

Estudo Comparativo de Algoritmos de CAC em Redes ATM

Niudomar Siqueira de Araújo Chaves
Shusaburo Motoyama

Departamento de Telemática - Faculdade de Engenharia Elétrica e de Computação
Universidade Estadual de Campinas - Unicamp
{niudomar, motoyama}@dt.fee.unicamp.br

Resumo

Neste trabalho são comparados os desempenhos de três algoritmos de CAC (Connection Admission Control) propostos na literatura para a implementação em redes ATM. A comparação é realizada através de simulação de eventos discretos. Dois dos algoritmos selecionados baseiam-se no conceito de capacidade equivalente, enquanto o terceiro é baseado em medições do tráfego corrente. Os resultados mostram que todos os algoritmos satisfazem a QoS especificada, mas superestimam as capacidades necessárias em algumas situações, fazendo com que haja uma baixa utilização dos recursos da rede.

1. INTRODUÇÃO

Uma rede ATM deve ser capaz de transportar tráfego multimídia de maneira eficiente e confiável. Diferentes tipos de tráfego devem ser atendidos simultaneamente com a qualidade de serviço exigida por cada um deles.

Esta qualidade deve ser negociada durante a fase de estabelecimento de conexão através de parâmetros descritores da fonte de tráfego, como taxa de pico de células, taxa média, comprimento máximo do surto. Os parâmetros de QoS (*Quality of Service*) que podem ser negociados na rede ATM são a razão de perda de células (CLR - *Cell Loss Ratio*), o atraso de transferência de célula (CTD - *Cell Transfer Delay*) e a variação de atraso de célula (CDV - *Cell Delay Variation*).

O Controle de Admissão de Conexões (CAC - *Connection Admission Control*) é uma função da rede ATM que avalia se uma conexão pode ser aceita ou deve ser recusada por falta de recursos disponíveis. Essa decisão é fundamental para garantir o adequado funcionamento da rede e a manutenção da qualidade de serviço de todas as conexões ativas.

Vários métodos de CAC têm sido propostos na literatura. A maioria destes algoritmos é baseada no conceito de banda efetiva como em [1] e [2]. A banda efetiva consiste na taxa de atendimento necessária para garantir a QoS de uma única conexão isoladamente.

Outros métodos sugerem o uso de CAC dinâmico. Nestas soluções, a decisão é baseada nos parâmetros descritores da fonte de tráfego e em medidas em tempo real do tráfego efetivamente sendo atendido pela rede [3], [5].

O objetivo deste trabalho é avaliar, através de simulação de eventos discretos, os algoritmos baseados em banda efetiva propostos por Guérin, Ahmadi e Naghshineh [1], por Kesidis, Walrand e Chang [2] e o algoritmo de CAC dinâmico proposto por Saito e Shiomoto [3].

2. ALGORITMOS

2.1 Algoritmo 1: Modelo de Fontes Combinadas

Este algoritmo [1] utiliza o conceito de capacidade equivalente. A capacidade equivalente, ou banda efetiva, consiste na largura de banda necessária para atender o tráfego de uma conexão mantendo a qualidade de serviço da nova conexão e daquelas em andamento dentro de valores especificados.

O cálculo do valor necessário de banda é baseado unicamente nas informações dos parâmetros descritores de tráfego da conexão solicitada e nos recursos da rede, como tamanho do *buffer*. A grande vantagem do conceito de banda efetiva é que o cálculo total da banda em uso consiste na adição das capacidades equivalentes de todas as conexões ativas. Assim, a rede operará de maneira similar ao modo de comutação de circuitos.

O conceito de capacidade equivalente é baseado na análise da taxa de decaimento assintótica da distribuição estacionária da ocupação de um *buffer*. A taxa de decaimento dependerá do tipo de fonte a que é submetido o *buffer*.

O cálculo da capacidade equivalente apresentado em [1] utiliza a fonte do tipo fluido Markoviano de dois estados, ou *on-off*. Este tipo de fonte pode ser caracterizada por sua taxa de pico R , a fração do tempo em que está ativa r (carga) e o comprimento médio dos surtos gerados b .

O modelo de nó de comutação utilizado em [1] é baseado inicialmente em uma única fonte *on-off* e um *buffer* finito, mostrado na Fig. 2.1.

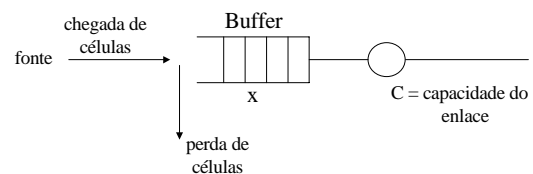


Figura 2.1: Multiplexador ATM

Seja $P_i(t, x)$ a probabilidade de que a fonte esteja no estado i e o conteúdo do *buffer* seja x no instante t .

Se $i = 1 \Rightarrow$ ativo;

Se $i = 0 \Rightarrow$ silêncio.

Utilizando o modelo de aproximação por fluxo de fluido, para análise de ocupação do *buffer*, as probabilidades $P_i(t, x)$ podem ser equacionadas e resolvidas. Supondo o caso de uma única fonte conectada e invertendo a expressão da distribuição da ocupação de um *buffer* finito de tamanho K e submetido ao tráfego de uma única fonte, a probabilidade de perda de células ε é dada por,

$$\varepsilon = \beta \cdot \exp\left(-\frac{K(c-rR)}{b(1-r)(R-c)c}\right), \quad (2.1)$$

onde

$$\beta = \frac{(c-rR) + \varepsilon r(R-c)}{(1-r)c}, \quad (2.2)$$

e c é a capacidade equivalente da fonte em questão. O parâmetro ε é também considerado a taxa de perda de células, CLR.

Mesmo no caso simples de uma única fonte, não há um resultado explícito para a capacidade equivalente e a equação precisa ser resolvida numericamente. A resolução numérica pode levar a tempos de resposta inadequados ao funcionamento da rede. É necessário utilizar alguma simplificação e β é assumido ser igual a 1. Esta aproximação resulta num limitante superior para c .

Fazendo $\alpha = \ln(1/\varepsilon)$, temos

$$\alpha = \frac{K(c-rR)}{b(1-r)(R-c)c}. \quad (2.3)$$

O resultado final para a capacidade é

$$c = \frac{a - K + \sqrt{(a-K)^2 + 4Kar}}{2a} R, \quad (2.4)$$

onde $a = \alpha b(1-r)R$.

Para o caso de N fontes, a banda efetiva é calculada por

$$C_E = \sum_{i=1}^N \hat{c}_i \quad (2.5)$$

onde \hat{c}_i é a banda efetiva da fonte i .

Tem sido observado que a taxa estacionária de *bits* do tráfego agregado de várias fontes se aproxima de uma distribuição normal com média m e variância σ^2 [5]. Como na Eq. 2.5 as fontes não foram consideradas agregadas, a banda C_E pode estar superestimada. Assim, é considerada também uma fonte agregada gaussiana. O valor da capacidade equivalente para a fonte agregada pode ser obtido por aproximações para a inversa da distribuição normal de probabilidades.

Para a fonte gaussiana, a banda efetiva é dada por [1]

$$C_G = m + \alpha' \sigma, \quad (2.6)$$

onde

$$\alpha' = \sqrt{-2 \ln(\varepsilon) - \ln(2\pi)},$$

$m = \sum_{i=1}^N m_i$, é a taxa média total de *bits*, e

$\sigma^2 = \sum_{i=1}^N \sigma_i^2$; σ_i^2 é a variância da taxa de bits da i -ésima fonte.

A banda efetiva final será

$$C_F = \min\{C_E, C_G\}. \quad (2.7)$$

Se $C_F \leq C_T$, onde C_T é a capacidade total do enlace, aceita-se a conexão; caso contrário, a chamada é rejeitada.

2.2 Algoritmo 2: Modelo de Fontes Gerais

Kesidis, Walrand e Chang [2] mostraram a existência de banda efetiva para uma classe mais geral de fontes. O

modelo utilizado é baseado na suposição de que a probabilidade de ocupação de um *buffer* infinito, em estado estacionário, ultrapassar um limite K obedece a um decaimento exponencial:

$$P\{X > K\} \leq e^{-K\delta},$$

onde δ é uma constante positiva chamada taxa de decaimento assintótica.

Além disso, o modelo é baseado na existência da função

$$h(\delta) = \lim_{t \rightarrow \infty} \frac{\ln E\{\exp(A(t)\delta)\}}{t} \quad (2.8)$$

onde $A(t)$ é o número de células geradas no intervalo de tempo $[0, t]$ e δ é um valor real.

A banda efetiva é dada por [2]

$$c(\delta) = \frac{h(\delta)}{\delta}. \quad (2.9)$$

O cálculo de $h(\delta)$ é, em geral, bastante complexo, e dificulta a sua utilização em CAC. Entretanto, resultados são apresentados em [2] para fontes de taxa constante, Poisson, fontes de Markov discretas no tempo, fluidos de Markov e processos poissonianos modulados por Markov. Para a fonte do tipo fluido markoviano *on-off*, as expressões acima podem ser calculadas e a banda efetiva é dada por [2]

$$c = \alpha + \sqrt{\alpha^2 + \beta^2} \quad (2.10)$$

onde

$$\alpha = \frac{1}{2} \left(R - \frac{1}{\delta b} - \frac{1}{\delta T_{off}} \right), \text{ e} \quad (2.11)$$

$$\beta = \frac{R}{\delta T_{off}}. \quad (2.12)$$

T_{off} é o tempo médio dos períodos de silêncio da fonte,

$$T_{off} = \frac{b(1-r)}{r}. \quad (2.13)$$

O valor de δ é dado por

$$\delta = \frac{\ln(1/\varepsilon)}{K}, \quad (2.14)$$

onde ε é a taxa de perda de células (CLR) especificada no contrato de tráfego.

2.3 Algoritmo 3: Alocação Dinâmica

Saito e Shiomoto [3] propuseram um esquema de alocação dinâmica. Além dos parâmetros fornecidos pelo usuário, a rede baseia-se no tráfego efetivamente medido para decidir se pode ou não atender a uma nova conexão. A proposta tem a vantagem de não ser baseada em nenhum modelo específico de fonte de tráfego. O método é baseado na estimação da distribuição de probabilidade das fontes agregadas, através da contagem de células que chegam num intervalo fixo s .

O modelo consiste de n conexões sendo atendidas por um enlace de capacidade C' *bits/s*, um buffer de tamanho K , e $p_i^*(j)$ é a distribuição de probabilidade da chegada de j células da conexão i durante o intervalo de medição s . Assumindo $C = C's/L \leq K+1$, onde L é o comprimento de

uma célula ATM em *bits*, o limite superior para a probabilidade de perda de células, ε , é dado por

$$\varepsilon \leq \frac{\sum_{k=0}^{\infty} [k-C]^+ p_1^* \dots p_n^*(k)}{\sum_{k=0}^{\infty} k p_1^* \dots p_n^*(k)} \quad (2.15)$$

onde

$$[x]^+ = \begin{cases} x, & \text{se } x > 0 \\ 0, & \text{de outra forma} \end{cases}$$

e $*$ representa a operação de convolução.

Um usuário, ao solicitar conexão, deve fornecer a taxa de pico de bits ξ e a taxa média de bits ψ . Para incluir a chamada solicitada no cálculo de ε para a tomada de decisão é definido o vetor θ que representa o pior caso da distribuição de chegada de células da conexão solicitada em relação aos períodos de surto e silêncio [3]. Dessa forma,

$$\theta_{n+1}(j) = \begin{cases} a_{n+1}/R_{n+1}, & j = R_{n+1} \\ 1 - (a_{n+1}/R_{n+1}), & j = 0 \\ 0, & \text{de outra forma} \end{cases} \quad (2.16)$$

onde $R_{n+1} = \lceil \xi_{n+1}s/L \rceil$ e $a_{n+1} = \psi_{n+1}s/L$. $\lceil x \rceil$ representa o menor inteiro maior ou igual a x . R_{n+1} e a_{n+1} representam o número máximo e o número médio de células que chegam, durante o intervalo de medição s , da chamada $n+1$. Acrescentando a informação da conexão requisitada temos

$$\varepsilon \leq \frac{\sum_{k=0}^{\infty} [k-C]^+ p_1^* \dots p_n^* \theta_{n+1}(k)}{\sum_{k=0}^{\infty} k p_1^* \dots p_n^*(k) + a_{n+1}} \quad (2.17)$$

Através de medição é possível obter uma estimativa para $p_1^* \dots p_n^*(k)$. Usando a estimativa $\hat{\mathbf{p}} = (\hat{p}(0), \hat{p}(1), \dots)$ pode-se obter o valor estimado da probabilidade de perda de células com a aceitação da nova conexão, $\hat{\varepsilon}_1$, que é

$$\hat{\varepsilon}_1 \leq \frac{\sum_{k=0}^{\infty} [k-C]^+ \hat{p} * \theta_{n+1}(k)}{\sum_{k=0}^{\infty} k \hat{p}(k) + a_{n+1}} \quad (2.18)$$

Se os erros na estimativa $\hat{\mathbf{p}}$ forem suficientemente pequenos, o valor $\hat{\varepsilon}_1$ pode ser usado como o limite superior da probabilidade de perda. Se o valor encontrado for inferior ao CLR desejado, a conexão é aceita, caso contrário, recusada.

O método precisa usar um processo de contagem do número de células que chegam no intervalo de medição s para estimar a distribuição de probabilidade do número de células que chegam nesse tempo. Um conjunto de N intervalos de medição é chamado de período de renovação. A distribuição de frequência medida nos N intervalos do t -ésimo período de renovação é identificada por $\{q(k; t), k = 0, 1, \dots\}$ e o valor médio do número de chegada de células é

$$b(t) = \sum_{k=0}^{\infty} k q(k; t) \quad (2.19)$$

Sempre que um período de renovação é completado as estimativas $\hat{\mathbf{p}}$ e \hat{a} devem ser atualizadas. A forma de atualização depende da ocorrência de aceitação de novas conexões ou encerramento de conexões que já estavam ativas.

Se o número de conexões ativas durante um período de renovação não é alterado, as estimativas são atualizadas da seguinte forma:

$$\hat{\mathbf{p}}(t+1) = \alpha \mathbf{q}(t) + (1-\alpha) \hat{\mathbf{p}}(t) \quad (2.20)$$

$$\hat{a}(t+1) = \alpha b(t) + (1-\alpha) \hat{a}(t) \quad (2.21)$$

onde α é um valor entre 0 e 1, que tem a função de regular o peso entre o tráfego medido no último período de medição e o resultado acumulado durante os períodos anteriores.

Quando uma nova conexão é estabelecida no t -ésimo período de renovação, deve-se fazer uso dos descritores de tráfego fornecidos pelo usuário que solicita a conexão e, por isso, é incluído no cálculo o vetor θ .

$$\hat{p}(k; t+1) = \hat{p}(\cdot; t) * \theta_{n+1}(k), \quad (k=0, 1, \dots) \quad (2.22)$$

$$\hat{a}(t+1) = \hat{a}(t) + a_{n+1} \quad (2.23)$$

Quando uma conexão é encerrada é necessário subtrair seu tráfego dos valores estimados e a atualização dos parâmetros fica da seguinte forma:

$$\hat{p}(0; t+1) = \sum_{k=0}^{a'} \hat{p}(k; t) \quad (2.24)$$

$$\hat{p}(k; t+1) = \hat{p}(k+a'; t) \quad (k \geq 1) \quad (2.25)$$

$$\hat{a}(t+1) = \hat{a}(t) - a' - \sum_{k=0}^{a'} (k-a') \hat{p}(k; t) \quad (2.26)$$

$$\text{onde } a' = \left\lceil \hat{a}(t) - \sum_{i=1}^n a_i \right\rceil.$$

O vetor θ é calculado utilizando a taxa média e a taxa de pico da fonte que solicita a conexão. Esses valores de taxa média e de pico são referentes ao tempo do intervalo de medição s , ou seja, são calculados através do produto da taxa (em células/segundo) pelo intervalo s (em segundos). O parâmetro s é tomado a partir de um número inteiro de *slots* de tempo do enlace. Entenda-se por *slot* o tempo necessário para a transmissão de uma célula ATM pelo enlace. A largura de s (em *slots*) é dividida pela velocidade do enlace (em células/segundo), para se obter a largura do intervalo de medição (em segundos).

3. SIMULAÇÃO

O modelo de nó utilizado é um multiplexador ATM submetido ao tráfego de um conjunto de fontes idênticas. O multiplexador é modelado por um *buffer* de tamanho K , com disciplina de atendimento do tipo FIFO (*First In-First Out*) e um único servidor com taxa de atendimento constante igual à capacidade do enlace. As fontes são do tipo fluido markoviano de dois estados (*on-off*) e são caracterizadas pela taxa de pico R , comprimento médio dos surtos gerados b e carga r .

Para obter resultados mais próximos do que é esperado encontrar em sistemas reais, utilizou-se um enlace com taxa de transmissão de 155,52 *Mbps*. A taxa de perda de células, ϵ , foi fixada em 10^{-3} . O tempo de tráfego simulado foi de 900 segundos, o que representa um fluxo de células significativo num multiplexador ATM operando à taxa escolhida. Há um período inicial da simulação (*warm-up*) para permitir que se atinja o regime estacionário, só após o qual o tempo de 900 segundos é contado e as estatísticas passam a ser obtidas.

As simulações estão divididas em três grupos para analisar o comportamento dos algoritmos em resposta às variações no tamanho do *buffer*, no comprimento médio do surto e na carga das fontes.

Para comparar a eficiência dos esquemas de CAC, foi efetuado um grupo adicional de simulações para estimar o limite superior de atendimento do multiplexador. Nessas simulações, através de ajustes sucessivos, o número de conexões foi aumentado ou diminuído até que a taxa de perda de células obtida estivesse próxima, porém inferior, ao valor desejado. Os resultados assim obtidos representam os valores máximos de conexões que o multiplexador pode aceitar e a comparação com esses valores indica se as três propostas analisadas utilizam a faixa eficientemente ou desperdiçam largura de banda de transmissão do enlace.

O método de alocação dinâmica necessita da especificação de três parâmetros para seu funcionamento: a largura do intervalo de medição, s , a largura do período de renovação e α , utilizado na previsão exponencial. O comprimento do período de renovação é dado em número de intervalos de medição, que, por sua vez, é dado em número de slots de tempo equivalentes a uma célula no enlace. O período de renovação é sempre especificado com largura igual ao tamanho do *buffer* utilizado. Em todas as simulações a largura do período de renovação foi especificado em 10.000 e o parâmetro α em 0,2.

Como a estimação do tráfego é um processo demorado, optou-se por distribuir as chegadas das chamadas ao longo do período de *warm-up* de tal forma que apenas uma conexão fosse solicitada durante um período de renovação. Dessa forma, o efeito da chegada de muitas chamadas num único período de renovação não foi avaliado nas simulações. Isso pode ser um problema para o método devido à demora que este apresenta para perceber mudanças nas características do tráfego.

A desconexão e conseqüente atualização do vetor de estimação do tráfego não foi implementada. Esse é um processo também bastante lento e a implementação de todas as etapas de uma conexão com duração suficiente para estimar a taxa de perda de células com este algoritmo provocaria tempos de simulação extremamente longos. Outro motivo é que o interesse maior é avaliar a taxa de perda de células no regime estacionário e a retirada de conexões exigiria a implementação de um regime contínuo de chegada de novas conexões, possivelmente numa distribuição poissoniana. A simulação

ficaria muito complexa e os tempos de execução excessivamente longos.

4. RESULTADOS

O primeiro conjunto de simulações mostra o efeito da variação do comprimento do *buffer* na resposta dos algoritmos analisados. Utiliza fontes com taxa de pico de 8 *Mbps*, comprimento médio de surto de 100 células e carga de 0,2. Como as características do tráfego gerado por uma fonte multimídia ainda não são bem conhecidas, foram escolhidas, aleatoriamente, cargas que possibilitam maior ganho estatístico pelo multiplexador e, por isso, são as de maior interesse na determinação eficiente de sua banda efetiva.

Pelas curvas da Fig. 4.1 pode-se observar que o número de conexões aceitas pelos três algoritmos difere fundamentalmente na região em que o *buffer* é pequeno ($K < 1.000$ células). Os dois esquemas de capacidade equivalente apresentam o mesmo resultado para *buffers* de 700 ou mais células. O algoritmo de Guérin, Ahmadi e Naghshineh [1] tem melhor aproveitamento do enlace. Além disso, pode-se verificar que aumentos no tamanho do *buffer* além de 5.000 células têm efeito muito pequeno. Contudo, é importante ressaltar que este limite varia de acordo com o comprimento do surto; para valores maiores de surto a saturação acontecerá com valores maiores do *buffer*. O tamanho do *buffer* apresenta uma importância menor no método de Saito et al., fazendo com que este tenha desempenho superior aos demais para valores de *buffer* pequeno e desempenho inferior para valores maiores.

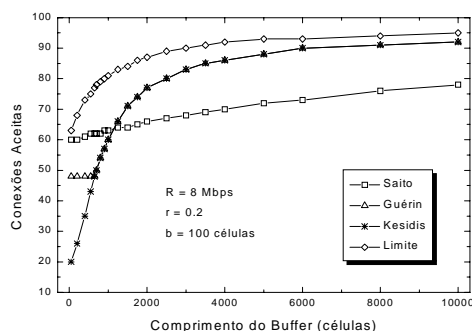


Figura 4.1: Número de conexões aceitas em função do tamanho do *buffer*.

A utilização obtida no enlace é expressiva apenas para valores grandes do *buffer* (Fig. 4.2). Tamanhos de *buffer* até 1.000 células não possibilitam utilização elevada.

Novamente, percebe-se a vantagem da aproximação gaussiana implementada no algoritmo de Guérin et al., pois o número de conexões aceitas é igual na região de capacidade equivalente e maior na região onde a aproximação gaussiana prevalece.

A capacidade de 2.000 células já é suficiente para atingir uma utilização do enlace de, aproximadamente, 0,8. A proposta de Saito et al. é superior as outras apenas quando o *buffer* é pequeno.

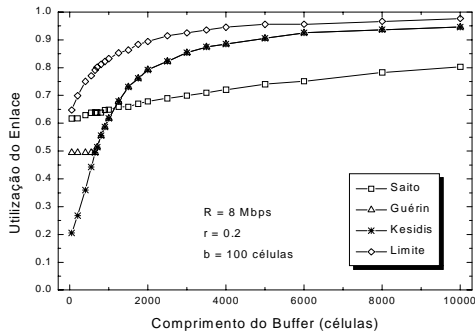


Figura 4.2: Utilização do enlace em função do tamanho do buffer.

Os atrasos obtidos (Fig. 4.3) são muito pequenos apenas quando o *buffer* também é pequeno ($K < 2.000$ células). Além deste ponto, o atraso cresce rapidamente. Valores maiores do *buffer* permitem alta utilização do enlace, contudo provocam tempos de espera muito elevados.

Verifica-se que utilizações da ordem de 0,6, como foi obtido pela alocação dinâmica, resultam em tempos de espera praticamente nulos, o que é desejável nas conexões de tempo real. Para utilizações superiores a 0,6, o tempo de atraso aumenta rapidamente e pode chegar a valores muito elevados.

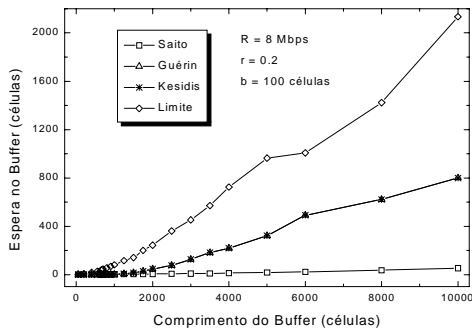


Figura 4.3: Tempo médio de espera em função do tamanho do buffer.

O segundo conjunto de simulações verifica o comportamento do algoritmo frente à variação do comprimento médio do surto. Utiliza fontes com taxa de pico de 8 Mbps, capacidade do *buffer* de 4.000 células e carga de 0,3. Verifica-se que, novamente, o algoritmo de Guérin et al. [1] tem melhores resultados (Fig. 4.4). Aqui pode-se deduzir que a diferença entre os algoritmos acontece quando o *buffer* é pequeno em relação ao tamanho médio do surto.

Os algoritmos de Saito et al. e Kesidis et al mostram-se pouco sensíveis à variação do tamanho médio do surto numa larga faixa. O esquema de alocação dinâmica apresenta resultado inferior aos outros métodos para surtos de até 500 células, mas é bem superior aos outros para valores maiores.

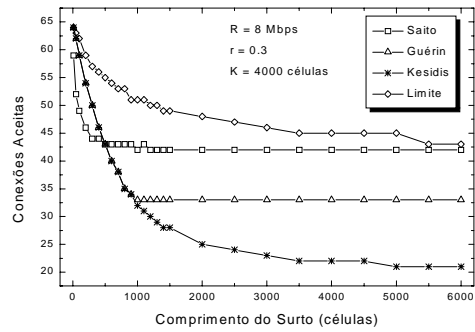


Figura 4.4: Número de conexões aceitas em função do comprimento médio do surto.

A utilização cai muito rapidamente com o crescimento do tamanho do surto (Fig. 4.5), mas pode ser bastante alta para surtos menores que 500 células. Obviamente, o tamanho do *buffer* influi fortemente nesta queda abrupta. Valores maiores permitiriam um decaimento mais suave.

Ainda assim, o método de alocação dinâmica consegue uma utilização do enlace de 0,65, enquanto o de Guérin et al. fica em 0,5. O limite estimado apresenta uma utilização em torno de 0,7.

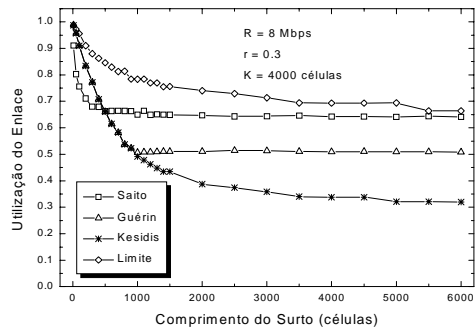


Figura 4.5: Utilização do enlace em função do comprimento médio do surto.

O tempo de espera é elevado nos pontos em que o enlace tem alta utilização (Fig. 4.6). Se o tempo for um parâmetro crítico, é necessário optar por operar o multiplexador em níveis de utilização menores.

Na maior parte da região estudada, os tempos de atraso são muito pequenos, atingindo o limite em torno de 100 células.

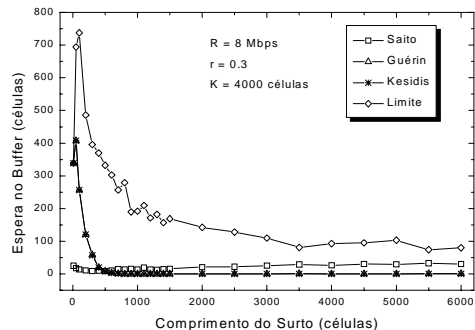


Figura 4.6: Tempo médio de espera no buffer em função do comprimento médio do surto.

O terceiro conjunto de simulações mostra como a variação da carga, r , das fontes afeta os resultados dos esquemas estudados. Utiliza fontes com taxa de pico de 8

Mbps, capacidade do *buffer* de 1.000 células e comprimento médio dos surtos de 200 células. As curvas, novamente, confirmam que os modelos são conservadores.

Em especial, a proposta de Kesidis, Walrand e Chang [2] é excessivamente conservadora quando a carga das fontes é pequena (Fig. 4.7). Comprovamos através de simulações adicionais que esta diferença diminui na medida em que a razão K/b é maior. Em outras palavras, o desempenho deste algoritmo se aproxima do outros a medida que o tamanho do *buffer* é maior em relação ao tamanho médio dos surtos.

A proposta de Saito et al. tem melhores resultados para cargas até 0,4, mas é superado ligeiramente pelos outros esquemas nas cargas mais altas.

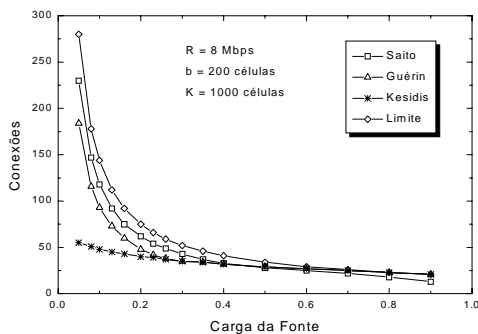


Figura 4.7: Número de conexões aceitas em função da carga das fontes.

Para obter uma boa utilização do enlace é necessário um *buffer* suficientemente grande para armazenar as eventuais seqüências longas de surtos agregados. Pode-se ver que a utilização foi baixa na região em que a carga é pequena (Fig. 4.8). Isso significa que o *buffer* utilizado não foi bastante grande e havia espaço para aceitação de mais conexões com uma memória maior. Contudo, foi observado que isso limitou os atrasos de células em valores pequenos (Fig. 4.9).

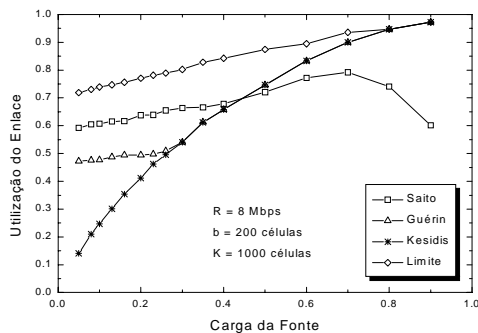


Figura 4.8: Utilização do enlace em função da carga das fontes.

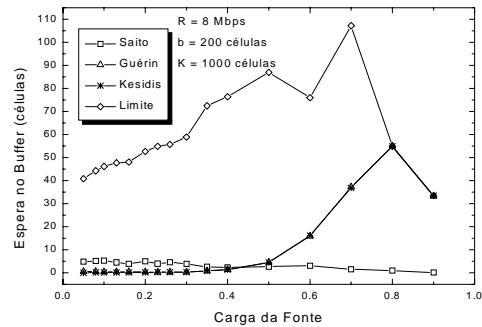


Figura 4.9: Tempo médio de espera no buffer em função da carga das fontes.

5. CONCLUSÕES

Os três algoritmos selecionados foram estudados do ponto de vista de máxima aceitação de conexões dentro dos limites de atendimento da rede. Este estudo levou em consideração a variação do tamanho do *buffer*, do surto e da carga das fontes.

Todos os algoritmos mostraram-se conservadores. A proposta de Kesidis, Walrand e Chang [2] teve o pior desempenho dentre os três. O método de Saito e Shiomoto [4] conseguiu bons resultados em várias situações, mas teve baixo desempenho com *buffers* grandes mostrando-se pouco sensível ao seu tamanho.

6. REFERÊNCIAS BIBLIOGRÁFICAS

- [1] R. Guérin, H. Ahmadi, and M. Naghshineh, "Equivalent capacity and its application to bandwidth allocation in high-speed networks," *IEEE J. Select. Areas Commun.*, vol. 9, pp. 968-981, 1991.
- [2] G. Kesidis, J. Walrand, and C.-S. Chang, "Effective bandwidths for multiclass Markov fluids and other ATM sources," *IEEE Trans. Networking*, vol. 1, no. 4, pp. 424-28, Aug. 1993.
- [3] H. Saito, and K. Shiomoto, "Dynamic call admission control in ATM Networks," *IEEE J. Select. Areas Commun.*, vol. 9, no. 7, pp. 982-989, 1991.
- [4] H. G. Perros and K. M. Elsayed, "Call admission control schemes: a review," *IEEE Communications Magazine*, November 1996.
- [5] C. Courcoubetis, G. Kesidis, A. Ridder, J. Walrand, and R. Weber, "Admission control and routing in ATM networks using inferences from measured buffer occupancy," *IEEE Trans. Comm.*, Feb/Mar/Apr, 1995.