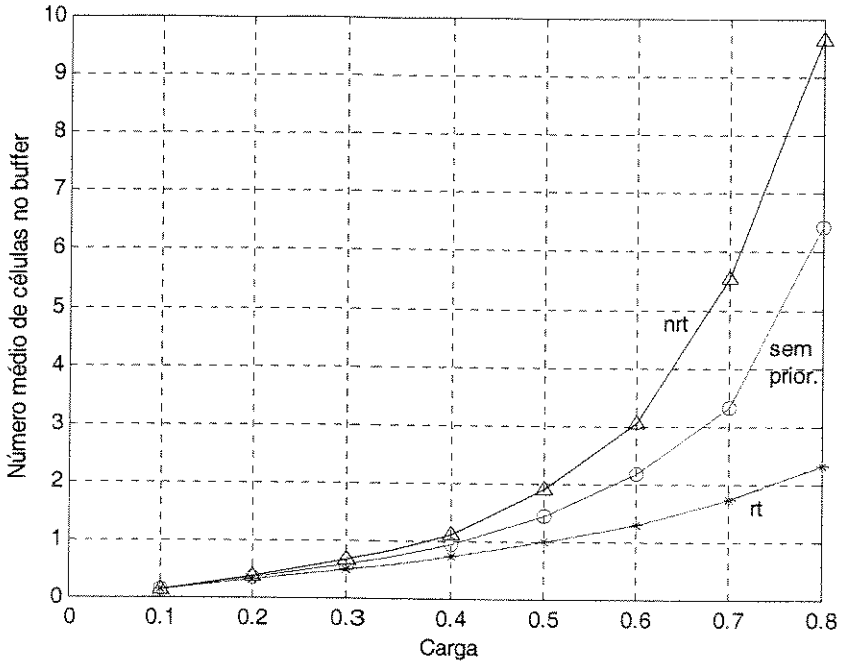


Figura 4.29 – Número médio de células no *buffer* para os terminais *rt* e *nrt* com tráfegos poissoniano e em *batch* uniformes.

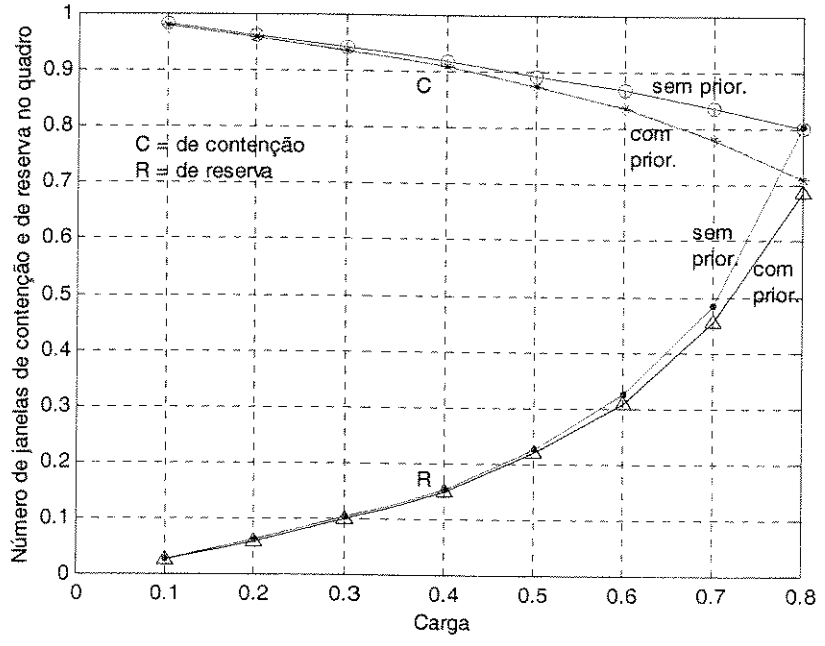


Figura 4.30 – Número médio de janelas de contenção e de reserva no quadro com tráfegos poissoniano e em *batch* uniformes.

IV.6 – TRÁFEGOS NÃO UNIFORMES

A avaliação do desempenho do protocolo sob o ponto de vista dos terminais pressupõe parâmetros de qualidade de serviço pré-definidos para cada uma das conexões. No

estabelecimento ou não da conexão, isto é, nos procedimentos de CAC (*Connection Admission Control*) é necessária a avaliação prévia do desempenho para a aceitação da conexão. Em geral, uma avaliação é possível através do conceito de banda efetiva e o mecanismo de agendamento deve ponderar cada conexão pela sua correspondente banda efetiva.

Nesse trabalho o enfoque foi analisar comparativamente os mecanismos do protocolo de múltiplo acesso e não avaliar a qualidade de serviço conferida aos terminais pelos procedimentos de CAC. Entretanto, algumas simulações com tráfego não uniforme entre os terminais foram realizadas para avaliar o comportamento dos terminais com tráfegos distintos.

A Tabela 4.3 mostra o percentual relativo de carga entre os dez terminais usados na simulação.

Terminal	Percentual
A	0.25
B	0.125
C	0.0625
D	0.03125
E	0.03125
F	0.25
G	0.125
H	0.0625
I	0.03125
J	0.03125

Tabela 4.3 – Percentual de carga nos terminais.

O número de eventos das simulações (1.000.000 janelas) foi determinado para garantir intervalos de confiança menores que 3% do valor médio estimado com 95% de confiabilidade.

A Figura 4.31, tempo médio de transferência para *batches* não uniformes, apresenta o comportamento dos terminais de maior tráfego (25% cada) comparado com os de menor tráfego (3.125% cada). Observa-se que os terminais de maior tráfego possuem um desempenho significativamente pior que os de menor tráfego decorrente do atendimento cíclico de uma célula por terminal. A figura mostra também que a média de desempenho, nesse cenário não uniforme é muito próxima do desempenho dos terminais quando todos têm o mesmo tráfego (uniforme) até a carga de 0.7. A Figura 4.32, número médio de células no *buffer,* apresenta comportamento similar ao tempo médio de transferência.

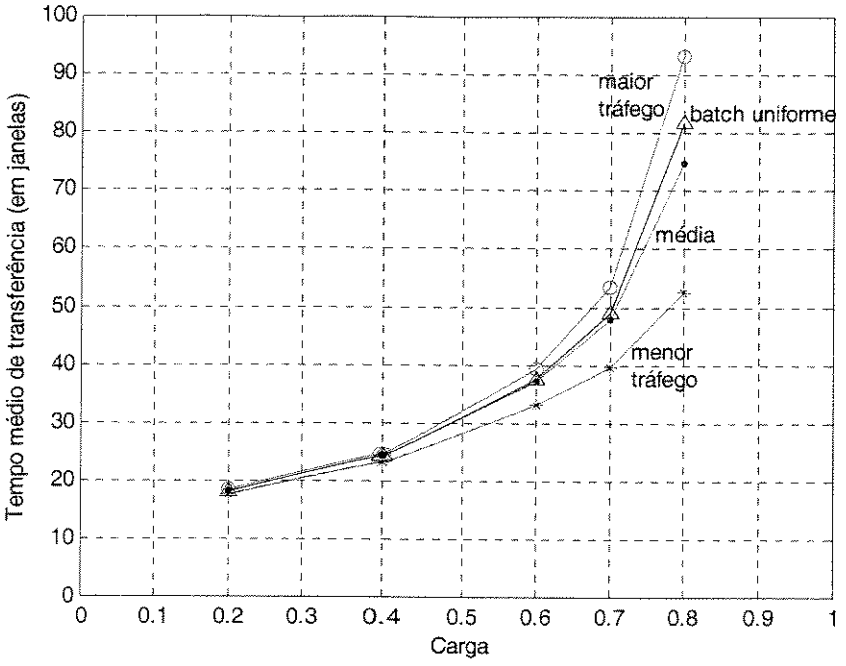


Figura 4.31 – Tempo médio de transferência com tráfegos em *batch* não uniforme.

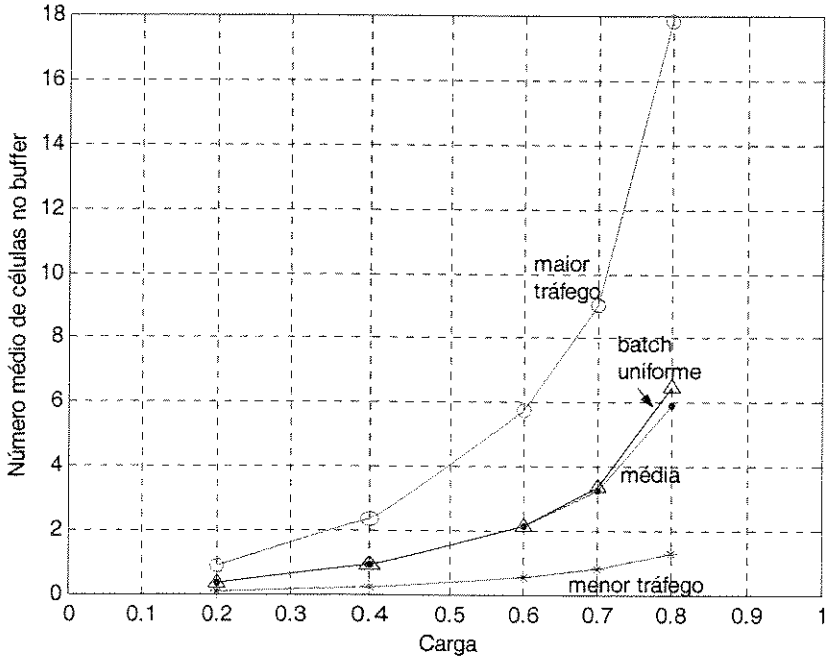


Figura 4.32 – Número médio de células no *buffer* com tráfegos em *batch* não uniforme.

O agendamento cíclico pode ser ponderado por uma série de parâmetros, como por exemplo, a banda efetiva de cada uma das conexões. Para ilustrar o impacto nos tempos de transferência de diferentes estratégias no atendimento cíclico, foi simulado um agendamento com o número de janelas de reserva igual ao número de células reservadas.

A Figura 4.33, tempo médio de transferência para *batches* não uniformes, apresenta o desempenho dos terminais com maior e menor tráfego, comparando-os para duas estratégias

distintas de atendimento cíclico: reserva por terminal ($r = ntr$); e reserva por célula ($r = ncr$). Observa-se uma melhoria de desempenho significativa para os terminais de maior tráfego quando altera-se a estratégia de atendimento de reserva por terminal para reserva por célula. Perda correspondente de desempenho ocorre para os terminais de menor tráfego. A Figura 4.34, número médio de células no *buffer*, mostra que as alterações nos tamanhos das filas são menos significativas.

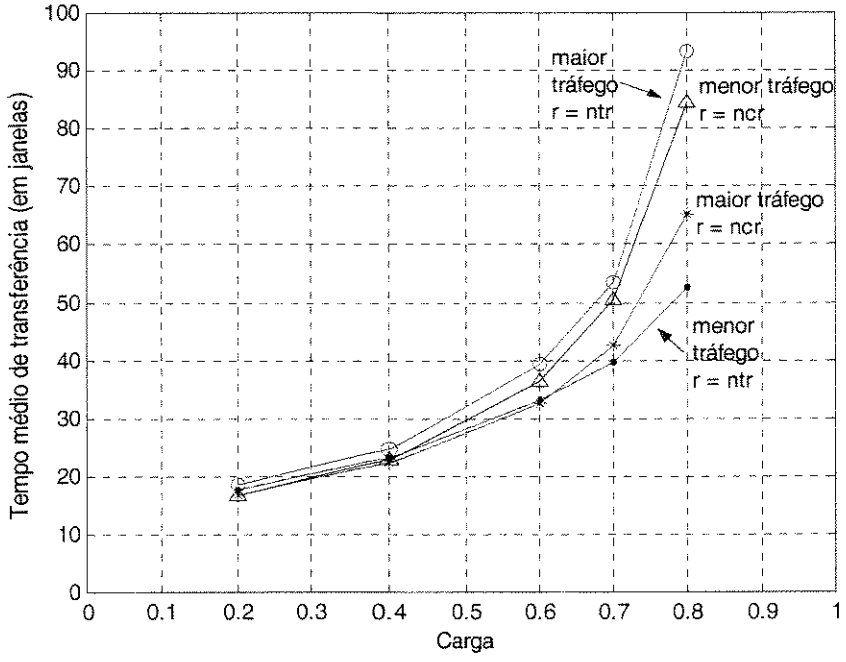


Figura 4.33 – Tempo médio de transferência com tráfegos em *batch* não uniforme.

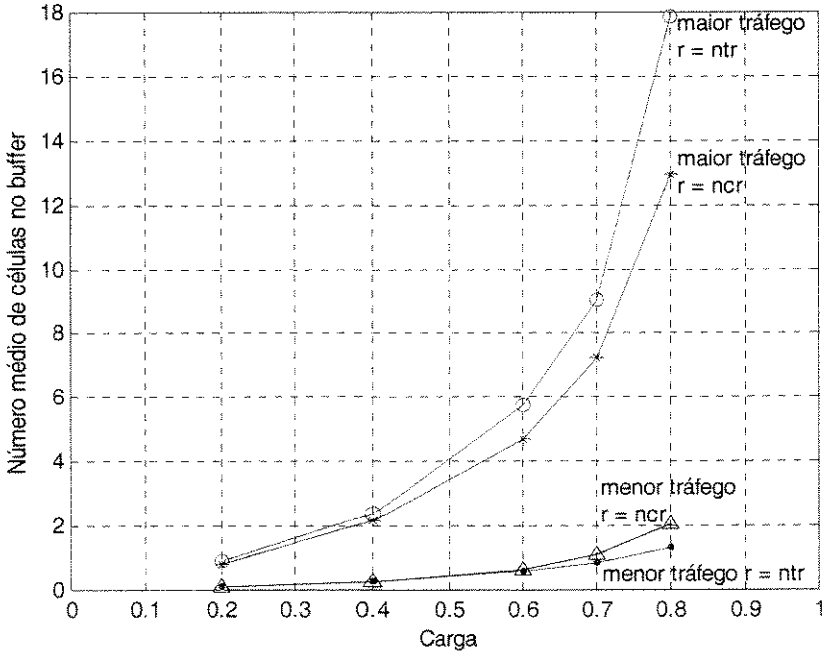


Figura 4.34 – Número médio de células no *buffer* com tráfegos em *batch* não uniforme.

CONCLUSÃO

Os algoritmos de resolução de contenção são empregados para resolver as colisões que ocorrem quando dois ou mais terminais enviam solicitações à estação rádio-base em uma mesma janela. Os algoritmos *n-ary tree* e *backoff* têm um desempenho similar na transmissão por contenção. Entretanto, na utilização da contenção como mecanismo de reserva, não se conhecia o desempenho comparativo dos algoritmos. Nesse trabalho, concluiu-se pela superioridade dos algoritmos *tree* sobre o algoritmo *backoff*. O desempenho do *binary tree* é similar ao *ternary tree* quanto à vazão, mas o *ternary tree* produz menor interferência co-canal. O algoritmo *binary tree* pode ser usado para dar prioridade ao tráfego em tempo real concomitantemente com a utilização do algoritmo *ternary tree* para tráfego não em tempo real.

O algoritmo de alocação dinâmica de janelas de contenção foi desenvolvido nesse trabalho de tese para propor novas janelas de contenção e para tratá-las simultaneamente no quadro. Inicialmente as múltiplas janelas de contenção foram tratadas utilizando-se quadro de tamanho fixo com agendamento cíclico das janelas de reserva, com e sem a utilização da informação de *piggybacking*. Verificou-se um ganho significativo de desempenho do algoritmo de adaptação em relação à alocação fixa do número de janelas de contenção. A importância do uso do mecanismo de *piggybacking*, para cargas altas, foi evidente. O algoritmo adaptativo manteve reduzido o número de terminais por janela de contenção (inferior a 2, em média). No caso adaptativo, uma parcela significativa do tráfego foi escoada pelas janelas de contenção, diferente do caso fixo, no qual a maior parte do tráfego foi escoada pelas janelas de reserva.

No agendamento cíclico considerado, o número de janelas de reserva do quadro é igual ao número de terminais com reserva agendada (uma única janela de reserva por terminal em cada quadro), produzindo um quadro variável com tamanho reduzido. O desempenho da utilização de quadro variável foi superior ao de quadro fixo. Testou-se também o uso do número de janelas de reserva do quadro igual a 1 quando o número de células reservadas fosse maior que zero. Os resultados da simulação foram idênticos aos do caso de número de janelas de reserva igual ao número de terminais com reserva agendada, até a carga de 35% e após essa carga, o desempenho se degradou.

A estratégia para manter o tamanho do quadro o menor possível foi melhor aproveitada com a diminuição das janelas de contenção. Parcela significativa do tráfego escoou via janela de contenção no caso sem mini-janelas, o que aumentou o tamanho médio do quadro resultando em um pior desempenho. O ganho de desempenho, no caso de mini-janelas, foi significativo para todo o intervalo de carga considerado.

A avaliação dos mecanismos do protocolo de múltiplo acesso é dependente das características do tráfego gerado pelos terminais. Observou-se que o tráfego poissoniano estimulou bastante o mecanismo de *piggybacking*, enquanto o de *batch* fez uso intenso do mecanismo de reserva. Observou-se, também, a degradação do desempenho com as chegadas em *batch* quando comparado com as chegadas poissonianas. O maior tempo para esvaziamento dos *buffers* justificou a diferença desse desempenho.

A prioridade de atendimento aos terminais com tráfego em tempo real (*rt*) ocorreu em dois níveis: no acesso às mini-janelas de contenção; e no agendamento das janelas de reserva (atendeu-se primeiro os terminais com tráfego em tempo real e uma célula de cada terminal). O ga-

nho de desempenho para os terminais *rt* foi significativo para cargas altas (maior que 0.5) e mais acentuado para tráfegos em *batch*.

A simulação para tráfego desigual entre os terminais mostrou que a estratégia de atendimento cíclico pode ter um impacto considerável no desempenho apresentado aos terminais. A análise do desempenho apresentada aos terminais, entretanto, depende de uma série de fatores sendo a qualidade do canal, um dos determinantes. Essa análise de desempenho, ao ser feita, deve necessariamente considerar simultaneamente os mecanismos de CAC e os correspondentes parâmetros de serviço. Este enfoque está além do escopo dessa tese e indica um possível caminho a seguir no aprofundamento das análises realizadas nesse trabalho.

REFERÊNCIAS BIBLIOGRÁFICAS

[AGSF00] – Ali, M.; Grover, R.; Stamatelos, G.; Falconer, D.D. – *Performance Evaluation of Candidate MAC Protocols for LMCS/LMDS Networks*. IEEE Journal on Selected Areas in Communications, July 2000.

[AMMPY99] – Akyildiz, I. F.; McNair, J.; Martorell, L. C.; Yesha, Y. – *Medium Access Control Protocols for Multimedia Traffic in Wireless Networks*, IEEE Network, July/August 1999.

[ARENA5.0] - Arena Pro 5.0 – Software de simulação de evento discreto - RockWell Software http://www.arenasimulation.com/

[BA01] – Brand, A.; Aghvami, H. – *Multiple Access Protocols for Mobile Communications: GPRS, UMTS and Beyond*, John Wiley & Sons, LTD., England, 2001.

[BDMMP96] – Bauchot, F.; Decrauzat, S.; Marmigère, G; Merakos, L; Passas N. – *MASCARA, a MAC Protocol for Wireless ATM*, Proc. ACTS Mobile Summit'96, Granada, Spain, Nov. 1996, pp. 17-22.

[BMN96] - Bisdikian, Chatschik; MacNeil, Bill; Norman, Rob – *msSTART: A Random Access Algorithm for the IEEE 802.14 HFC Network*, Computer Communications 19 (1996) 876-887.

[Ca79] – Capetanakis, John I. – *Tree Algorithms for Packet Broadcast Channels*, IEEE Trans. Inform. Theory, September 1979, Vol. IT-25, no. 5, pp. 505-515.

[CCL01] – Chen, Wen-Tsuen; Chen, Sheng-Hsien; Liu, Jen-Chu – *An Efficient QoS Guaranteed MAC Protocol in Wireless ATM Networks*, Proc. 15th International Conference on Information Networking, Japan, 2001, pp. 785-792.

[DAVIC 1.4] – DAVIC 1.4 Specifications Part 8: *Lower Layer Protocols and Physical Interfaces*, Digital Audio Visual Council, Geneva, 1998. http://www.davic.org

[DH99] – Dilson, D.A.; Haas, Z.J. – *A Dynamic Packet Reservation Multiple Access Sheme for Wireless ATM*, Mobile Netw. Appl., 1999, 4, pp. 87-99.

[DOCSIS99] – (MCNS/DOCSIS) *Multimedia Cable Network System's / Data Over Cable Service Interface Specifications*, (Radio Frequency Interface Specification SP-RFIv1.1-I07-010829). http://www.cablemodem.com/specifications.html

[FAG95] – Falconer, D.D.; Adachi F.; Gudmundson B. – *Time Division Multiple Access Methods for Wireless Personal Communications* – IEEE Communications Magazine, January 1995.

[GMPS97] - Golmie, Nada; Masson, Sandrine; Pieris, Gerard; Su, David H. – *A MAC Protocol for HFC Networks: Design Issues and Performance Evaluation*, Computer Communications 20 (1997) 1042-1050.

[Gu97] – Guedes, Leonardo G. R. – http://www.eee.ufg.br/~lguedes/cm/cm.htm - *Comunicações Móveis*

[GVGR89] – Goodman, D.J.; Valenzuela, R.A.; Gayliard, K.T.; Ramamurthi, B. - *Packet Reservation Multiple Access for Local Wireless Communications*, IEEE Trans. Commun., Vol.37, No.8, Aug. 1989, pp.885-90.

[KL01] – Kwok, Y.K.; Lau, V.K.N. – *Performance Evaluation of Multiple Access Control Schemes for Wireless Multimedia Services*, IEEE Proc.-Commun., Vol. 148, No.2, April 2001.

[Kl76] – Kleinrock, Leonard – *Queueing Systems, Volume 2: Computer Applications*. John Wiley & Sons, Inc., USA, 1976.

[KLE95] – Karol, Mark; Liu, Zhao; Eng, Kai – *Distributed-Queueing Request Update Multiple Access (DQRUMA) for Wireless Packet (ATM) Networks.* Proceedings of INFO COM'95, 1995, pp. 1224-1231.

[KW96] – Kim, J.G.; Widjaja, I. – *PRMA/DA: A New Media Access Control Protocol for Wireless ATM*, Proceedings of ICC'96, 1996, pp.240-244.

[Li98] – Lin, Ying-Dar – *On IEEE 802.14 Medium Access Control Protocol.* National Chiao Tung University, 1998, Taiwan. IEEE Communications Surveys & Tutorials.
http://www.comsoc.org/livepubs/surveys/public/4q98issue/lin.html

[LK00] – Lau, V.K.N; Kwok Y.K.: *On Channel Adaptive Multiple Access Control with Queued Transmission Requests for Wireless ATM,* Proceedings of 2000 IEEE Conference on High Performance Switching and Routing, Heidelberg, Germany, June 2000, pp. 473-481.

[Ma99] – Manning, Trevor – *Microwave Radio Transmission Design Guide*, Artech House, Inc., USA, 1999.

[MATLAB5.3] – Matlab 5.3 – Software de computação numérica – MathWorks
http://www.mathworks.com/

[MB88] - Merakos, Lazaros F.; Bisdikian, Chatschik – *Delay Analysis of the N-ary Stack Random-Access Algorithm*, IEEE Transactions on Information Theory, Vol. 34, No. 5, September 1988.

[MWAND98] – The Magic WAND – Wireless ATM MAC – Final Report – ACTS Project AC085, 1998.

[Pe95] - Petras, Dietmar – *Medium Access Control Protocol for Wireless, Transparent ATM Access*, Proceedings of IEEE Wireless Communication System Symposium, 1995.

[PMSBMD98] - Passas N.; Merakos, L.; Skyrianoglou D.; Bauchot f.; Marmigère G.; Decrauzat S. – *MAC Protocol and Traffic Scheduling for Wireless ATM Networks*, ACM Mobile Networks and Applications Journal, Wireless Lan's, 1998.

[PZK97] – Pahlavan K., Zahedi A., Krishnamurthy P. – IEEE Communications Magazine, Vol. 35, No. 11, November,1997.

[RBK97] - Reede, Ivan; Brandt, Matt; Karaoguz, Jeyhan - *Cable-TV Access Method and Physical Layer Specification*, Standards Project IEEE 802.14a Draft3 Revision1, 1997.

[Sa98] – Sari, Hilmet – *Some Design Issues in Local Multipoint Distribution Systems*. Alcatel Accesss Systems Division, 1998, France.

[Sk01] – Sklar, Bernard – *Digital Communications: Fundamentals and Applications*, Prentice Hall, USA, 2001.

[SM98] – Sriram K.; Magill, P. – *Enhanced Throughput Efficiency by Use of Dynamically Variable Request Minislots in MAC Protocols for HFC and Wireless Access Networks*. Journal of Telecommunications Systems Vol. 9, 1998.

[SMM97] – Sánchez J.; Martinez, R.; Marcellin, M.W. – *A Survey of MAC Protocols Proposed for Wireless ATM* – IEEE Network, November/December 1997.

[TGSF99] – Tarigari, M.; Grover, R.; Stamatelos, G.; Falconer, D.D. – *MAC Alternatives for LMCS/LMDS Networks*. IEEE International Conference, 1999, Canada, ICC'99.