

# Um modelo para geração de carga de servidores Web utilizando classificação de conteúdo

Carlos Marcelo Pedroso\*, Keiko Fonseca

Pós Graduação em Engenharia Elétrica e Informática Industrial  
Universidade Tecnológica Federal do Paraná - UTFPR  
Avenida Sete de Setembro 3165, Curitiba, Paraná, Brasil

c.pedroso@pucpr.br, keiko@cpgei.cefetpr.br

**Abstract.** *Performance modeling is an important issue to capacity planning and overload control of Web servers. This paper presents a model for workload generation of Web servers based on the classification of the server files in groups. The proposed model was validated with data of 1998 World Cup Web server and with more recent data from IRCache proxy servers. It is also presented a computational simulation showing the model capability to generate self-similar traffic. The proposed model show good perspectives in future research, mainly in capacity planning and techniques to improve Web servers performance.*

**Resumo.** *A modelagem de desempenho é um tópico importante para realização do planejamento de capacidade e controle de carga de servidores Web. Neste artigo é proposto um modelo para geração de carga de servidores Web através da classificação dos arquivos transmitidos pelo servidor em grupos. O modelo proposto foi validado através de dados do servidor da copa do mundo de 1998 e de dados mais recentes de servidores de proxy do projeto IRCache. É apresentada uma simulação computacional do modelo que mostra a capacidade de geração de tráfego com perfil auto-similar. O modelo proposto mostra boas perspectivas para pesquisa de novos métodos de dimensionamento e técnicas para a melhoria de desempenho de servidores.*

## 1. Introdução

O desenvolvimento de modelos que permitam representar o desempenho de servidores Web é uma importante área de pesquisa. Bons modelos permitem realizar previsões acuradas sobre métricas de desempenho [Cao et al. 2003] e, a partir destas, por exemplo, o planejamento da capacidade do servidor. Entre as características desejadas para um modelo estão a sua simplicidade, obtida ao se restringir somente aos aspectos que influenciem significativamente no comportamento que se deseja analisar, e a sua tratabilidade, geralmente associada à complexidade de se gerar resultados analisáveis a partir do modelo.

A carga de tráfego da Internet é dominada pelo protocolo HTTP (*Hyper Text Transfer Protocol* [Fielding et al. 1999]), que foi identificado em medições do backbone da Sprint como sendo a classe de aplicativos mais utilizado na Internet com uma

---

\* Atualmente o autor está também com a Pontifícia Universidade Católica do Paraná (PUCPR)

faixa de 31% a 59% do total de bytes transmitidos [Cao et al. 2004]. Nos últimos 10 anos (e possivelmente ainda no futuro), geradores de tráfego para sintetizar o tráfego Web são um componente essencial em virtualmente toda a simulação da Internet. Esta popularidade possivelmente deve ampliar-se devido ao fato de que o protocolo HTTP está sendo utilizado como interface para entregar conteúdos para aplicativos mais especializados.

Em se tratando de planejamento de capacidade de servidores Web, os modelos denominados de *modelos de desempenho* devem ser precisos para capturar o comportamento real do servidor: modelos inadequados podem resultar em super ou sub-dimensionamento da capacidade do sistema.

Nos últimos anos, muito esforço tem sido realizado para desenvolver modelos que descrevem apropriadamente o tráfego de redes de computadores [Crovella and Bestavros 1995], [Leland et al. 1994], [Willinger and Park 2000], [Muscariello et al. 2004], [Nossonson and Attiya 2004] e [Gong et al. 2005]. Um dos resultados apresentados por estes trabalhos mostra que o modelo auto-similar descreve apropriadamente o tráfego agregado observado na saída de servidores Web, sendo necessário o desenvolvimento de novos modelos para representação do tráfego atual.

Um modelo de tráfego para servidores Web pode considerar características de comportamento do usuário ou não. Por exemplo, modelos baseados na teoria de filas, como o proposto por [Cao et al. 2003] (modelo M/G/1/K\*PS) não consideram o comportamento do usuário nem o conteúdo do servidor. Por outro lado, em modelos como propostos por [Barford and Crovella 1998], [Nuzman et al. 2002] ou por [Muscariello et al. 2004], o conhecimento do comportamento do usuário é aplicado para representação do tráfego real.

Neste artigo é apresentado um modelo de desempenho para servidores Web utilizando a classificação semântica do conteúdo transmitido pelo servidor e o comportamento do usuário. O modelo proposto baseia-se no fato observado de que o tamanho dos objetos de cada classe semântica não segue, via de regra, distribuições de cauda pesada e sim distribuições de decadência exponencial ou sub-exponencial. Com isto, evita-se as características pouco desejáveis para análise exibidas por distribuições de cauda pesada, como a não convergência do desvio padrão. Assim sendo, o uso da classificação semântica permite estabelecer um modelo mais tratável para avaliação do desempenho de servidores Web.

A nova abordagem de modelagem aqui proposta contribui em 3 pontos, ao possibilitar a geração de carga sintética para simulações computacionais envolvendo servidores Web, a análise de desempenho do servidor através de cadeias de Markov e a aplicação deste modelo na pesquisa de novas técnicas relacionadas a melhoria de desempenho de servidores (por exemplo, algoritmos de descarte de informação em áreas de *cache*).

A validação do modelo foi realizada a partir da análise estatística de dados de vários servidores que possuem seus arquivos de *log* disponíveis para uso em pesquisa. Foram analisados os arquivos de *log* do servidor da copa do mundo de 1998 [Arlitt and Jin 2000] e de servidores de Proxy do IRCache, um projeto do NLANR - *National Laboratory for Applied Network Research* [Wessels and Claffy 1996]. Os servidores cache do NLANR registram os acessos de toda a Internet e estão geograficamente distribuídos pelo Estados Unidos da América para balanceamento de carga e de forma a atender todos os

continentes. Desta forma, as milhares de requisições diárias destes servidores refletem o comportamento típico de acessos a servidores Web em geral, pois não se restringem a um grupo restrito de clientes ou a aplicações específicas.

Neste artigo utilizamos o conceito de *sessão* para definir o período de tempo onde um cliente transfere uma página e em seguida outros arquivos referenciados por esta página, que podem ser chamados de *objetos* embutidos. O termo *objeto* refere-se à resposta enviada por um servidor Web.

Este artigo está estruturado da seguinte forma. A Seção 2 mostra os trabalhos relacionados. A Seção 3 descreve o modelo baseado na classificação semântica dos objetos transmitidos por servidores Web. A Seção 4 apresenta a caracterização do sistema utilizando dados coletados de sistemas reais. A Seção 5 mostra os resultados de uma simulação computacional do modelo proposto. As conclusões e trabalhos futuros são apresentados na Seção 6.

## 2. Trabalhos Relacionados

Os geradores de tráfego Web em uso atualmente são normalmente baseados nos dados de dois projetos pioneiros que introduziram modelos que procuravam capturar características comportamentais de clientes e servidores: os estudos de Mah [Mah 1997] e Barford e Crovella [Barford and Crovella 1998]. Geradores de tráfego utilizando os conceitos propostos foram construídos como parte do simulador amplamente utilizado NS-2 [Breslau et al. 2000].

O modelo chamado SURGE (*Scalable URL Reference Generator*), proposto por [Barford and Crovella 1998], é um dos mais citados na literatura para geração de tráfego Web. O SURGE é baseado em um autômato *ON-OFF* que captura características comportamentais do sistema; será descrito em detalhes na Seção 3 por tratar-se da base utilizada para o desenvolvimento do modelo proposto neste artigo.

[Hernandez-Campos et al. 2003] apresenta um modelo empírico do tráfego gerado pelo protocolo HTTP. Ao invés de confiar nos *logs* de servidores ou clientes, seu método é baseado nos pacotes coletados durante a conversação HTTP. Através da análise de tráfego, Hernández-Campos et al. determinaram estatísticas e distribuições para elementos de alto nível, como tamanho de arquivos transferidos, o número de arquivos por sessão e o comportamento do usuário. Estas quantidades formam um modelo que pode ser utilizado em simulações para imitar o tráfego gerado por aplicações.

Uma ferramenta chamada Harpoon, projetada para geração de pacotes no nível de fluxo IP é apresentada por [Sommers and Barford 2004]. A ferramenta gera pacotes TCP e UDP que possuem as mesmas características medidas em roteadores reais em termos de byte, pacote, temporal e espacial. A ferramenta se distingue de outras pela sua capacidade de auto configuração a partir de arquivos de *log*. A análise se diferencia de outras tentativas de montagem modelos utilizando o conceito de fluxos por combinar distribuições empíricas com características que podem ser medidas em roteadores de redes reais. O modelo propõe a geração de fluxos baseado em um nível de duas hierarquias. Sessões são formadas por uma série de conexões separadas por um tempo de duração. Conexões são formadas pela transferência de objetos com intervalo de tempo caracterizado por distribuições de probabilidade. O tamanho dos arquivos transferidos possui uma

distribuição de cauda pesada e um modelo ON-OFF gera características auto-similares no nível de pacotes.

[Cao et al. 2004] propõe um modelo que representa o tráfego Web como uma coleção de conexões TCP independentes, cada qual caracterizada por valores das seguintes variáveis: tempo entre conexões, RTT (*Round Trip Time*) para o cliente, RTT para o servidor, número de trocas requisição/resposta, intervalo de tempo entre trocas, tamanho de requisições individuais, tamanho de respostas individuais e atraso do servidor.

O modelo proposto na Seção 3 difere das abordagens existentes, propondo o agrupamento dos objetos transmitidos pelo servidor Web em classes. Isto resulta em uma modelagem precisa do comportamento do sistema que permite a geração de tráfego sintético para simulações, com boas possibilidades analíticas e novas perspectivas para o desenvolvimento de técnicas associadas à melhoria de desempenho.

### **3. Um modelo composto para servidores Web utilizando classificação semântica de conteúdo**

O modelo é baseado no SURGE, proposto por [Barford and Crovella 1998]. O SURGE utiliza um autômato ON-OFF para capturar o comportamento do usuário e distribuições de probabilidade para caracterizar o tamanho dos arquivos armazenados no servidor. Quando o sistema está no estado ON, a sessão está ativa enviando os objetos requisitados na sessão. O intervalo de tempo entre os arquivos enviados durante a sessão é denominado de tempo *active-off*. O tamanho dos arquivos e o número de referências em uma sessão de usuário também são utilizados. As principais variáveis deste modelo são:

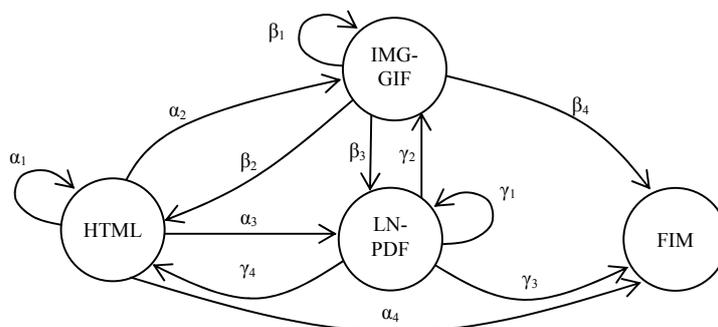
- *Tempo OFF*: É o tempo que o usuário permanece pensando. Normalmente modelado por uma distribuição de Pareto;
- *Tamanho dos arquivos*: É o tamanho dos objetos transmitidos. Normalmente modelado por uma distribuição de Pareto;
- *Número de referências*: Número de arquivos transmitidos em uma sessão de usuário. Também modelado normalmente por uma distribuição de Pareto;
- *Tempo active-off*: É o intervalo de tempo entre os arquivos transmitidos em uma sessão de usuário. Modelada pela distribuição de Weibull;
- *Popularidade*: É o número relativo de acessos realizados a um arquivo individual. A popularidade de arquivos em servidores Web segue, via de regra, a lei de Zipf. A lei de Zipf argumenta que se os arquivos forem ordenados do mais popular para o menos popular, então o número de referências a um arquivo tende a ser inversamente proporcional a sua posição na classificação  $r$ , ou  $P = kr^{-1}$  para uma constante positiva qualquer  $k$ ;
- *Localidade temporal*: A localidade temporal indica que, uma vez tendo sido requisitado um arquivo, a probabilidade de que ele seja novamente requisitado no futuro aumenta. Para o estudo desta variável os acessos são armazenados em uma estrutura de pilha. A distância entre os acessos nesta pilha são estudados e modelados comumente com uma distribuição Lognormal.

Trabalhos importantes reportam que o tamanho dos arquivos transmitidos por servidores Web podem ser modelados por distribuições de cauda pesada, por exemplo, a distribuição de Pareto [Crovella and Bestavros 1995] [Barford and Crovella 1998]. Existe

grande dificuldade no tratamento matemático deste tipo de distribuição devido a sua grande variabilidade [Willinger and Park 2000].

Neste artigo nós propomos que os arquivos transmitidos pelo servidor sejam agrupados em classes de arquivos para que depois as classes sejam estudadas para determinar suas características, como a distribuição de probabilidade do tamanho dos arquivos e o tempo de permanência em cada classe. A proposta desta abordagem foi apresentada anteriormente por [Pedroso et al. 2005].

No modelo proposto, a atividade do usuário é modelada por um autômato finito como no SURGE. O estado ON será considerado como o tempo gasto em uma sessão ativa. Durante o tempo de uma sessão ativa serão produzidas requisições para transferência de diversos arquivos. Os arquivos transmitidos serão classificados em classes semânticas previamente identificadas com auxílio dos arquivos de *log* do servidor. As demais variáveis do SURGE, como o popularidade, localidade temporal e tempo *active-off*, continuam válidas. É razoável supor que a chegada de novas sessões de usuários constitui-se de um processo de Poisson [Roberts 2000].



**Figura 1. Diagrama hipotético de classes semânticas para transmissão de objetos durante o estado ON**

Desta maneira, a cada sessão ativa de usuário, após a primeira requisição ser produzida, uma seqüência de arquivos será transmitida. Estas várias requisições serão produzidas pelo próprio programa cliente (*browser*) automaticamente para transferir todos os arquivos necessários para apresentar o conteúdo para o usuário. A Figura 1 ilustra um diagrama de estados hipotético utilizando uma classificação com três classes de arquivo: arquivos em formato de hipertexto (HTML), imagens em formato GIF (IMG-GIF) e notas de aula em formato PDF (LN-PDF).

A Figura 1 deve ser interpretada da seguinte maneira: Um acesso inicial a um arquivo pertencente à classe HTML será seguida por uma transmissão de um novo arquivo pertencente a outra classe de tráfego. No caso específico deste exemplo,  $\alpha_1$ ,  $\alpha_2$  e  $\alpha_3$  representam as probabilidades do próximo arquivo transmitido pertencer respectivamente às classes HTML, IMG-GIF ou LN-PDF.  $\alpha_4$  é a probabilidade de não haverem mais requisições de transmissão de arquivos para esta sessão, iniciando-se um período OFF. A relação  $\sum_{j=1}^4 \alpha_j = 1$  deve ser sempre verdadeira. O mesmo ocorre para as probabilidades de transição a partir de cada uma das classes de arquivos do modelo. No exemplo,  $\sum_{j=1}^4 \beta_j = 1$  e  $\sum_{j=1}^4 \gamma_j = 1$  para as classes IMG-GIF e LN-PDF.

De modo genérico, pode-se representar o diagrama de estados da Figura 1 através de uma matriz quadrada  $P$  com as probabilidades de transição de estados, dada por

$$P = \begin{bmatrix} p_{00} & p_{01} & \dots & p_{0n} \\ p_{10} & p_{11} & \dots & p_{1n} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1 \end{bmatrix} \quad (1)$$

onde  $n$  representa o número de classes de arquivos, incluindo-se o estado representando o fim da sessão. Na montagem da matriz, o estado  $n$  é o estado que representa o fim da sessão. Para um estado qualquer  $k$ ,  $\sum_{j=0}^n P_{kj} = 1$ . Este diagrama de transição de estados representa a sucessão de classes de arquivos transmitidos, e não o tamanho dos arquivos transmitidos em cada estado. O tamanho dos arquivos transmitidos em cada estado deve ser determinado através da análise dos arquivos do servidor. A determinação da distribuição de probabilidade de cada classe de arquivos tipicamente encontrada na Internet é mostrada na seção 4.

A análise do arquivo de *log* revela a quantidade de classes de arquivos do servidor. A separação em classes pode facilitar a tarefa de modelagem do tamanho dos objetos, uma vez que a classificação pode ser realizada de acordo com diversos critérios. Neste trabalho, a classificação se fez através da extensão do arquivo para a maioria das classes. Em dois casos foi necessário um critério mais elaborado que considera a extensão e o tamanho do arquivo, de modo a obter-se uma melhor aderência a distribuições clássicas de probabilidade. No entanto, outros critérios poderiam ser utilizados para a separação das classes de arquivos.

Uma matriz de transições de probabilidade entre as classes de arquivos para uma dada sessão também pode ser extraída dos arquivos de *log*, de modo a completar o modelo.

#### 4. Levantamento das classes de arquivos e da matriz de transição de estados

Foram desenvolvidos programas para extração de dados sobre as sessões de usuário utilizando uma heurística semelhante àquela utilizada por [Mah 1997] e [Hernandez-Campos et al. 2003], onde uma sessão é identificada a partir do primeiro acesso de um endereço IP ao servidor e procurando as requisições subseqüentes realizadas por este mesmo endereço, inferindo que estas refletem o comportamento da sessão. Quando não há mais atividade do usuário dentro de um certo tempo limite, a sessão é considerada encerrada. Considerou-se um tempo limite de 120 segundos para concluir sobre o encerramento da sessão.

Os resultados aqui apresentados foram obtidos através da análise de coletas de dados dois sistemas. O primeiro sistema em estudo é o IRCache, que é um projeto do NLANR - *National Laboratory for Applied Network Research* [Wessels and Claffy 1996]. Os servidores cache do NLANR registram os acessos de toda a Internet e estão geograficamente distribuídos pelo Estados Unidos da América para balanceamento de carga e de forma a atender todos os continentes. Desta forma, as milhares de requisições diárias destes servidores refletem o comportamento típico de acessos a servidores Web em geral, pois não se restringem a um grupo restrito de clientes ou a aplicações específicas.

O segundo sistema em estudo é o servidor Web da Copa do Mundo de 1998, já estudado por [Arlitt and Jin 2000]. Neste artigo, este sistema será referenciado por ser-

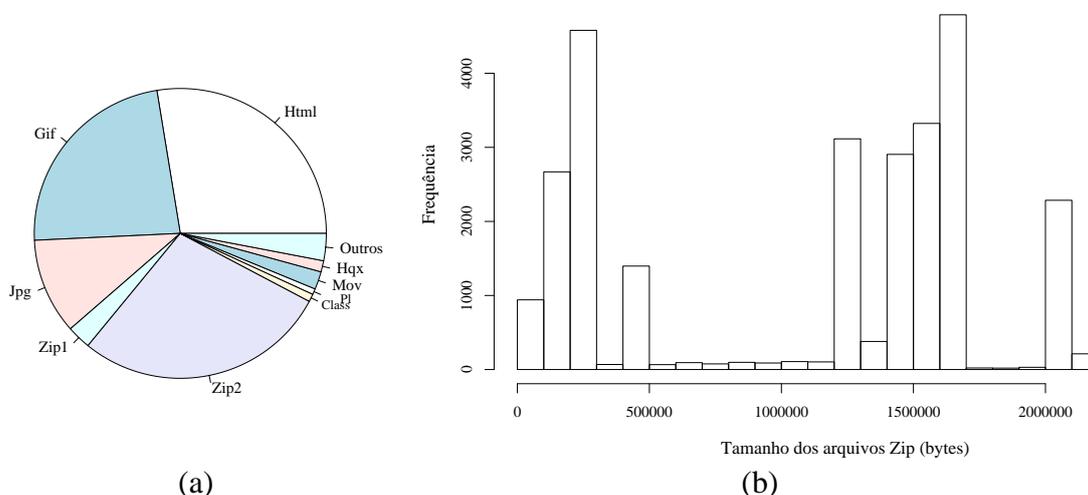
**Tabela 1. Coleções de dados em estudo**

Amostra	Local	Data	Número de linhas
IRCache	Nova York	28/Novembro/2004	366.234
IRCache	Nova York	29/Novembro/2004	20.530
IRCache	Palo Alto	29/Novembro/2004	140.636
WC98	Dia 37	31/Maio/1998	5.586.176
WC98	Dia 52	15/Junho/1998	7.000.000
WC98	Dia 73	6/Julho/1998	7.000.000

vidor WC98 (*World Cup 98*). Embora os dados do WC98 sejam relativamente antigos, o fato de já existirem estudos publicados sobre este sistema possibilita que os resultados obtidos sejam confrontados com outros estudos. Foram estudados 3 dias diferentes totalizando aproximadamente 20 milhões de requisições. Os arquivos analisados no trabalho são apresentados na Tabela 1.

#### 4.1. Identificação das principais classes de tráfego

A observação dos *logs* mostrou as principais classes de tráfego em cada um dos sistemas em estudo. A extração das classes de objetos transmitidos foi realizada observando-se principalmente a extensão dos arquivos transmitidos. Para identificação da contribuição da classe foi levantado o volume de tráfego efetivamente transmitido em cada classe de tráfego.



**Figura 2. (a) Volume de tráfego para as principais classes de arquivo transmitidos identificados nos arquivos de *log* do servidor da WC98 (b) Histograma do tamanho dos arquivos da classe ZIP do servidor da WC98**

No servidor WC98 verifica-se a existência de poucas classes de arquivos que contribuíram significativamente na formação do tráfego de saída do servidor, o que permite a construção de um modelo bastante simplificado. Os tipos mais importantes foram *HTML*, *JPEG*, *GIF*, *ZIP1* e *ZIP2*, como mostrado na Figura 2(a). Os arquivos com extensão ZIP foram divididos em duas categorias, ZIP1 e ZIP2, porque a observação do histograma da classe ZIP mostrou dois comportamentos distintos (Figura 2(b)). A classe ZIP foi separada em duas classes de acordo com o tamanho do arquivo transmitido. Arquivos maiores

que  $10^6$  bytes foram classificados como ZIP2 e os outros como ZIP1. Isto é necessário para evitar o uso de distribuições de cauda pesada para caracterizar a classe ZIP, o tamanho de arquivo resultante nas classes ZIP1 e ZIP2 puderam ser caracterizados pela distribuição Lognormal<sup>1</sup>. O volume de tráfego por tipo de arquivo está de acordo com o reportado em [Arlitt and Jin 2000].

**Tabela 2. Percentual observado em relação ao volume total trafegado em bytes para as principais classes de arquivos transmitidos no sistema IRCache**

Classe	Amostra 1	Amostra 2	Amostra 3	Média
GIF1	0,93	1,26	0,96	1,05
GIF2	4,91	6,60	5,02	5,51
HTML	22,11	15,71	20,94	19,59
JPEG	20,88	25,86	10,25	19,00
MPEG	4,22	3,7	3,11	3,68
OCTET-STR	13,85	28,07	20,15	20,69
Outros	33,1	18,8	39,57	30,49

No sistema IRCache foram identificadas as seguintes classes principais: *HTML*, *GIF1*, *GIF2*, *JPG*, *MPEG*, *OCTET-STREAM*. A participação de cada classe na formação do tráfego pode ser visto na Tabela 2. Os arquivos com extensão GIF foram separados em duas classes para a realização do teste de aderência à distribuições conhecidas. O critério utilizado foi o mesmo do caso anterior com a classe ZIP do servidor WC98. Neste caso, os arquivos GIF com tamanho menor que 350 bytes foram classificados como GIF1 e os arquivos com tamanho maior ou igual a 350 bytes foram classificados como GIF2.

#### 4.2. Identificação da matriz de probabilidade transição de estados das classes de arquivos

A Tabela 3 mostra a matriz de probabilidade de transição de classes obtidos a partir dos arquivos de *log* dos dias 37, 52 e 73 do servidor WC98. Os dados sobre o início e fim das sessões foram extraídos utilizando-se a heurística descrita na seção 4. Os dados foram recolhidos de modo a oferecer um estimador para a Equação 1.

Outro fato relevante a ser observado é que a matriz de probabilidade de transição entre as classes de arquivo permanece constante ao longo dos três dias observados, o que revela que esta matriz descreve o comportamento do acesso aos arquivos do servidor em cada sessão de usuário, e pode ser utilizada para avaliação de desempenho e geração de carga.

O resultado obtido para o sistema IRCache não é apresentado por este sistema tratar-se de um *Proxy* que agrupa requisições à diversos servidores Web. Neste caso, a caracterização da matriz de probabilidade de transição de estados não estaria relacionado a nenhum servidor Web em particular e sim à características mais gerais de sistemas Web.

#### 4.3. Caracterização do tamanho dos objetos transmitidos em cada classe de arquivo

Foi realizada a caracterização do tamanho do arquivo transmitido em nas principais classes identificadas na seção 4.1.

<sup>1</sup>Função densidade de probabilidade:  $f(x) = \frac{1}{x\sigma\sqrt{2\pi}} e^{-(\ln x - \mu)^2 / 2\sigma^2}$

**Tabela 3. Matriz de probabilidade transição de estados para o servidor da WC98**

	HTML	JPEG	GIF	ZIP1	ZIP2	MOV	HQX	CLASS	PL	OUTROS	FIM
HTML	0,3105	0,0881	0,5244	0,0003	0,0002	0,0000	0,0000	0,0045	0,0002	0,0219	0,0495
JPEG	0,1279	0,3145	0,5075	0,0014	0,0002	0,0003	0,0000	0,0053	0,0002	0,0089	0,0333
GIF	0,0657	0,0346	0,8634	0,0001	0,0002	0,0000	0,0000	0,0110	0,0002	0,0064	0,0180
ZIP1	0,1539	0,0258	0,1658	0,2601	0,0159	0,0000	0,0001	0,0008	0,0025	0,0117	0,3630
ZIP2	0,0908	0,0122	0,1142	0,0091	0,1323	0,0000	0,0025	0,0008	0,0039	0,0039	0,6298
MOV	0,1234	0,1428	0,1076	0,0014	0,0014	0,3094	0,0000	0,0000	0,0000	0,0093	0,3043
HQX	0,0815	0,0163	0,1068	0,0036	0,0724	0,0000	0,1757	0,0000	0,0054	0,0126	0,5253
CLASS	0,0661	0,0202	0,8622	0,0000	0,0000	0,0000	0,0000	0,0217	0,0000	0,0042	0,0251
PL	0,0794	0,0451	0,3973	0,0008	0,0023	0,0000	0,0000	0,0002	0,2850	0,0067	0,1828
OUTROS	0,1830	0,0509	0,6424	0,0005	0,0003	0,0000	0,0000	0,0082	0,0002	0,0380	0,0761

A análise dos objetos transmitidos pelo sistema IRCache mostrou que o tamanho do arquivo transmitido em cada classe pode ser modelado na maioria dos casos pela distribuição Lognormal. Para realizar o teste de aderência, a distribuição de probabilidade acumulada da amostra foi comparada com distribuições clássicas de probabilidade. A Figura 3 mostra uma comparação entre a distribuição acumulada teórica e a distribuição amostral para as principais classes. O eixo horizontal ( $x$ ) representa o tamanho do arquivo transmitido e o eixo vertical ( $P(X \leq x)$ ) indica a probabilidade acumulada. Todas as classes de tráfego aderiram a distribuição Lognormal, à exceção da classe GIF1, que aderiu a distribuição Normal<sup>2</sup>.

Este resultado previne o problema relatado recentemente por [Gong et al. 2005] sobre a falha dos testes de aderência e dificuldades de caracterização quando a distribuição alvo é do tipo cauda pesada, como ocorreria se os arquivos fossem agrupados em uma única classe. Mesmo para as classes menos importantes e não listadas na Tabela 2 o tamanho dos arquivos transmitidos pôde se caracterizado por uma distribuição Lognormal. Isto contrasta com a distribuição de cauda pesada reportada na literatura observada quando os arquivos são tratados em conjunto.

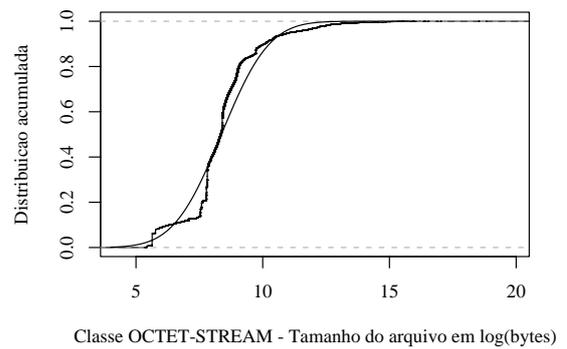
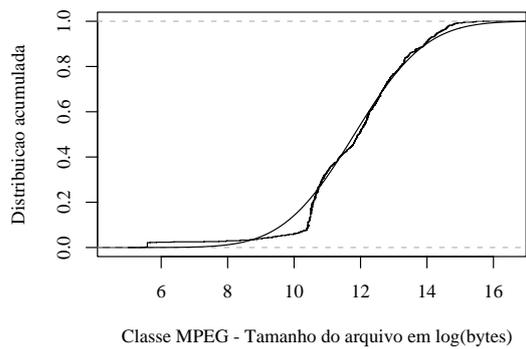
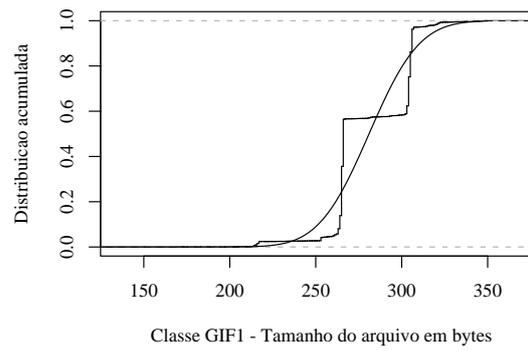
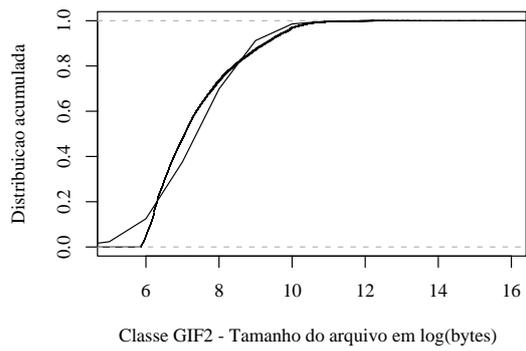
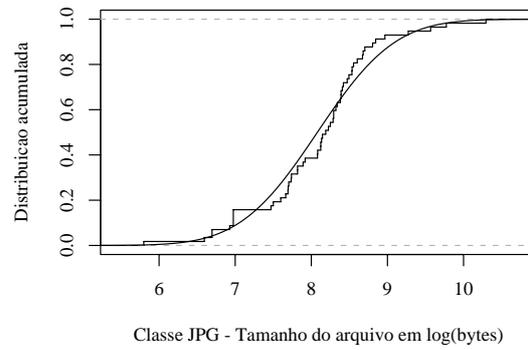
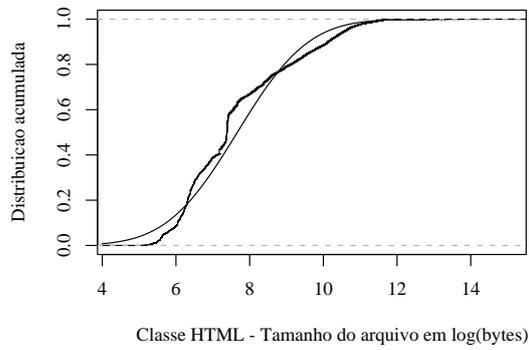
A caracterização do tamanho dos arquivos em cada classe foi repetida para o servidor WC98. A Figura 4 mostra a distribuição acumulada empírica das principais classes do servidor WC98 comparada com a distribuição Lognormal. A linha contínua representa a distribuição teórica e os pontos representam a amostra. Verifica-se visualmente uma boa aderência dos dados à distribuição Lognormal. O sumário para estatísticas básicas dos arquivos analisados da WC98 são apresentados na Tabela 4(a). Esta caracterização está consistente com o estudo similar realizado por Arlitt e Jin em [Arlitt and Jin 2000].

#### 4.4. Caracterização do tempo de permanência em cada classe

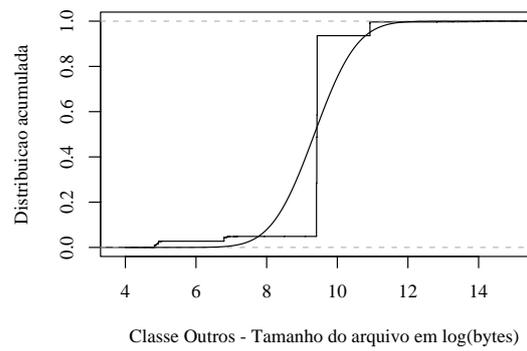
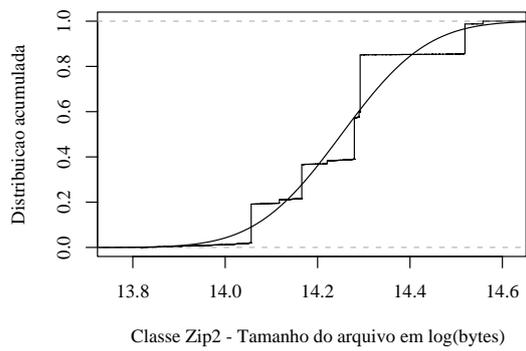
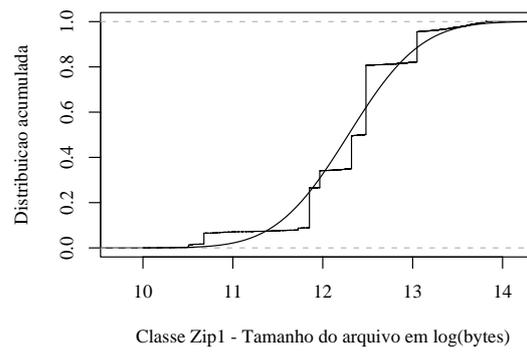
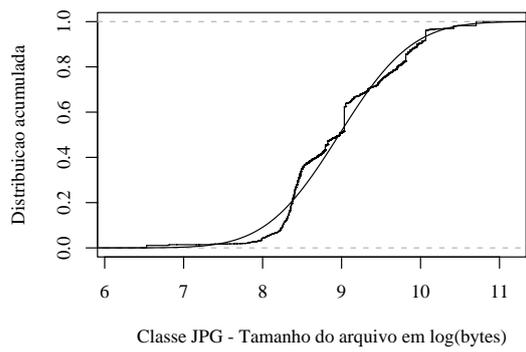
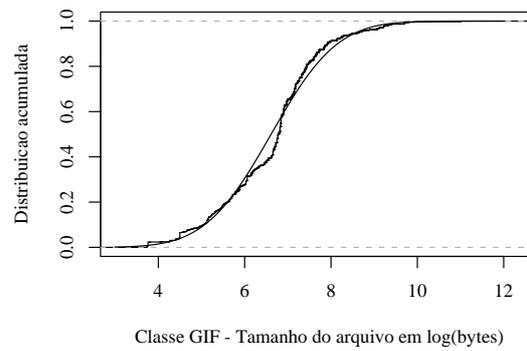
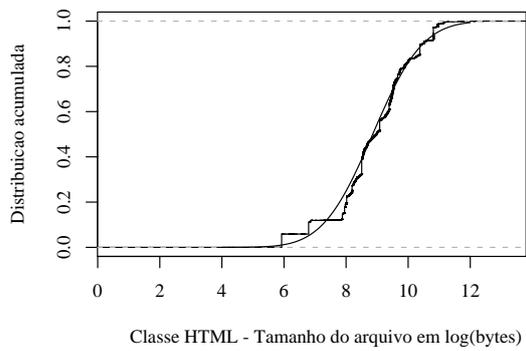
Outro resultado importante foi o tempo de permanência em cada classe. Descobrimos que este tempo pode ser descrito por uma distribuição de Weibull<sup>3</sup>. O tempo de permanência na classe é resultado do tempo necessário para a transferência do arquivo somado com o tempo de processamento do servidor e do cliente. O tamanho do arquivo de cada classe

<sup>2</sup>Função densidade de probabilidade:  $f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-(x-\mu)^2/2\sigma^2}$

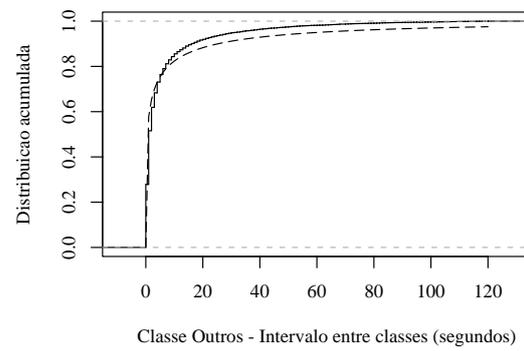
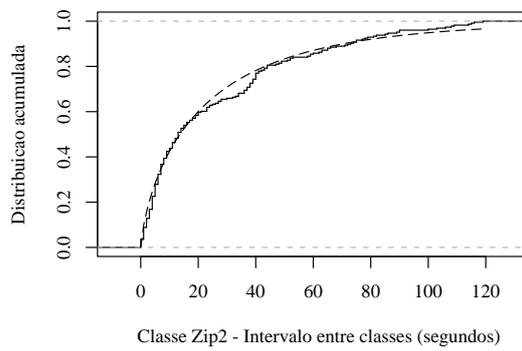
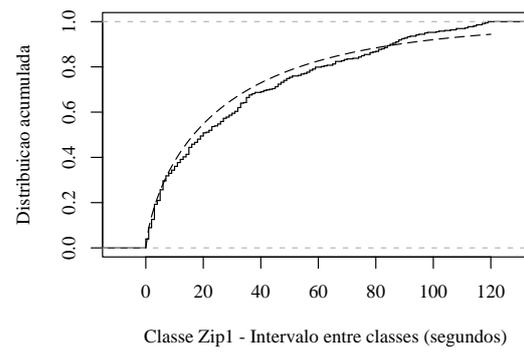
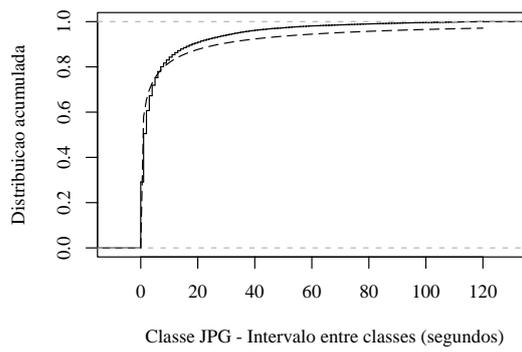
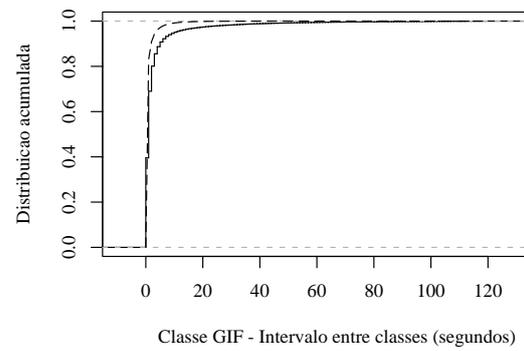
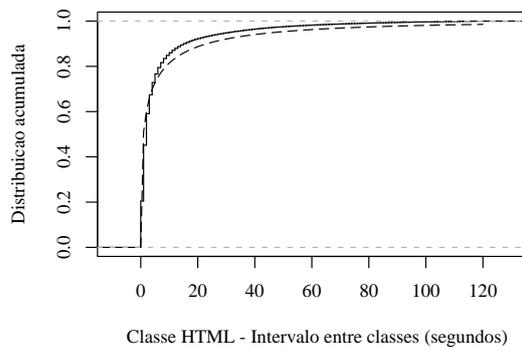
<sup>3</sup>Função densidade de probabilidade:  $f(x) = \frac{bx^{b-1}}{a^b} e^{-(x/a)^b}$



**Figura 3. Distribuição de probabilidade teórica e distribuição amostral para cada uma das classes de arquivos do sistema IRCache. A classe GIF1 é comparada com a distribuição Normal e as demais com a distribuição Lognormal**



**Figura 4. Distribuição acumulada para o tamanho dos arquivos das principais classes de arquivos do servidor WC98 comparado com a distribuição Lognormal**



**Figura 5. Distribuição acumulada de probabilidade para o tempo de permanência na classe HTML comparada com a distribuição de Weibull (linha pontilhada)**

**Tabela 4. (a) Sumário de estatísticas e ajuste de distribuição para o tamanho dos arquivos pertencentes às principais classes de arquivos do servidor WC98 e (b) Distribuições ajustadas para o tempo de transmissão de arquivos em cada classe para o servidor WC98**

Classe	Modelo	Parâmetros
GIF	Lognormal	$\mu = 1.646, \sigma = 4.100$
HTML	Lognormal	$\mu = 13.550, \sigma = 16.284$
JPEG	Lognormal	$\mu = 10.210, \sigma = 8.142$
ZIP1	Lognormal	$\mu = 254.100, \sigma = 159.773$
ZIP2	Lognormal	$\mu = 1.564.000, \sigma = 233.676$
MOV	Lognormal	$\mu = 1.441.000, \sigma = 285.251$
PL	Lognormal	$\mu = 64.720, \sigma = 212.349$
HQX	Lognormal	$\mu = 2.236.000, \sigma = 451.173$
CLASS	Lognormal	$\mu = 4.627, \sigma = 994$
OUTROS	Lognormal	$\mu = 18.210, \sigma = 68.428$

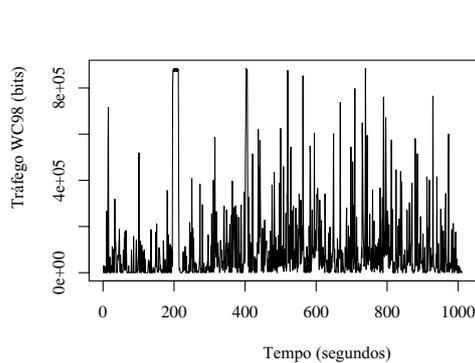
(a)

Classe	Modelo	Parâmetros
GIF	Weibull	$b = 0,43, a = 0,26$
HTML	Weibull	$b = 0,37, a = 2,46$
JPEG	Weibull	$b = 0,29, a = 1,62$
ZIP1	Weibull	$b = 0,72, a = 27,58$
ZIP2	Weibull	$b = 0,73, a = 22,52$
MOV	Weibull	$b = 1,36, a = 53,97$
PL	Weibull	$b = 0,50, a = 9,12$
HQX	Weibull	$b = 0,36, a = 2,83$
CLASS	Weibull	$b = 0,28, a = 0,62$
OUTROS	Weibull	$b = 0,30, a = 1,64$

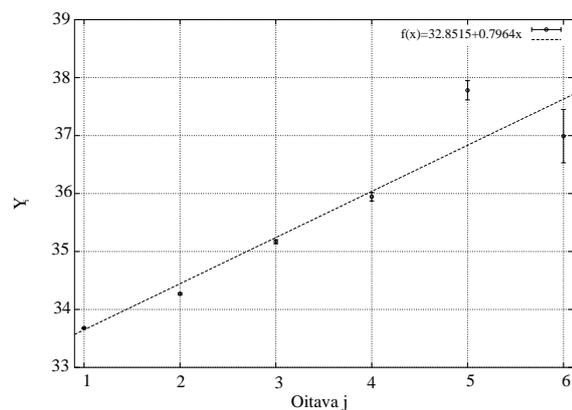
(b)

e o tempo necessário para transmiti-lo são caracterizados por distribuições diferentes. No entanto, a caracterização do tempo de permanência em cada classe não é parte do modelo aqui proposto; esta caracterização será valiosa no estudo analítico utilizando-se cadeias de Markov para o estudo de desempenho do sistema. Este estudo será tema de trabalhos futuros.

A Figura 5 mostra a distribuição acumulada do tempo gasto nas principais classes para o servidor WC98 comparado com a distribuição Weibull, ilustrando a boa aderência observada. A Tabela 4(b) mostra os parâmetros da distribuição de Weibull utilizada para caracterizar o tempo de transmissão dos arquivos de acordo com sua classe.



(a)



(b)

**Figura 6. (a) Série temporal representando o tráfego de saída do simulado com dados do servidor WC98 na escala de 1 segundo e (b) Estimador do parâmetro de Hurst utilizando a transformada Wavelet para o tráfego de saída simulado com dados do servidor WC98**

## 5. Validação e simulação

Para validar o modelo foi realizada uma simulação utilizando o software simulador NS-2 [Breslau et al. 2000]. A topologia de simulação consiste de 100 clientes acessando simultaneamente um servidor Web. O enlace de transmissão do servidor é de 1Mbps, enquanto os clientes estão conectados à rede com enlaces de 10Mbps. Os parâmetros utilizados na simulação são os mesmos levantados no servidor WC98, descritos nas Tabelas 4 e 3. Foi desenvolvido um programa para gerar a seqüência de requisições à páginas de acordo com o modelo proposto e este tráfego sintético foi submetido ao simulador NS-2 como se fosse um arquivo *trace* real de um servidor Web. O intervalo de tempo *active on* foi gerado de acordo com a distribuição de Weibull, como descrito por Barford e Crovella em [Barford and Crovella 1998]. O processo de chegada de sessões de usuário segue o processo de Poisson, com uma chegada média de 450 requisições por hora. Esta taxa de chegada foi configurada de modo a gerar uma carga considerável ao servidor simulado sem no entanto produzir uma condição de congestionamento permanente.

A Figura 6(a) mostra o tráfego resultante observado na saída do servidor na escala de 1 segundo. Foi realizado o teste para estimar o parâmetro de Hurst do tráfego agregado utilizando-se o método da transformada Wavelet [Willinger and Park 2000] utilizando-se as ferramentas disponibilizadas por Roughan et al. em [Roughan et al. 1998]. A Figura 6(b) mostra a densidade espectral de potência em cada oitava calculada através do método da transformada Wavelet. O parâmetro de Hurst calculado foi de  $H \approx 0.9$  (para um intervalo de confiança de 95% o valor de H está no intervalo [0.843, 0.957]), o que demonstra a capacidade do modelo na geração de tráfego com características auto-similares, como reportado por [Crovella and Bestavros 1995] na observação do tráfego de sistemas servidores Web.

## 6. Conclusões e trabalhos futuros

O modelo SURGE é um dos melhores modelos disponíveis para geração de tráfego Web, mas exige que o tamanho dos arquivos seja caracterizado de acordo com uma única distribuição de probabilidade. Normalmente, a distribuição utilizada é uma distribuição de cauda pesada, o que pode levar a problemas na análise estatística [Gong et al. 2005]. No entanto, [Barford and Crovella 1998] sugere o uso da distribuição Lognormal para a distribuição do corpo do tamanho dos arquivos transferidos e a distribuição de Pareto para descrever a cauda, tornando difícil a parametrização de um sistema real. Além disso, isto resulta em um problema para implementação do modelo em simuladores de redes, como no NS-2. Em geral os simuladores não são capazes de utilizar mais de uma distribuição de probabilidade para descrever o tamanho de arquivos transmitidos e o usuário é forçado a utilizar somente uma distribuição, o que pode levar a uma geração de tráfego incorreta.

O modelo proposto resolve estes problemas através da separação dos arquivos transmitidos em classes. O modelo proposto neste artigo captura as características do tráfego realizando a separação dos arquivos transmitidos em diversas classes utilizando apenas a extensão do arquivo. No entanto, é possível realizar a montagem de classes baseadas em critérios mais refinados. Foi mostrado que, para cada classe, a distribuição Lognormal pode ser utilizada para caracterizar o tamanho do arquivo e a distribuição de Weibull pode ser utilizada para caracterizar o tempo de permanência em cada classe.

Como contribuições desta nova abordagem de modelagem citamos: a ampliação

de possibilidades em termos de novos modelos analíticos, por exemplo, através de cadeias de Markov, uma geração de tráfego sintético mais precisa do que o SURGE através do refinamento em classes de arquivos e a novas possibilidades de pesquisa de técnicas de melhoria de desempenho de sistemas Web, por exemplo, o desenvolvimento de algoritmos de gerência de áreas de cache.

A caracterização do tráfego de um servidor Web através do novo modelo requer as seguintes informações:

1. Classes de arquivos transmitidos pelo servidor;
2. Média e desvio padrão do tamanho de cada classe de arquivo.
3. Probabilidades de transição de estados para transmissão de classes de arquivos em uma sessão;
4. Intervalo entre chegada de sessões.

Os dados necessários podem ser extraídos dos arquivos de *log* do servidor. Neste trabalho foram desenvolvidos programas para a extração e classificação dos dados dos arquivos de *log* de servidores Web. Também foram desenvolvidos geradores de tráfego e realizada a simulação com o software NS-2.

Um possível trabalho futuro é a utilização do modelo para realizar análise de desempenho utilizando cadeias de Markov. Como o tempo de permanência em cada classe foi identificado como possuindo distribuição de Weibull, pode ser realizada uma análise aproximada (a distribuição de Weibull é caso geral da distribuição Exponencial) utilizando cadeias de Markov. Esta análise pode levar ao desenvolvimento de novos métodos de dimensionamento e técnicas para a melhoria de desempenho de servidores.

## Referências

- Arlitt, M. and Jin, T. (2000). A workload characterization study of the 1998 World Cup Web site. *IEEE Network*, 14:30–37.
- Barford, P. and Crovella, M. (1998). Generating representative web workloads for network and server performance evaluation. In *Joint International Conference on Measurement and Modeling of Computer Systems - Performance Evaluation Review (SIGMETRICS '98/PERFORMANCE '98)*.
- Breslau, L., Estrin, D., Fall, K., Floyd, S., Heidemann, J., Helmy, A., Huang, P., McCanne, S., Varadhan, K., Xu, Y., and Yu, H. (2000). Advances in network simulation. *IEEE Computer*, 33(5):59–67.
- Cao, J., Andersson, M., Nyberg, C., and Kihl, M. (2003). Web server performance modeling using an M/G/1/K\*PS queue. In *International Conference on Telecommunication (ICT 2003)*.
- Cao, J., Cleveland, W. S., Gao, Y., Jeffay, K., Smith, F. D., and Weigle, M. (2004). Stochastic Models for Generating Synthetic HTTP Source Traffic. In *IEEE Infocom*.
- Crovella, M. and Bestavros, A. (1995). Self-similarity in World Wide Web traffic: Evidence and possible causes. *IEEE/ACM Transactions on Networking*, 5(6):835–846.
- Fielding, R., Gettys, J., Mogul, J., Frystyk, H., Masinter, L., Leach, P., and Berners-Lee, T. (1999). Hypertext Transfer Protocol – HTTP/1.1. RFC 2616 (Draft Standard). Updated by RFC 2817.

- Gong, W.-B., Liu, Y., Misra, V., and Towsley, D. F. (2005). Self-similarity and long range dependence on the internet: a second look at the evidence, origins and implications. *Computer Networks*, 48(3):377–399.
- Hernandez-Campos, F., Jeffay, K., and Smith, F. (2003). Tracing the evolution of the web traffic: 1995-2003. In *IEEE/ACM MASCOTS 2003 – The 11th International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems*.
- Leland, W., Qaqqu, M., Willinger, W., and Wilson, D. (1994). On the self-similar nature of Ethernet traffic (extended version). *IEEE/ACM Transactions on Networking*, 2(1):1–15.
- Mah, B. A. (1997). An empirical model of http network traffic. In *INFOCOM '97: Proceedings of the INFOCOM '97. Sixteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Driving the Information Revolution*, page 592, Washington, DC, USA. IEEE Computer Society.
- Muscariello, L., Mellia, M., Meo, M., and Marsan, M. (2004). An MMPP-based hierarchical model of internet traffic. In *IEEE international conference on communications ICC2004*.
- Nossenson, R. and Attiya, H. (2004). Evaluating self-similar processes for modeling Web servers. In *Symposium on Performance Evaluation of Computer Telecommunication Systems (SPECTS 2004)*.
- Nuzman, C., Saniee, I., and Weiss, A. (2002). A compound model for TCP connection arrivals for LAN and WAN applications. *Elsevier Journal of Computer Networks*, 3(40):319 – 337.
- Pedroso, C. M., Kotelok, M., and Fonseca, K. (2005). Um modelo para avaliação de desempenho de servidores web utilizando classificação de conteúdo [short paper]. In *4th International Information and Telecommunication Technologies Symposium (I2TS2005)*.
- Roberts, J. (2000). *Self-similar network traffic and performance evaluation*, chapter Engineering for quality of service, pages 401–420. John Wiley & Sons, Inc.
- Roughan, M., Veitch, D., and Abry, P. (1998). On-line estimation of the parameters of long-range dependence. In *Proceedings Globecom '98*, volume 6, pages 3716–3721, Sydney.
- Sommers, J. and Barford, P. (2004). Self-configuring network traffic generation. In *IMC '04: Proceedings of the 4th ACM SIGCOMM conference on Internet measurement*, pages 68–81, New York, NY, USA. ACM Press.
- Wessels, D. and Claffy, K. (1996). Evolution of the NLANR cache hierarchy: Global configuration challenges. Technical report, NLANR, October 1996. <http://www.nlanr.net/Papers/Cache96/>.
- Willinger, W. and Park, K. (2000). *Self-similar network traffic and performance evaluation*. John Wiley & Sons, New York, 1st edition.