

Análise Transiente de Modelos de Fonte Multimídia *

Carlos E. Fisch de Brito ^t Edmundo de Souza e Silva
Morganna C. Diniz [‡] Rosa M.M. Leão

Universidade Federal do Rio de Janeiro
Departamento de Ciência da Computação, NCE e COPPE/Sistemas
Cx. P. 2324, Rio de Janeiro, RJ - 20001-970 - Brasil
e-mail: {fisch, edmundo, morganna, rosa}@land.ufrj.br

Resumo

O tráfego a ser transportado pelas modernas redes de alta velocidade, se caracteriza por uma grande quantidade de informação e uma grande variabilidade na taxa de transmissão. Além disso é primordial garantir uma qualidade de serviço (QoS) mínima para a aplicação do usuário durante uma conexão, de forma a satisfazer os requisitos da mesma. Por este motivo, muito esforço tem sido feito no sentido de compreender os problemas encontrados na multiplexação ATM com o objetivo de maximizar a utilização dos recursos disponíveis sem comprometer a qualidade do serviço oferecido às aplicações. Dentre os principais problemas que causam a queda na qualidade de serviço, podemos destacar a perda de células causada pela falta de espaço de armazenamento nos *buffer*s dos comutadores. Uma vasta literatura existe sobre esse tópico. O objetivo deste trabalho é o de apresentar um novo algoritmo que permite o cálculo eficiente da distribuição do tamanho do *buffer* no tempo t de uma fila finita alimentada por uma ou mais fontes, diretamente do modelo de fonte usado. Podemos então estudar a variação do tamanho da fila com o tempo, ao invés de obter este tamanho apenas em regime, como na maioria dos resultados na literatura. Como subproduto, obtemos outras medidas de interesse, como o tempo médio em que o *buffer* permanece cheio acima de um dado valor, durante um intervalo finito de observação. Estes resultados são importantes, por exemplo, para estudar a influência de diferentes tráfegos sobre um canal de transmissão.

Abstract

The *traffic carried* by modern high speed networks have drastically *different* characteristics. New multimedia, real time applications require that a minimum QoS be offered by the network. Therefore a lot of modeling effort has been spent in order to better understand the issues involved in multiplexing multimedia traffic over high speed links, and to maximize the allocation of network resources. Among the main problems that cause QoS degradation is the loss of cells due to buffer overflow in the switches. The main goal of this work is to present a new algorithm to calculate efficiently the transient queue length distribution, when the queue is fed by given traffic sources, directly from the traffic model being used. Thus, the algorithm we developed allows the study of the queue length variation over a given period of time, unlike most studies that are aimed at the steady state behavior. As a byproduct, we obtain the fraction of time the queue length is above a given value, during a finite observation interval. These results are important, for instance to study the influence of different types of traffic over a high speed channel.

*Este trabalho conta com o apoio do CNPq/ProTeM e PRONEX.

^tO trabalho de C.E.F. Brito foi apoiado por bolsa de mestrado da CAPES.

[‡]O trabalho de M.C. Diniz foi apoiado por bolsa de doutorado do CNPq.

transiente do *buffer* pode ser bastante diferente para os dois modelos, embora estes sejam oriundos da mesma fonte real. Trabalhos futuros incluem um estudo detalhado do comportamento transiente de vários outros modelos Markovianos de tráfego incluindo o *pseudo self-similar* de forma a entender melhor a sensibilidade das medidas de interesse com os parâmetros do modelo.

Referências

- [1] D. Anick, D. Mitra, and M. M. Sondhi. Stochastic theory of a data-handling system with multiple sources. *The Bell System Technical Journal*, 61(8):1871-1894, 1982.
- [2] R.L.M. Carmo, E. de Souza e Silva, and S.C. de Lucena. Cálculo de descritores de tráfego em modelos markovianos de fontes multimídia. In *SBRC'97*, pages 189-204, 1997.
- [3] Carlos Eduardo Fisch de Brito. Medidas transientes para recompensa acumulada limitada: Novas técnicas de solução e aplicações para redes de alta velocidade, Junho 1999. Tese de Mestrado - UFRJ - COPPE/Sistemas.
- [4] E. de Souza e Silva and H. R. Gail. Calculating availability and performability measures of repairable computer systems using randomization. *Journal of the ACM*, 36(1):171-193, 1989.
- [5] E. de Souza e Silva and H.R. Gail. Performability analysis of computer systems: from model specification to solution. *Performance Evaluation*, 14:157-196, 1992.
- [6] E. de Souza e Silva and H.R. Gail. The uniformization method in performability analysis. Technical report, IBM Research Report RC 20383, Yorktown Heights, N. Y., 1996.
- [7] E. de Souza e Silva and H.R. Gail. An algorithm to calculate transient distributions of cumulative rate and impulse based reward. *Communications in Statistics - Stochastic Models*, 14(3):509-536, 1998.
- [8] E. de Souza e Silva and H.R. Gail. Transient solutions for Markov chains. In W. Grassmann, editor, *Computational Probability*, pages 43-81. Kluwer Academic Publishers, 2000.
- [9] E. de Souza e Silva, H.R. Gail, and R. Vallejos Campos. Calculating transient distributions of cumulative reward. In *Proc. Performance '95 and 1995 ACM SIGMETRICS Conf.*, pages 231-240, 1995.
- [10] E. de Souza e Silva, H.R. Gail, and J.C. Guedes. Transiente distributions of cumulative rate and impulse based reward with applications. In *System Modelling and Optimization*, pages 298-306. CRC Press, 1999.
- [11] M.C. Diniz. Técnicas de solução para modelos provenientes de redes multimídia, Fevereiro 2000. Tese de Doutorado - UFRJ - COPPE/Sistemas.
- [12] L. Donatiello and V. Grassi. On evaluating the cumulative performance distribution of fault-tolerant computer systems. *IEEE Trans. on Computers*, 40(11):1301-1307, 1991.
- [13] D.P. Heyman and T.V. Lakshman. What are the Implications of Long-Range Dependence for VBR-Video Traffic Engineering. *IEEE/ACM Transactions on Networking*, 4(3):301-317, Junho 1996.

Valor esperado da fração de tempo que o sistema fica com o *buffer* acima de um percentual

O valor esperado da fração do tempo que o sistema fica com o *buffer* acima de um determinado percentual, durante um intervalo de observação, é mostrado na Figura 7. Neste exemplo consideramos valores de *buffer* variando de 500 a 4000. O valor máximo de 4000 para o tamanho de *buffer* foi escolhido de forma a obter, em estado estacionário, uma probabilidade de ocupação acima de 99%, em torno de 10^{-6} , isto é, a fração de tempo em que o *buffer* está praticamente cheio é muito baixa. Calculamos a medida da Figura 7 para o modelo do histograma considerando como estado inicial aquele em que a taxa da fonte é a maior possível, e o intervalo de observação é de $t=20\text{ms}$.

Pela figura podemos observar que, a medida que aumentamos o tamanho do *buffer*, a fração de tempo com o *buffer* cheio cai rapidamente até o *buffer* atingir o valor de 2000. A partir deste ponto a queda é bem lenta. Além disso, podemos observar que, mesmo quando o *buffer* é igual a 4000, mais de 12,5% (em média) dos 20ms (isto é, mais de 2,5ms) o *buffer* está cheio. Isto indica que mesmo que a probabilidade de perda seja baixa os períodos de perda podem ser longos em relação ao intervalo entre transmissão de células.

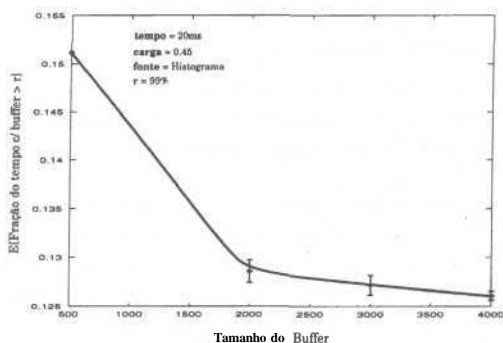


Figura 7: Fração do tempo que o *buffer* fica acima de um certo percentual

5 Conclusões

Neste trabalho apresentamos um novo algoritmo para o cálculo da distribuição da recompensa acumulada em um dado intervalo de tempo, em modelos Markovianos com recompensa, quando existe um limite inferior e superior para o valor desta recompensa. Este resultado é usado em modelos de fluido de fontes, onde o objetivo é o cálculo da distribuição *transiente* do tamanho de *buffer*, alimentado por uma fonte genérica. O modelo resultante não inclui explicitamente o tamanho do *buffer* como variável de estado, e o algoritmo de solução tem o mesmo custo computacional independentemente deste parâmetro. Mostramos também como é possível calcular a fração de tempo em que modelo fica acima de um dado limite de recompensa, em um intervalo de tempo.

Para ilustrar a aplicabilidade dos resultados, usamos dois modelos parametrizados a partir de uma mesma fonte real. As medidas obtidas mostram que o comportamento

de um determinado valor de ocupação no tempo igual a 20ms ($P[b(t) > rB]$) quando o sistema está em alta carga, a medida em questão é bastante diferente quando o sistema está em baixa carga. Portanto, mesmo em baixa carga, o modelo de histograma causa um enchimento rápido do *buffer* em comparação ao modelo *on-off*.

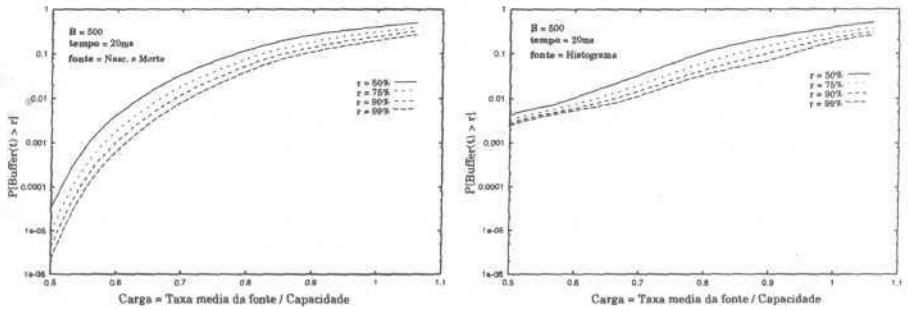


Figura 5: Variação da capacidade de serviço

• Variação do Estado Inicial do Modelo

A Figura 6 exibe os resultados obtidos quando variamos o estado inicial do modelo. A característica mais interessante, é novamente a diferença entre o comportamento dos dois modelos. No modelo com a fonte de nascimento e morte as probabilidades aumentam suavemente, conforme iniciamos o intervalo em estados com taxas de transmissão cada vez maiores. No modelo com a fonte de histograma, existem bruscas variações quando alteramos o estado inicial da fonte. Isto se explica porque, para a fonte de nascimento e morte, as taxas de transmissão de células e as probabilidades de transição de estado variam *linearmente* com o índice do estado. No modelo de histograma, estes parâmetros dependem da amostragem sobre a seqüência de quadros do filme, portanto podem existir estados com alta taxa de transmissão e alta probabilidade de permanência, apesar da taxa média ser a mesma do modelo de nascimento e morte.

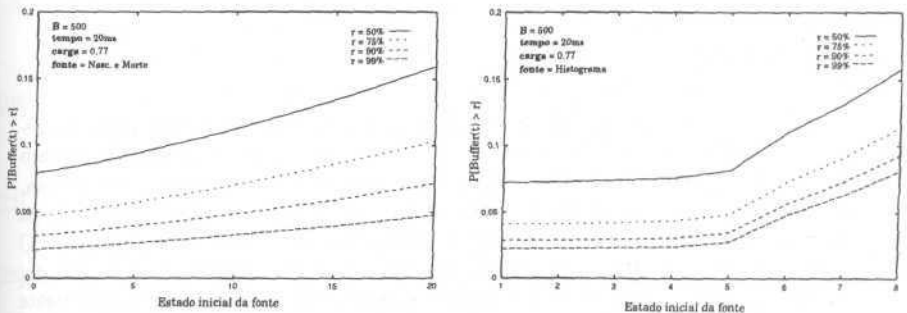


Figura 6: Variação do estado inicial do modelo

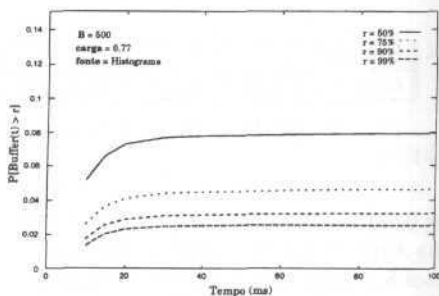
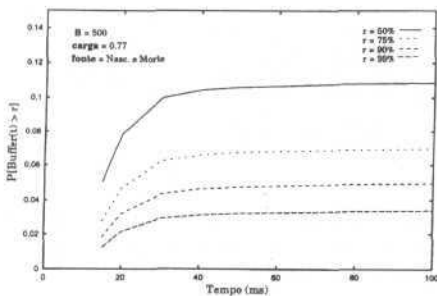


Figura 3: Variação do intervalo de observação

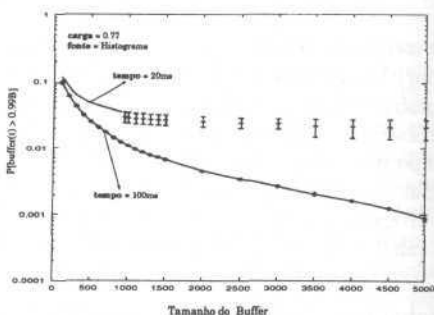
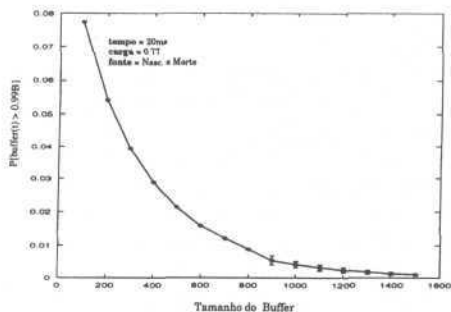


Figura 4: $P[\text{buffer}(t) > 0.99B]$ com o tamanho do *buffer*

É interessante observar na Figura 4 que, a partir do tamanho de *buffer* igual a 900 células, a restrição dada pela equação (11) não é satisfeita. Entretanto, conforme argumentamos após o Corolário 2, podemos calcular os limites superior e inferior para a medida. As linhas verticais indicam os limites na figura.

A Figura 4 mostra para os dois modelos que, mesmo aumentando bastante o tamanho do *buffer* de armazenamento de células, a probabilidade do *buffer* estar com 99% de ocupação no tempo igual a 20ms permanece aproximadamente constante, para valores do *buffer* acima de 1000 células. Entretanto, esta probabilidade diminui significativamente com o tamanho do *buffer* quando $t = 100\text{ms}$ (ver figura relativa ao modelo de histograma). Este resultado mostra a importância da análise transiente. Ao analisarmos o sistema antes deste atingir o regime estacionário, obtemos valores para a probabilidade de ocupação do *buffer* que são duas ordens de grandeza superiores aos valores obtidos para um tempo próximo ao regime estacionário (considerando *buffer* de 5000 células).

- *Variação da capacidade de serviço*

Neste experimento, variamos a capacidade do servidor, e obtivemos medidas para o sistema sujeito à carga de 0.5 até 1.06. O estado inicial considerado para os modelos foi aquele onde todas as fontes estão em Off (modelo de nascimento e morte) e estado de menor taxa da fonte (modelo de histograma). Pela Figura 5 notamos que, apesar das duas fontes apresentarem valores semelhantes para a probabilidade do *buffer* estar acima

Estado	s_1	s_2	s_3	s_4	s_5	s_6	s_7	s_8
cel/ms	117.70	230.67	343.61	456.58	569.55	682.52	795.49	908.43

Tabela 1: Taxas de transmissão de células em cada estado

$\alpha = 0.0712$ e taxa de morte $\beta = 0.1525$, onde a taxa de transmissão em cada estado é igual a $n * \lambda$ cel/ms sendo n o índice do estado e $A = 50$ cel/ms. Já no modelo do histograma, temos 8 estados com as taxas de transmissão apresentadas na Tabela 4.1.

4.2 Resultados dos Experimentos

Apresentamos a seguir os resultados obtidos com os dois modelos descritos na seção anterior. Uma medida de interesse adotada para a avaliação do desempenho do sistema, é a probabilidade do *buffer* do comutador estar com ocupação acima de um determinado percentual em um instante t . Uma outra medida calculada é o valor esperado da fração do tempo que o *buffer* fica acima de um determinado percentual em um intervalo de observação. Nos experimentos, é avaliada a sensibilidade das medidas de interesse à variação de diversos parâmetros, tais como intervalo de observação, tamanho do *buffer*, capacidade do canal e estado inicial da fonte. (Abaixo, chamamos de *carga* a razão entre taxa média de transmissão da fonte com a capacidade do canal.)

Probabilidade do *buffer* do comutador estar com ocupação acima de um determinado percentual

- *Variação do intervalo de observação*

Neste primeiro conjunto de medidas, procuramos analisar o sistema para diversos intervalos de observação e diversos percentuais de ocupação do *buffer*. Os resultados estão exibidos na Figura 3 onde o rótulo r representa o percentual de ocupação do *buffer*. Os intervalos escolhidos ficaram na faixa de 10 a 100 ms. O estado inicial do modelo de nascimento e morte é aquele onde todas as fontes estão em *Off* do modelo de histograma é o estado onde a fonte tem a menor taxa. Uma característica interessante que pode ser observada neste exemplo, é a diferença entre o comportamento dos modelos das duas fontes. No primeiro modelo, notamos que o sistema entra em estado estacionário por volta de $t = 60ms$. Para o segundo modelo, o estado estacionário é atingido ao redor de $t = 40ms$.

- *Variação do tamanho do buffer*

Neste experimento, avaliamos a distribuição do tamanho da fila do *buffer* ($b(t)$) com a sua capacidade de armazenamento de células. Foram examinados sistemas com capacidade para armazenar de 100 a 1500 células para o modelo de nascimento e morte e sistemas com o *buffer* variando de 100 a 5000 células para o modelo de histograma.

O estado inicial considerado para o modelo de nascimento e morte é o mesmo do experimento anterior (todas as fontes em *Off*). Já para o modelo de histograma, consideramos o seu estado inicial como sendo o de maior taxa da fonte e calculamos a medida para dois instantes de tempo: 20ms e 100ms. (*Este estado inicial foi escolhido de forma a evidenciar o comportamento descrito abaixo.*)

Logo,

$$E[\Psi(r, t)] = \int_0^t P[CR(\tau) > r] d\tau. \quad (13)$$

Substituindo (8) em (13), e resolvendo a integral obtemos

$$E[\Psi(r, t)] = \sum_{n=0}^{\infty} E_{n+1, \Lambda}(t) \sum_{s \in S} \Upsilon_s(n, 0, l-1). \quad (14)$$

onde $E_{n+1, \Lambda}(t) = 1 - \sum_{j=0}^n e^{-\Lambda t} (\Lambda t)^j / j!$ é a distribuição de Erlang com $n + l$ estágios. É interessante notar que a recursão de (8) pode ser usada tanto para se obter $P[CR(t) > B]$ como para se obter $E[\Psi(r, t)]$, com praticamente o mesmo custo computacional.

4 Aplicações

Nesta seção apresentamos alguns resultados de experimentos onde estudamos o comportamento do modelo de um sistema composto por fontes de tráfego e um comutador ATM. Neste modelo, são associadas à fonte de tráfego recompensas tal que o valor da recompensa acumulada em t representa o tamanho do *buffer* do comutador. Uma grande vantagem do método de recompensa acumulada limitada, como citado anteriormente, é que o estado da fila não é representado explicitamente no modelo. Como consequência, a complexidade da solução não é afetada pelo tamanho do *buffer* do sistema analisado. Nos nossos exemplos temos uma fonte de tráfego de 21 estados, mas se modelássemos este sistema usando a fonte acrescida de uma fila com *buffer* igual a 1000, teríamos um modelo Markoviano com 21000 estados. No modelo com recompensas, além da taxa de serviço ser constante (o que se aproxima mais da realidade no caso de células de tamanho fixo) temos somente os 21 estados da fonte.

4.1 Definição dos Modelos

A seguir apresentamos alguns detalhes dos modelos utilizados. A fonte de tráfego procura aproximar o comportamento de uma transmissão de vídeo codificado em MPEG. O tráfego gerado por esta fonte possui taxa de envio de células variável (VBR). Dois tipos de modelos para a fonte de tráfego foram considerados. O primeiro modelo é uma cadeia de Markov de nascimento e morte, representando um conjunto de fontes *OnOff*dênticas e independentes. O modelo é parametrizado através do casamento de estatísticas calculadas para a seqüência real com as obtidas para o modelo. O segundo modelo [22], conhecido como modelo de histograma, é construído através de uma amostragem sobre a seqüência real de vídeo. O objetivo da amostragem é capturar a variação da taxa de transmissão ao longo do tempo. Assim, obtêm-se amostras da seqüência real em instantes determinados do tempo, estabelecem-se faixas de taxas de transmissão e calcula-se a fração de transições de uma faixa a outra. A partir desses dados, constrói-se uma cadeia de Markov, que representa a seqüência de vídeo. Como no caso anterior, a cada estado da cadeia é associada uma taxa de transmissão. (Lembramos que a cada estado a recompensa associada é dada pela taxa de transmissão menos a capacidade transmissão do canal.)

A partir da seqüência do filme *Star Wars* disponível em domínio público, obtemos para o modelo de Nascimento e Morte 20 fontes *OnOff*çada uma com taxa de nascimento

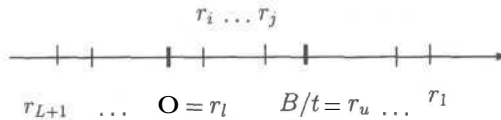


Figura 2: Taxas de Recompensa Violando a Restrição

modificarmos o modelo, de modo que $r_i = \dots = r_j = r_l = 0$ então, o modelo passa a satisfazer (11), e a medida obtida representa um limite inferior para $P[CR(t) > r]$ do modelo original. Isto é verdade uma vez que $CR(t)$ deste novo modelo é sempre inferior ao do modelo original para qualquer caminho amostrai. Por outro lado, fazendo $r_i = \dots = r_j = r_u = B/t$, a medida obtida representa um limite superior para $P[CR(t) > r]$ do modelo original. Desta forma é sempre possível obter uma solução com limites.

3.3 Custos Computacionais

Para o cálculo de $P[CR(t) > r]$ a partir de (8) é preciso truncar a soma infinita. Isso pode ser facilmente feito de forma a conseguir um resultado tão preciso quanto se queira [6]. É possível mostrar que $N < At$ onde $A = \max_i \{q_i\}$ em (1). Se truncarmos a série infinita em N , o custo de (8) para buffer limitado é igual a $2MN^2$, onde lembramos que M é a cardinalidade do espaço de estados. Outro ponto importante é que $P[CR(t) > B]$ pode ser calculado para valores distintos de t , em paralelo, sem que se incorra em nenhum custo adicional relevante.

É importante enfatizar que, no modelo com recompensas, o tamanho do *buffer* não é explicitamente representado na cadeia de Markov. Em outras palavras, a cadeia de Markov em questão representa apenas as fontes que alimentam o *buffer* e portanto a cardinalidade do espaço de estados M é pequena.

3.4 Valor Esperado do Período de Perda de Células

Uma medida importante para análise do comportamento de uma fila alimentada por fontes de diferentes tipos é o tempo médio em que o *buffer* se encontra acima de um determinado valor, durante um período de observação. Note que, mesmo que a probabilidade de perda para $t \rightarrow \infty$ seja pequena, podemos ter intervalos relativamente curtos (em comparação à duração de uma aplicação do usuário) com uma fração alta de tempo (durante o intervalo) onde o *buffer* fica congestionado. Isto pode levar a perdas significativas da qualidade de serviço para uma aplicação durante aquele período, embora no total as perdas sejam pequenas. A seguir mostraremos como calcular esse valor esperado, usando os resultados da seção anterior.

Seja $\Psi(r, t)$ a variável aleatória igual ao tempo que o sistema permanece com uma recompensa total acumulada acima do valor r . Considere a variável aleatória indicadora $\mathcal{I}\{CR(t) > r\}$. Portanto $\mathcal{I}\{CR(t) > r\} = 1$ quando $CR(t) > r$ e 0 caso contrário. Então,

$$\Psi(r, t) = \int_0^t \mathcal{I}\{CR(\tau) > r\} d\tau. \quad (12)$$

Prova: É obtida da prova probabilística do Teorema 1 [11]. □

A importância da interpretação probabilística dada pelo Teorema 2 é que podemos facilmente introduzir limites ao valor de $CR(t)$. Dos Teoremas 1 e 2, notamos que, para calcular os termos intermediários da recursão associados à taxa de recompensa r_h ($\Upsilon(n, m, h)$) para todo n e $m < n$, é necessário conhecer apenas os termos $\Upsilon(n, 0, h)$, $r_{c(s)} > r_h$, e $\Upsilon(n, n, h)$, $r_{c(s)} < r_h$. Ou seja, não são necessários os termos associados a nenhuma outra recompensa. O próximo resultado é fundamental para a introdução de limites para $CR(t)$.

Corolário 1 *Suponha que o último estado visitado pelo processo no intervalo $(0, t)$ seja o estado s . Então, as seguintes condições são verdadeiras, para qualquer instante do tempo t :*

1. Se $r_{c(s)} > 0$, então a recompensa acumulada pelo processo durante o intervalo $(0, t)$ não pode ser igual ao limite inferior.
2. Se $r_{c(s)} < 0$, então a recompensa acumulada pelo processo durante o intervalo $(0, t)$ não pode ser igual ao limite superior.

Prova: É obtida a partir do Teorema 2 [3]. D

Corolário 2 *A distribuição da recompensa acumulada em $(0, t)$ ($P[CR(t) > rt]$) com limites tais que $r_l t < rt < r_u t$ pode ser calculada da equação (8), desde que a restrição abaixo seja satisfeita:*

$$r_{L+1} < r_L < \dots < r_1 < 0 \leq r_u < \dots < r_1. \quad (11)$$

Prova: A prova consiste em mostrar que podemos obter todas as condições iniciais necessárias a recursão, usando o Corolário 1 e a interpretação probabilística dada pelo Teorema 2 (ver [3]). D

Apesar de (11) aparentemente restringir o uso do algoritmo com limites, na prática a restrição não apresenta maiores problemas, no caso de modelos de fluido. Como o *buffer* da fila não pode ser menor que 0, então $r_l = 0$, e a restrição (11) sempre é satisfeita para o limite inferior.

Seja B o tamanho finito do *buffer*, e r^+ a menor recompensa positiva do modelo de fluido. Então de (11), $r_u = B/t < r^+$. Para valores relativamente grandes de B , a restrição pode não ser satisfeita para t relativamente pequeno. Por outro lado, estamos interessados em valores de t onde pelo menos haja tempo suficiente para encher o *buffer* de forma a calcular a probabilidade de perda. Então, estamos interessados em valores de t tais que $A(i)t \gg B$ (onde $\lambda(t)$ é a taxa média de transmissão no período de observação $(0, t)$). Seja $\lambda(t) = \sigma B$ ($\sigma > 1$). Portanto, a restrição passa a ser $\lambda(t) < \sigma r^+$, que é muitas vezes satisfeita como veremos nos exemplos da seção 4.

Por outro lado, mesmo que a restrição não seja satisfeita, podemos *sempre* calcular limites superiores e inferiores para $P[CR(t) > B]$. Observemos a Figura 2, onde o valor das recompensas está indicado em relação aos limites superior (B/t) e inferior (0). Se

Seja \mathcal{R} o conjunto das $L+1$ taxas de recompensas $r_1 > r_2 > \dots > r_{L+1}$ associadas aos estados do modelo. Suponha também que não existem limites para os valores de $CR(t)$, que portanto pode variar de $r_{L+1}t$ a $r_1 t$.

Teorema 1 Para $i = 1, \dots, L+1$ e $r_i \in \mathcal{R}$,

$$P[CR(t) > r_i t] = \sum_{n=0}^{\infty} e^{-\lambda t} \frac{(\lambda t)^n}{n!} \sum_{s \in \mathcal{S}} \Upsilon_s(n, \mathbf{0}, i-1). \quad (8)$$

onde $\Upsilon_s(n, m, j)$ pode ser calculado pela seguinte recursão, para $n > 0$, $m = 0, \dots, n$ e $j = 1, \dots, L$ e $s \in \mathcal{S}$,

$$\Upsilon_s(n, m, j) = \begin{cases} \left(\frac{r_{c(s)} - r_j}{r_{c(s)} - r_{j+1}} \right) \Upsilon_s(n, m-1, j) + \left(1 - \frac{r_{c(s)} - r_j}{r_{c(s)} - r_{j+1}} \right) \sum_{s' \in \mathcal{S}} \Upsilon_{s'}(n-1, m-1, j) p_{s',s} \\ \quad \text{para qualquer } r_{c(s)} \geq r_j \\ \left(\frac{r_{j+1} - r_{c(s)}}{r_j - r_{c(s)}} \right) \Upsilon_s(n, m+1, j) + \left(1 - \frac{r_{j+1} - r_{c(s)}}{r_j - r_{c(s)}} \right) \sum_{s' \in \mathcal{S}} \Upsilon_{s'}(n-1, m, j) p_{s',s} \\ \quad \text{para qualquer } r_{c(s)} < r_j \end{cases} \quad (9)$$

Com as condições iniciais:

Para $n = 0$

$$\Upsilon_s(0, 0, j) = \begin{cases} \pi_s(0) & \text{se } j \text{ é tal que } r_{c(s)} \geq r_j \\ 0 & \text{se } j \text{ é tal que } r_{c(s)} < r_j \end{cases}$$

para $n > 0$

$$\Upsilon_s(n, m, j) = \begin{cases} \pi_s(0) & \text{se } m = 0, r_{c(s)} > r_j \text{ e } j = L \\ 0 & \text{se } m = n, r_{c(s)} < r_j \text{ e } j = 1 \\ \Upsilon_s(n, 0, j-1) & \text{se } m = n, r_{c(s)} \neq r_j \text{ e } j = 2, \dots, L \end{cases} \quad (10)$$

Prova: ver [11].

D

Observação: o Teorema acima é válido para $P[CR(t) > n]$ tal que $r_i \in \mathcal{R}$. Entretanto, pode ser demonstrado que as mesmas equações são obtidas mesmo que r_i não pertença a \mathcal{R} , bastando para isso incluir r_i no conjunto \mathcal{R} apesar da recompensa r_i não estar associada a nenhum estado do modelo. Em outras palavras, r_i é uma taxa de recompensa artificialmente introduzida.

Teorema 2 Os termos $\Upsilon_s(n, m, j)$ da recursão acima têm a seguinte interpretação probabilística para $m = 0$ e $m = n$, $n > 0$:

$$\begin{aligned} \Upsilon_s(n, 0, j) &= P[CR(t) > r_{j-1}t \mid n \text{ transições de estados em } (0, t) \\ &\quad \text{e o estado visitado após a } n\text{-ésima transição é } s] \\ \Upsilon_s(n, n, j) &= P[CR(t) > r_j t \mid n \text{ transições de estado em } (0, t) \\ &\quad \text{e o estado visitado após a } n\text{-ésima transição é } s] \end{aligned}$$

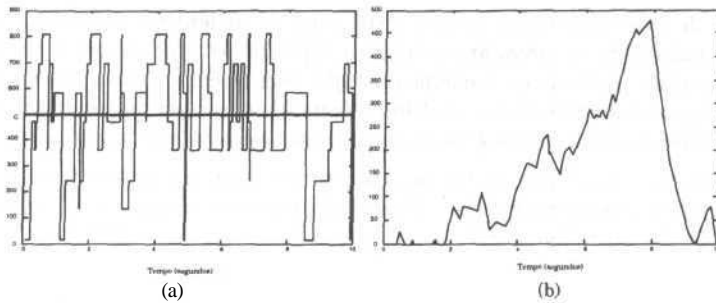


Figura 1: Comportamento das variáveis $r_c(X(t))$ e $b(t)$

Como consequência da observação acima, nosso problema se resume ao cálculo da distribuição de $CR(t)$ com limites, de forma **eficiente**. Este é o objetivo da seção seguinte.

3.2 Cálculo de $P[CR(t) > r]$

O cálculo da distribuição da recompensa acumulada em um intervalo de tempo $(0, t)$ ($P[CR(t) > r]$) em modelos Markovianos tem sido tratado em vários trabalhos. de Souza e Silva e Gail [4] usaram a técnica de uniformização e desenvolveram uma metodologia baseada em argumentos probabilísticos, para o cálculo de diversas medidas, incluindo $P[CR(t) > r]$. O algoritmo para o cálculo de $P[CR(t) > r]$ entretanto tem custo combinatorial com os parâmetros do modelo. Pattipati *et al* [19] empregaram métodos de solução de equações diferenciais parciais para resolver uma equação matricial semelhante à (7). O método entretanto é complexo e possui problemas numéricos. Donatiello e Grassi [12] usaram o método de uniformização e obtiveram uma expressão para a transformada de Laplace (em duas dimensões) da distribuição de $CR(t)$. Embora a transformada pudesse ser invertida, dando origem a equações recursivas com complexidade computacional polinomial com os parâmetros do modelo, a recursão é numericamente instável, e útil apenas para modelos bem pequenos. Em [9] um novo algoritmo com custo polinomial foi obtido baseado na metodologia de [4], e em [7] os resultados foram estendidos para incluir recompensas tipo impulso. Posteriormente, Nablí e Sericola [18] obtiveram um novo algoritmo para o caso de modelos com taxas de recompensas. Embora esse novo algoritmo não tenha sido obtido a partir de argumentos probabilísticos, e seu custo seja um pouco superior ao de [9], sua vantagem é que se utiliza de números positivos no intervalo $[0, 1]$ o que o torna atrativo do ponto de vista numérico. Este algoritmo foi estendido recentemente em [21], para incluir limite inferior em O para $CR(t)$.

O algoritmo que apresentamos abaixo, obtido recentemente em [11] é baseado em argumentos probabilísticos idênticos aos usados em [9]. A recursão final é ligeiramente diferente da obtida em [18], e também possui as mesmas vantagens numéricas. Uma grande vantagem dos argumentos probabilísticos é que, a partir deles, pudemos obter $P[CR(t) > r]$ com $CR(t)$ limitado não só inferiormente, mas também superiormente. O limite superior é importante para a solução de modelos com *buffer* limitado. O algoritmo final é simples e eficiente.

desempenho de redes multimídia, como a distribuição **transiente** do tamanho do *buffer* de um comutador, sem ser necessário introduzir explicitamente no modelo a variável de estado que indica a evolução do tamanho do *buffer* com o tempo. Isto é, para o cálculo dessa medida, apenas utilizamos o modelo da fonte. Isso nos permite analisar modelos que de outra forma seriam muito grandes para serem resolvidos exatamente.

No que se segue apresentamos um novo algoritmo de cálculo para a distribuição de $CR(t)$ com limites superior e inferior. Concluindo a seção mostramos que, a partir dos resultados obtidos, outra importante medida de interesse pode ser calculada.

3.1 Modelos de Fluido

Modelos de fluido (*fluidflow models*) tem sido muito usados para analisar o processo de perda de células em comutadores ATM (e.g. [1, 17, 14]). Neste caso o fluxo de chegada e saída de células de um comutador é aproximado por um fluido contínuo. Como, durante uma rajada, o tempo entre chegadas de células é aproximadamente constante, o intervalo entre chegadas de células é pequeno em relação à duração de uma rajada e o tempo de serviço de células é constante, etc., esses modelos tornam-se apropriados para descrever com precisão a variação do tamanho do *buffer*.

Seja $b(t)$ a quantidade de células armazenadas no *buffer* associado a um canal de saída de um comutador, no instante de tempo t . Seja $X = \{X(t), t > 0\}$ o processo que modela as fontes que alimentam o canal, isto é $X(t)$ fornece a taxa de transmissão no instante t . Supondo que X é uma cadeia de Markov, Seja $P_{ij}(t, x) = P\{X(t) = j, b(t) > x | X(0) = i\}$. É possível então obter a seguinte equação matricial diferencial parcial:

$$\frac{\partial \mathbf{P}(t, x)}{\partial t} + \mathbf{D} \frac{\partial \mathbf{P}(t, x)}{\partial x} = \mathbf{Q} \mathbf{P}(t, x) \quad (7)$$

onde \mathbf{D} é uma matriz diagonal cujo i -ésimo elemento é igual a taxa de transmissão no estado i da cadeia de Markov, subtraída da capacidade do canal.

A equação (7) pode ser resolvida usando análise espectral, envolvendo também a inversão numérica da transformada de Laplace de uma equação obtida de (7) (e.g. ver [20, 23]). A solução em estado estacionário envolve o cálculo dos autovalores e autovetores de $\mathbf{D}^{-1}\mathbf{Q}$, o que não é uma tarefa trivial a não ser para casos particulares. Além disso a equação supõe que o *buffer* em questão é infinito, como uma aproximação para o caso finito.

Não é difícil ver que o modelo de fluido pode ser imaginado como um modelo de recompensa. Suponha, como indicado acima, que o conjunto de fontes alimentando o *buffer* de um comutador seja modelado por uma cadeia de Markov X onde a taxa de transmissão (constante) no estado s_i é igual a λ_i . Seja C a capacidade do canal. Se ao estado S_i é associada uma taxa de recompensa $r_i = \lambda_i - C$, então a variável aleatória $CR(t)$ definida na seção 2 é igual ao tamanho do *buffer* no tempo t , desde que os valores de $CR(t)$ estejam limitados a 0 e ao valor máximo do *buffer* (B), isto é, $0 < CR(t) < B$ para todo o valor de t . A Figura 1 exibe o comportamento típico das variáveis $CR(t)$ (igual neste caso a $b(t)$) e $r_{c(X(t))}$. A figura abaixo foi obtida através da simulação de um processo de 8 estados (modelando uma fonte de transmissão). O gráfico (a) exibe a taxa de chegada de células no tempo t , para um determinado caminho do processo, e o gráfico (b) mostra o valor da variável $b(t)$ no decorrer deste caminho.

determinado valor de *buffer* no tempo t e a fonte estar transmitindo a uma das possíveis taxas. De (2) e (3) é possível então determinar a evolução da ocupação do *buffer* com o tempo.

Considere a variável aleatória indicadora $I(X(t) = s_i)$, tal que $I(X(t) = s_i) = 1$ se $X(t) = S_i$ e 0 caso contrário. Seja O , por exemplo, o conjunto de estados do modelo que correspondem a uma ocupação do *buffer* acima de um dado valor B . $\pi_O(t) = \sum_{i: s_i \in O} \pi_i(t)$ fornece a probabilidade do sistema estar com um valor acima de B em t . Outra variável aleatória de interesse é o tempo $\mathcal{T}_O(t)$ durante o qual o sistema passou acima de B durante $(0, t)$. Então:

$$\mathcal{T}_O(t) = \int_0^t I\{X(\tau) \in O\} d\tau. \quad (4)$$

É possível obter uma série de medidas associando taxas de recompensas aos estados de $X(t)$. Suponha que associemos uma taxa de recompensa igual a 1 para cada estado $s \in O$ e 0 caso contrário. Então, π_O é a recompensa "instantânea" em t , e $\mathcal{T}_O(t)$ é a recompensa acumulada em $(0, t)$. Em muitos casos, como veremos na próxima seção, é conveniente usarmos a teoria de modelos de Markov com recompensas para o cálculo de determinadas medidas de desempenho.

Generalizando, suponha que existem $L + 1$ taxas de recompensas $r_1 > r_2 > \dots > r_{L+1}$ e ao estado $s \in S$ é associada uma taxa de recompensa $r_{c(s)}$. Logo, uma recompensa r_j é acumulada por unidade de tempo que o sistema se encontra em qualquer estado s tal que $c(s) = j$. Por exemplo, se $S = \{1, 2, 3\}$ e estados 1 e 2 tem recompensa r_1 e 3 tem recompensa r_2 então $c(1) = c(2) = 1$ e $c(3) = 2$. A variável aleatória

$$IR(t) = r_{c(X(t))}, \quad (5)$$

é a taxa instantânea em t e a recompensa acumulada em $(0, t)$ é dada por

$$CR(t) = \int_0^t r_{c(X(\tau))} d\tau. \quad (6)$$

A recompensa acumulada $CR(t)$ tem sido chamada de *performabilidade* em modelos unificados de performance e dependabilidade. De uma maneira geral o nome tem sido usado para um conjunto de medidas, em especial medidas transientes.

Outra variável aleatória de interesse é o tempo total durante $(0, \hat{t})$ onde a recompensa está acima de um valor r ($\Psi(r, \hat{t})$). A variável $\Psi(r, \hat{t})$ pode ser obtida integrando-se a variável indicadora $I\{CR(t) > r\}$. O cálculo da distribuição de $\Psi(r, \hat{t}) > r$ será objeto da próxima seção.

3 Algoritmo de Cálculo da Recompensa Acumulada com Limites

Na seção 2 apresentamos um exemplo simples para ilustrar o uso de modelos de recompensa no cálculo de algumas variáveis de interesse. O objetivo inicial desta seção é o de mostrar que modelos de recompensa são úteis no cálculo de importantes medidas de

aproximação para um modelo de *buffer* infinito, dado que o *buffer* esvazia.

O objetivo deste artigo é o de apresentar um novo algoritmo geral para cálculo da distribuição da recompensa acumulada em um intervalo finito de tempo, onde a recompensa acumulada não pode ultrapassar limites inferior e superior *pré-estabelecidos*, o que não pode ser feito com o método descrito em [10]. (O algoritmo base é também completamente diferente daquele usado em [10].) O uso destes limites são essenciais para o cálculo eficiente de medidas importantes como veremos nas seções que se seguem. Além disso, mostramos como outras medidas importante podem ser obtidas com o mesmo algoritmo básico. Apresentamos alguns exemplos para demonstrar a utilidade do algoritmo de forma a determinar a influência de diferentes tipos de tráfego multimídia sobre o *buffer* de um canal multiplexado.

A seção 2 faz um apanhado geral sobre medidas transientes e apresenta o material básico necessário ao entendimento dos resultados do artigo. O algoritmo desenvolvido é apresentado na seção 3. Na seção 4 apresentaremos alguns exemplos que mostram a utilidade das medidas obtidas, e na seção 5 concluímos o trabalho.

2 Medidas Transientes

Nesta seção definimos algumas medidas transientes importantes para a análise de redes e de sistemas de computação em geral, e introduziremos a notação usada no artigo. Em particular, trataremos com maior ênfase de modelos Markovianos com recompensa, objeto de estudo neste trabalho. Maiores detalhes sobre o cálculo de medidas transientes podem ser encontrados em [8].

Considere uma cadeia de Markov de tempo contínuo, homogênea $X = \{X(t), t > 0\}$ com espaço de estado finito $S = \{s_i : i = 1, \dots, M\}$ e gerador infinitesimal Q .

$$Q = \begin{bmatrix} -q_1 & q_{12} & \cdots & q_{1M} \\ q_{21} & -q_2 & \cdots & q_{2M} \\ \vdots & \vdots & \ddots & \vdots \\ q_{M1} & q_{M2} & \cdots & -q_M \end{bmatrix}. \quad (1)$$

Seja a função de transição $\Pi(t)$, $t > 0$, onde para qualquer $t' > 0$,

$$\Pi(t)_{(i,j)} = P[X(t' + t) = s_j | X(t') = s_i].$$

$\Pi(t)$ satisfaz ambas equações de Kolmogorov:

$$\dot{\Pi}(t) = Q\Pi(t) \text{ e } \dot{\Pi}'(t) = \Pi(t)Q. \quad (2)$$

O vetor $\pi(t) = [\pi_1(t), \dots, \pi_M(t)]$ fornece as probabilidades de estado para X no tempo t , e são obtidos de $\Pi(t)$ por

$$\pi(t) = \pi(0)\Pi(t). \quad (3)$$

Como exemplo, considere um modelo onde uma fonte de taxa variável (tipo MMPP - Markov Modulated Poisson Process) alimenta o *buffer* finito de um computador. As variáveis de estado do modelo são o estado da fonte (que indica a taxa atual de transmissão) e o número de pacotes no *buffer*. $\pi_i(t)$ fornece a probabilidade do sistema estar com um

I Introdução

A maioria dos trabalhos relacionados a análise de desempenho de redes de computadores tem como base a obtenção de medidas em estado estacionário, isto é, medidas onde o intervalo de observação é "suficientemente grande". Na verdade, essas medidas são aproximações do comportamento do sistema para um intervalo finito e longo com relação ao intervalo entre ocorrência de eventos no sistema. Entretanto, várias aplicações exigem o cálculo de medidas durante um intervalo "curto", onde medidas em estado estacionário não são uma boa aproximação para o comportamento do sistema durante o período de observação. Neste caso medidas *transientes* tem que ser obtidas.

A análise transiente tem sido usada para estudar o comportamento dinâmico de modelos complexos de sistemas de comunicação. Por exemplo, em [24] um controle de congestionamento para uma rede de pacotes foi modelado e a medida de interesse é baseada nas probabilidades de estado no tempo t . Ren e Kobayashi [20] e Tanaka *et al* [23] estudaram a solução transiente de um modelo de fluido de filas de um comutador ATM. A medida de interesse é o tamanho da fila do comutador no tempo t . A análise é baseada numa equação diferencial parcial para a distribuição transiente do tamanho da fila.

Outras aplicações incluem o cálculo de descritores de modelos Markovianos de tráfego multimídia. Heyman e Lakshman [13] mostraram que modelos Markovianos de tráfego podem ser usados para prever com precisão medidas de performance, mesmo que o tráfego gerado por esses modelos não possua a propriedade de dependência de longa duração. Este resultado aponta como consequência a importância de se calcular descritores baseados em estatísticas de segunda ordem, tais como o índice de dispersão para contagem e a autocovariância, a partir de modelos Markovianos de fontes. Esses cálculos são fundamentados em análise transiente (e.g. [22, 2, 15]).

Modelos de Markov com recompensa são aqueles onde é atribuída uma taxa de recompensa a cada estado e/ou uma recompensa fixa (impulso) a uma transição. Neste caso, se ao estado i do modelo é associada uma taxa de recompensa r_i e uma recompensa tipo impulso p_{ij} é associada à transição $i \rightarrow j$, então o modelo adquire um valor r_i por cada unidade de tempo em que permanece em i e um valor p_{ij} a cada transição de i para j . As medidas de interesse estão relacionadas à recompensa total acumulada em um dado intervalo de tempo. Estes modelos, chamados de uma maneira geral de modelos de *performabilidade* (e.g. [16, 5]), são extremamente úteis na análise do desempenho e confiabilidade de sistemas. Por exemplo, na área de redes de computadores, se considerarmos uma fonte de células modelada por uma cadeia de Markov tal que cada estado representa uma taxa de transmissão de células por unidade de tempo, a recompensa total acumulada em $(0, t)$ é igual ao número de células transmitidas no intervalo.

Usando conceitos de modelagem de performabilidade, o trabalho de [10] apresentou um método para calcular a distribuição da recompensa total acumulada durante um intervalo $(0, t)$ em modelos com recompensa do tipo taxa e impulso. Os resultados foram usados para calcular medidas de interesse tal como a probabilidade transiente do *buffer* de um comutador ultrapassar um certo valor em um *buseriod*, usando modelos de fonte incluindo aqueles com taxas bem distintas. Entretanto, esse método não permite que se incluam limites para o valor da recompensa acumulada. Como a recompensa acumulada (sem limites) pode adquirir valores negativos e positivos ilimitados, o resultado é uma

- [14] B. Igelnik, Y. Kogan, V. Krivan, and D. Mitra. A new computational approach for stochastic fluid models of multiplexers with heterogeneous sources. *Queueing Systems*, 20:85-116, 1995.
- [15] R.M.M. Leão, E. de Souza e Silva, and S. Lucena. A set of tools for traffic modeling, analysis and experimentation. In *Proceedings of the 11th International Conference on Modelling Techniques and Tools for Computer Performance Evaluation*, volume Lecture Notes in CS 1786, pages 40-55. Springer, Março 2000.
- [16] J.F. Meyer. Performability evaluation: Where it is and what lies ahead. In *Proceedings of the IEEE International Computer Performance and Dependability Symposium (IPDS'95)*, pages 334-343. 1995.
- [17] Debasis Mitra. Stochastic theory of a fluid model of producers and consumers coupled by a buffer. *Adv. Appl. Prob.*, 20:646-676, 1988.
- [18] H. Nabil and B. Sericola. Performability analysis: a new algorithm. *IEEE Trans. on Computers*, 45(4):491-494, 1996.
- [19] K.R. Pattipati, Y. Li, and H.A.P. Blom. A unified framework for the performability evaluation of fault-tolerant computer systems. *IEEE Trans. on Computers*, 42(3):312-326, 1993.
- [20] Q.R. Ren and H. Kobayashi. Transient solutions for the buffer behavior in statistical multiplexing. *Performance Evaluation*, 23:65-87, 1995.
- [21] Bruno Sericola. Transient analysis of stochastic fluid models. Technical Report 3152, INRIA, 1997.
- [22] Paul Skelly, Mischa Schwartz, and Sudhir Dixit. A Histogram-Based Model for Video Traffic Behavior in an ATM Multiplexer. *IEEE/ACM Transactions on Networking*, 1(4):445-459, 1993.
- [23] T. Tanaka, O. Hashida, and Y. Takahashi. Transient analysis of fluid model for ATM statistical multiplexer. *Performance Evaluation*, 23:145-162, 1995.
- [24] H.R. van As. Transient analysis of Markovian queueing systems and its application to congestion control modeling. *IEEE Journal on Selected Areas in Communications*, SAC-4(6):891-904, 1986.