Escalonamento Realimentado para Diferenciação de Serviços e Garantia de Desempenho em ambientes SOA com requisitos Soft-RT*

Priscila T. M. Saito[1], Pedro N. Nobile[1], Francisco J. Monaco[1]
Universidade de São Paulo
Departamento de Sistemas de Computação
Instituto de Ciências Matemática e de Computação
São Carlos, SP, Brasil
{psaito[1], nobile[1], monaco[1]}@icmc.usp.br

Resumo

Trabalhos abordando provisão de QoS em nível de aplicação têm recebido crescente atenção. Diversas técnicas de escalonamento têm sido propostas objetivando garantias relativas ou absolutas de responsividade. No primeiro caso, investigam-se algoritmos para diferenciação de serviços baseados em atendimento preferencial à classes de serviço com distintas prioridades; no segundo, pretende-se oferecer garantias de desempenho especificadas para cada classe independentemente uma das outras. A integração de QoS relativa e absoluta não tem sido explorada da mesma forma. Este artigo apresenta uma estratégia de escalonamento realimentado capaz de atender a requisitos de QoS formulados em termos de limites superiores para o tempo médio de resposta das requisições, além de especificar que determinadas classes de usuários terão prioridade sobre outras.

1. Introdução

À medida que o uso de arquiteturas orientadas a serviços (SOA) vêm se disseminando na realização de serviços presentes no cotidiano da sociedade, aplicações computacionais com requisitos temporais de responsividade tornam-se cada vez mais comuns. Sob essa perspectiva, exemplos como ensino a distância, telemedicina, comércio eletrônico, dentre outros, demandam abordagens de análise e síntese pertinentes ao domínio dos sistemas de tempo-real - RT (*Real-Time*), na medida que devem atender restrições de tempo de resposta ditados pela dinâmica dos processos do mundo real com o qual interagem.

*Os autores agradecem o apoio financeiro da CAPES, CNPq e FAPESP.

Quando a violação de restrições temporais de responsividade não implica diretamente em falha, mas em degradação do serviço, o cumprimento de requisitos de tempo-real pode ser associado ao conceito de qualidade de serviço (QoS), referindo-se à capacidade dos elementos de um sistema em prover garantias acerca de determinados parâmetros associados à percepção da qualidade de um dado serviço oferecido. Exige-se que tais parâmetros permaneçam dentro de limites bem definidos [1].

A especificação da qualidade de serviço pode ser realizada segundo duas abordagens distintas: relativa ou absoluta. Quando em termos relativos, a preocupação com a QoS oferecida se dá comparando-se o tratamento oferecido para as diversas classes de serviço. O que se pretende garantir nesses casos é que uma classe de maior prioridade tenha um tratamento melhor que o de qualquer classe inferior. A QoS absoluta, por sua vez, estabelece requisitos de desempenho a serem atendidos, como garantir uma taxa mínima de serviço ou um atraso máximo de atendimento para as requisições.

Substancial contribuição para as técnicas de provisão de QoS é oriunda da área de redes de comunicação de dados, onde o desenvolvimento no nível de rede, tendo como referência o modelo OSI *(Open System Interconnection)*, se destaca como foco de pesquisa [2, 19, 12]. Algoritmos de roteamento para tráfego em tempo-real são especialmente relevantes nesse campo.

Por outro lado, iniciativas para a provisão de QoS em nível de aplicação começam a ganhar interesse [17, 18, 7]. Diversas técnicas têm sido propostas a fim de viabilizar e evoluir as formas de disponibilização de QoS em nível de aplicação [13, 1, 16]. Um tópico importante refere-se às políticas de escalonamento aplicadas ao atendimento de requisições pendentes no provedor de serviços.

Em se tratando de escalonamento, a distinção essencial entre QoS relativa e absoluta é a de que a enquanto a pri-

meira tem a prioridade de atendimento como base do contrato de serviço a ser garantida pelo provedor, a segunda especifica métricas e valores de desempenho a serem respeitadas, sendo a prioridade, neste caso, decidida dinamicamente em função do estado do sistema e das condições efetivas dos contratos estabelecidos.

Políticas baseadas tanto na diferenciação de serviços em termos relativos [4, 8], quanto na especificação absoluta da QoS no que se refere ao cumprimento de requisitos temporais em termos absolutos [11, 14], encontram-se na literatura, porém nenhuma delas trata de maneira eficaz a relação de QoS relativa e absoluta em uma mesma técnica.

Este artigo contribui para esse campo introduzindo uma técnica de *Feedback Scheduling* aplicável a métodos existentes capaz de oferecer garantias de QoS absoluta e relativa para sistemas Soft-RT com requisitos temporais dados em limites superiores para o tempo médio de resposta.

2 Garantias de Tempo Médio de Resposta

O nível de desempenho, oferecido aos sistemas, especifica a qualidade de serviço oferecida ao mesmo. Para quantificar tal desempenho, uma métrica geralmente utilizada é a média do tempo de resposta do sistema que representa o tempo médio de residência das requisições de um usuário no sistema, ou seja, o intervalo entre a submissão e o completo recebimento do resultado da requisição.

Tal métrica é considerada, portanto, uma boa alternativa, visto que em seu cálculo estão inclusos os tempos em fila das requisições e, a manipulação desses tempos, de acordo com o parâmetro contratual de cada classe, influencia diretamente na melhora do desempenho de aplicações *Web*. Sendo assim, as requisições que estão mais próximas ou foram descumpridas receberão maior prioridade do escalonador, ao contrário daquelas que toleram maiores tempos em fila.

A política EBS (*Exigency-Based Scheduling*) [1], a qual realiza o escalonamento de requisições *Web* em sistemas *Soft-RT* não-determinístico, provê garantias de QoS absoluta em nível de aplicação. Para tanto, utiliza um limite superior para a média dos tempos de resposta, a ser garantido às requisições de um determinado usuário, como parâmetro de qualidade de serviço. Esse parâmetro, denominado tempo médio de resposta contratado pela i-ésima classe (T_{c_i}), é especificado previamente por um acordo entre o provedor de serviços e o usuário, e utilizado pelo escalonador como base para atribuição de prioridades. Nesse caso, o valor do parâmetro, deve ser observado pelo servidor durante uma sessão.

A Figura 1 ilustra o sistema modelado como uma rede de fila. Os eventos de interesse que ocorrem em um sistema real do gênero são eventos para tratamento da chegada de requisições, solicitação de serviço e liberação de recurso. O seguinte exemplo ilustra como opera o algoritmo EBS com um servidor monoprocessado e uma fila única de espera para processamento.

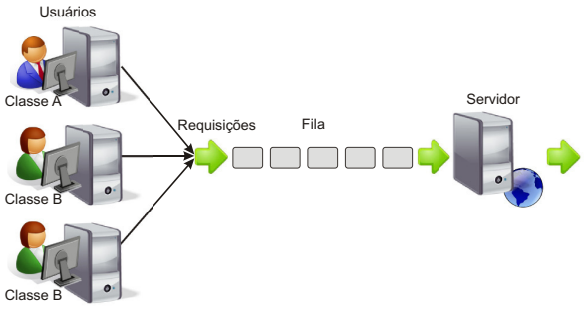

Figura 1. Representação do Modelo de Servidor *Web* Seqüencial com QoS.

As novas requisições que chegam ao sistema e não encontram servidor disponível são transferidos para a fila de espera, visto que a política não é preemptiva. Ao término da execução de uma requisição j do usuário u, o valor de T_{e_u}, média instantânea de tempo de resposta efetivamente oferecida ao usuário u, é recalculada. A Equação 1 mostra esse cálculo, o qual corresponde à média entre o antigo tempo de resposta efetivo de u (T'_{e_u}) e o tempo de residência da requisição j recém atendida. Os valores de $time()$, $timeStamp_j$ e R_u representam, respectivamente, o tempo atual, o tempo de chegada da requisição e o número de requisições anteriormente submetidas por u.

$$T_{e_u} = \frac{(T'_{e_u} \cdot R_u) + (time() - timeStamp_j)}{R_u + 1} \quad (1)$$

Após o valor de T_{e_u} ser recalculado, o escalonador procura na fila a requisição de maior prioridade, a qual acessará o servidor e então o ciclo se repete.

Um dos fatores considerados para definição da prioridade de uma requisição é dada pelo tempo de espera máximo (*deadline* - D_j), o qual representa o quanto uma requisição ainda pode esperar na fila antes de começar a descumprir seu contrato, ou seja, antes que o valor de T_{e_u} ultrapasse o valor de T_{c_i}. Tal fator, pode ser calculado isolando-se a variável D_j, na Inequação 2, em que T_{w_j} expressa o tempo de espera em fila da requisição j até o momento.

$$\frac{(T_{e_u} \cdot R_u) + T_{w_j} + D_j}{R_u + 1} \leq T_{c_i} \quad (2)$$

A medida que o valor de D_j diminui, maior é a sua prioridade, pois maior é sua urgência. Em certas circunstâncias, requisições urgentes podem assumir os mesmos valores

de *deadline*. Embora apresentem a mesma urgência, as requisições podem impor pesos distintos ao sistema, uma vez que, o tempo de processamento é um fator impactante sobre a exigência imposta ao sistema. A minimização desse impacto torna-se, então, de extrema importância, visto que um sistema sob menor carga terá melhores condições para lidar com os requisitos de serviço de suas requisições.

Portanto, faz sentido também, dentre as mais urgentes, reordenar as requisições pelos seus valores esperados de processamento, visto que, requisições menores tendem a sair mais rapidamente do sistema, diminuindo o tempo de espera das demais, e consequentemente, afetando a quantidade de requisições aguardando por atendimento. Isso possibilita uma melhor utilização de recursos do sistema, bem como menores médias de tempo de resposta, contribuindo, assim, para obtenção de melhores níveis de qualidade de serviço.

Sendo assim, a atribuição de prioridades é dada baseando-se em uma estratégia híbrida que considera as políticas de escalonamento EDF (*Earliest Deadline First*) – a qual prioriza as requisições cujos deadlines encontram-se próximos do tempo atual – e SJF (*Shortest Job First*) – a qual prioriza os *jobs* mais curtos para minimizar a média dos tempos de resposta. A Equação 4 mostra essa atribuição de prioridades, em que a prioridade de uma dada requisição j em fila, do usuário u, é dada por P_j e as requisições que apresentarem maior urgência (menores valores de D_j) e menor valor esperado do tempo de processamento (T_{P_j}) serão classificadas como mais prioritárias.

$$\begin{aligned} P_j &= D_j \cdot T_{p_j} \qquad (3) \\ P_j &= \left(\Big(T_{c_u} \cdot (R_u + 1) \Big) - (T_u \cdot R_u) - T_{w_j} \right) \cdot T_{p_j} \end{aligned}$$

Em alguns casos o *deadline* pode assumir valores negativos, representando que o tempo que a requisição pode aguardar na fila é menor que zero, ou seja, o contrato foi violado. Se existirem duas requisições (R_1 e R_2) mais urgentes, com seus respectivos *deadlines* (D_i) negativos e tempos de processamento (T_{p_i}), por exemplo, uma requisição R_1 com $D_1 = -1$ e $T_{p_1} = 7.5ut$ e uma requisição R_2 com $D_2 = -2$ e $T_{p2} = 2.5ut$. Se o escalonamento proposto (multiplicar o *deadline* pelo tempo de processamento) for aplicado nesse exemplo, embora R_2 seja mais urgente, a requisição R_1 receberia maior prioridade e seria inicialmente escalonada, devido $P_1 < P_2$ ($P_1 = -7.5ut$ e $P_2 = -5.0ut$). Nesses casos, é realizada uma correção, de forma a manter o objetivo proposto.

A Equação 4 representa tal correção, em que a prioridade das requisições mais urgentes, que ainda não tiveram seus *deadlines* descumpridos, é diretamente proporcional ao seu *deadline* e ao seu custo esperado de processamento. Já para aquelas com descumprimento de *deadline*, a prioridade é inversamente proporcional ao seu custo esperado de processamento, visto que, quanto menor for esse custo, menor será o valor de P_j resultante e portanto maior será sua prioridade de escalonamento. Sendo assim, garante-se que as requisições mais urgentes, independente de terem descumprido ou não seus *deadlines*, e com menores custos esperados de processamento sejam escalonadas primeiro.

$$P_j = \begin{cases} D_j \cdot T_{p_j} & \text{se } D_j \geq 0 \\ D_j \cdot \frac{1}{T_{p_j}} & \text{se } D_j < 0 \end{cases} \qquad (4)$$

3 Definição do Problema

A política EBS [1], abordada anteriormente, oferece como contribuição uma técnica de escalonamento para sistemas computacionais interativos cujo desempenho no atendimento às especificações estocásticas de responsividade temporal se mostraram superiores à alternativas convencionais, tais como a FIFO, a EDF e a SJF.

Dentre outras características, a EBS produz um balanceamento das cargas computacionais entre classes de serviços (A e B) de modo a poupar recursos, em termos de demanda de potência, destinados à classe B, a qual apresenta um maior tempo médio de resposta estabelecido em contrato, ou seja, um contrato mais "relaxado", para garantir recursos às classes que deles mais necessitam (classe A - cujo contrato é mais estrito). Essa propriedade permite à EBS cumprir os contratos de QoS absoluta baseados em tempo médio de resposta para cargas mais altas que outras técnicas convencionalmente empregadas em escalonamento de serviços interativos, como na *Web*, por exemplo.

Todavia, em determinados cenários de carga, para efetuar o balanceamento de potência computacional, a aplicação da EBS tem como efeito o fato de que uma classe, mesmo apresentando um maior valor de contrato, recebe um atendimento melhor (tempo médio de resposta menor) em relação às demais classes, as quais apresentam valores de contrato menores, e portanto são mais estritas.

Em contratos de QoS absoluta, onde não existe qualquer relação de prioridade entre classes, tal fato não constitui dificuldade. Nesses casos, a relação dos tempos de resposta entre as classes de serviços pode ser qualquer, podendo inclusive inverter-se ao longo do tempo, desde que os limites superiores (contrato A e contrato B) estabelecidos para cada uma delas sejam respeitados (Figura 2).

Enquanto tal fato não seja um problema para o caso de QoS absoluta, é plausível considerar qual a percepção dos usuários familiarizados ao modelo de atendimento preferencial da QoS relativa. Se uma classe com contrato mais estrito (menor tempo médio de resposta contratado) demanda mais recursos do sistema, e por isso deva arcar com custos maiores, prover-lhe um tempo de resposta superior (Figura

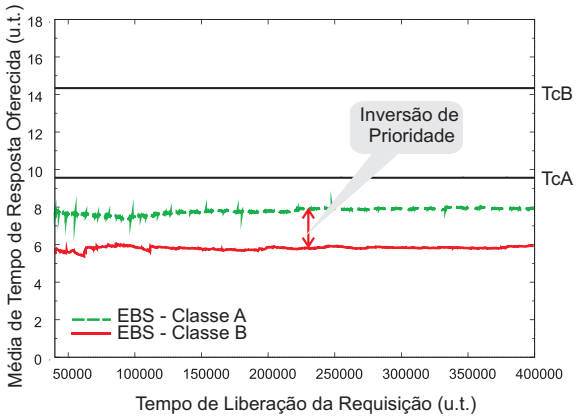

Figura 2. Representação do cenário apresentado pela política EBS: cumprimento dos contratos A e B em termos de QoS absoluta e ocorrência de inversão de prioridade em termos de QoS relativa.

2) pode se lhe apresentar intuitivamente como uma situação de conflito. Quando o contrato de QoS é estabelecido de modo relativo, tal que é prometido a uma classe um serviço "melhor" que a outra, a mesma situação denomina-se *inversão de prioridade* e constitui uma falha no atendimento às especificações. Para o efeito da elaboração de modelos de negócios de provedores de serviços, seria conveniente investigar a possibilidade de evitar tal circunstância, atendendo, assim, às expectativas dos usuários.

4 Abordagem Proposta

A abordagem introduzida neste artigo utiliza-se da técnica de *Feedback Scheduling* e de contratos virtuais para ajustar adaptativamente os parâmetros de operação do Algoritmo EBS, afim de evitar inversões de prioridades em sistemas cuja operação seja associada a um contrato de QoS com garantias absolutas e relativas de limite superior de tempo médio de resposta.

4.1 Escalonamento Realimentado

Garantir que o tempo médio entre eventos de um dado processo seja superiormente limitado dentro de uma janela de ocorrências conveniente pode ser útil em diversas formas. Pode ser de ajuda para medir o *throughput* de sistemas estocásticos e para estimar demandas de recursos. Para sistemas computacionais interativos, isso representa não apenas um parâmetro de QoS praticável face ao não-determinismo do ambiente, mas também uma métrica temporal de responsividade com significado intuitivo para a percepção do usuário, tendo assim um impacto relevante na qualidade de serviço [7], [3], [6], [9].

Mesmo em tais sistemas o projeto de algoritmos de escalonamento de tarefas que competem pelos recursos disponíveis não é simples. As dificuldades para abordagens analíticas direcionam interesses para métodos heurísticos. Ainda assim, em ambientes com pouca previsibilidade, algoritmos de escalonamento baseados no pré-ajuste de parâmetros fixos em fase de projeto tende a ser ineficiente em termos de demandas calculadas com base no pior caso, ou ineficazes frente à variações na carga ou na capacidade de processamento.

Um conceito de crescente relevância na área de sistemas de tempo-real é o de *Feedback Scheduling* (escalonamento retroalimentado), que corresponde à aplicação dos fundamentos da Teoria de Controle em problemas de escalonamento. A abordagem baseia-se no princípio da retroalimentação negativa, em que a saída do sistema é comparada a um valor de referência desejado, a diferença entre eles é tomada como entrada para o sistema, e manipulada de modo a causar neste uma reação contrária ao desvio entre referência e saída. Com isso é possível controlar o mecanismo de escalonamento em tempo de operação, de maneira a compensar variações na carga ou nos parâmetros do sistema. *Feedback Scheduling* constitui um dos paradigmas mais recentes e mais promissores dentro da área de escalonamento de tempo-real em ambientes não determinísticos [10], [15].

4.2 Contratos Virtuais Adaptativos para a Classe $B - vc_B$

A Figura 3 ilustra um cenário de ocorrência de *inversão de prioridade* em termos de QoS relativa. Pode-se observar que em termos de QoS absoluta não ocorre problemas, pois os contratos das classes A e B estão sendo cumpridos, ou seja, os tempos médios de resposta oferecidos tanto aos usuários da classe A como aos usuários da classe B encontram-se abaixo dos estabelecidos em seus respectivos contratos A e B. Porém, em termos de QoS relativa, percebe-se que usuários da classe de serviço B recebem um atendimento melhor em relação aos da classe A.

Portanto, para resolver o problema de inversão, uma possível solução, apresentada na Figura 3, seria oferecer tempos médios de resposta efetivos T_{e_B} suficientes apenas para atender aos parâmetros de QoS dos usuários da classe B. Para tanto, considera-se a existência de um contrato virtual T_{vc_B}, de forma a relaxar suas restrições, desde que os tempos de resposta efetivos sejam mantidos abaixo de seu limite superior contratado (T_{c_B}). Isso é possível, visto que, os tempos médios de resposta efetivamente oferecidos aos usuários da classe de serviço B são bem menores que os es-

tabelecidos em cada um de seus contratos, conforme pode ser observado pela Figura 3.

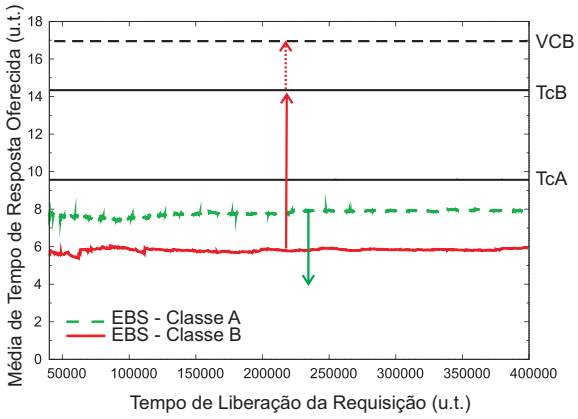

Figura 3. Representação do cenário de ocorrência de inversão de prioridade em termos de QoS relativa e solução considerando-se um contrato virtual T_{vc_B}.

Essa solução possibilita ao escalonador priorizar uma requisição cuja QoS esteja próxima ao limite especificado, em detrimento de outra, cujo serviço, fornecido ao longo do tempo, tenha sido eventualmente realizado com qualidade superior a do nível contratado, revertendo, assim, o cenário de inversão. O Algoritmo 1 descreve o cálculo do contrato virtual T_{vc_B}.

Algorithm 1 - Algoritmo para calcular o contrato virtual T_{vc_B}

if $T_{e_B} <= \frac{T_{c_A}+T_{e_A}}{2}$ **then**
  $k \Leftarrow (T_{c_B} - T_{e_B}) \cdot T_{e_A}$
else
  $k \Leftarrow 1$
end if
$T_{vc_B} \Leftarrow k \cdot T_{c_B}$

O algoritmo 1 atua nos cenários em que o tempo médio de resposta da classe B é menor e/ou bastante próximo do tempo médio de resposta da classe A.

Para garantir um tratamento melhor aos usuários da classe A, um limiar que define o limite inferior para os tempos médios de resposta da classe B é estabelecido. Este limiar é definido como sendo o valor médio do contrato da classe A e o valor efetivo oferecido aos usuários dessa classe.

Enquanto a classe B apresenta melhor tratamento, há um aumento do contrato virtual proporcional à diferença entre o contrato de B e seu tempo efetivo, e proporcional ao tempo médio de resposta efetivo da classe A.

O contrato virtual permite que o tempo médio de resposta de B aumente em direção ao seu contrato, mas não permite que se aproxime positivamente da média dos tempos de resposta da classe A. Uma vez que a classe A apresenta valores menores que os de B e a QoS relativa é obtida, o contrato virtual deixa de atuar ($k = 1$) e passa a ser o próprio contrato real de B (T_{c_B}). Os passos do método proposto são ilustrados na Figura 4.

5 Experimentos

Essa seção apresenta os resultados experimentais da utilização do método proposto para escalonamento de sistemas *Soft-RT*, que considere, principalmente, a percepção dos usuários familiarizados ao modelo de atendimento preferencial da QoS relativa.

5.1 Descrição dos Cenários

A política proposta foi avaliada em diferentes cenários, os quais foram definidos, por meio da variação de alguns parâmetros, os mesmos considerados no desenvolvimento da política EBS para análise se a política proposta se apresenta ou não adequada nos cenários de ocorrência das inversões.

Um dos parâmetros que define os cenários é a porcentagem de variação de contrato (V), ou seja, a porcentagem de discrepância dos serviços, oferecidos por cada algoritmo em relação ao tempo de residência em um sistema de escalonamento convencional FIFO (T_{FIFO}). Utilizou-se, como base, médias de tempo de resposta que seriam oferecidas por um servidor convencional sem suporte à QoS, com o intuito de se definir contratos de serviços viáveis e não contratos com tempos impossíveis de serem garantidos com qualquer tipo de algoritmo de escalonamento.

Foram atribuídos à V os valores: 5%, 10% e 20%. Sendo que, quanto maior for a variação dos contratos em relação ao escalonamento FIFO, maior e menor será a dificuldade de atendimento dos usuários das classes A e B, respectivamente. As Equações 5 e 6 ilustram os cálculos dos contratos de A e de B, respectivamente.

$$T_{c_A} = T_{FIFO} - (T_{FIFO} \cdot V) \quad (5)$$

$$T_{c_B} = T_{FIFO} + (T_{FIFO} \cdot V) \quad (6)$$

Outro parâmetro considerado é a proporção de requisições da classe A no sistema. Esse parâmetro especifica um sistema sobrecarregado de requisições da classe que apresenta valor de contrato menor ($90\%A$ e $10\%B$), situações estas em que, dependendo da variação contratual,

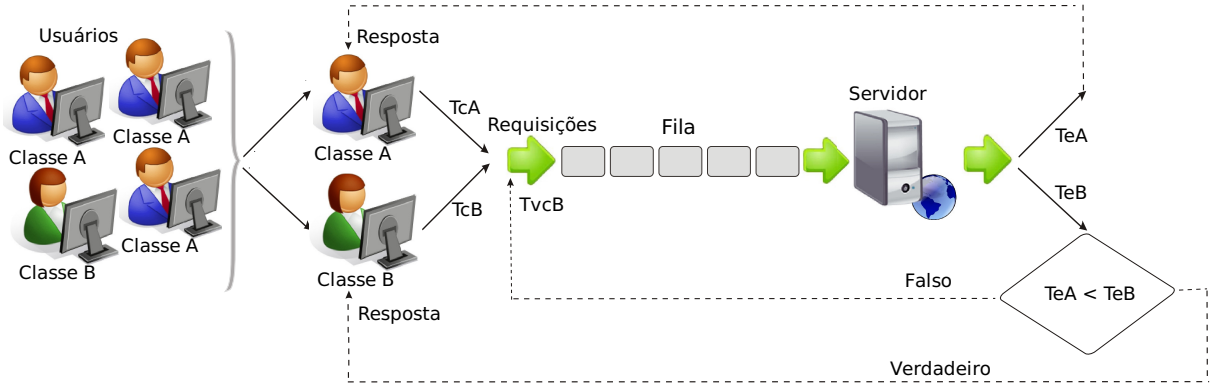

Figura 4. Passos do método proposto.

ocorrem as inversões e portanto, deve ser utilizado o contrato virtual (T_{vc_B}) proposto.

Para avaliação da eficiência e do desempenho da política proposta, foi utilizada como abordagem, a modelagem e simulação orientada a eventos. Sendo assim, considerando os conceitos de redes de filas, utilizou-se o mesmo modelo de servidor *Web* com qualidade de serviço apresentado (Figura 1).

Para a validação do modelo e da política de escalonamento utilizou-se a ferramenta de simulação SimpackJ [5]. A fim de incluir a nova política de escalonamento, algumas modificações e extensões foram realizadas nessa ferramenta. Definiu-se novos campos para descrever de modo mais completo as entidades que representam as requisições *Web*, incluiu-se suporte a QoS, e alterou-se procedimentos que implementam eventos, como os de requisição e liberação de recurso, além do acréscimo de alguns mecanismos de instrumentação.

Para execução dos experimentos de simulação, foram especificadas duas classes de serviço, uma classe A com um contrato mais estrito (valor de contrato menor) e uma outra classe B com um contrato mais "relaxado" (valor de contrato maior). Os cenários foram compostos por 20 usuários onde os 10 primeiros apresentavam menores valores de tempo médio de resposta contratado e os outros 10 apresentavam maiores valores. Considerando cada uma das classes, os 10 usuários apresentam um mesmo modelo de geração de requisições, caracterizado por um intervalo de chegada e tempo de execução, descritos por uma distribuição exponencial com médias 4 u.t. e 3 u.t., respectivamente, obtendo-se, portanto, uma taxa de utilização do sistema de 75%.

Fixou-se como 100.000, o número de requisições submetidas em cada cenário simulado, de forma a apresentar uma ampla amostragem dos dados. Além disso, com o intuito de obter confiabilidade estatística, a simulação foi executada utilizando-se as diferentes sementes, i.e., replicações com fluxos de números aleatórios diferentes, disponibilizadas pelo SimpackJ. Sendo assim, os resultados obtidos foram analisados, segundo a média e o intervalo de confiança.

5.2 Resultados

Para avaliar a política proposta, os resultados obtidos com a simulação dos cenários apresentados na seção anterior são analisados.

As Figuras 5 a 7 ilustram a seleção de alguns cenários principais de proporção de requisições ($90\%A - 10\%B$) e variação de contrato (5%, 10% e 20%, respectivamente), dentre todos os estudados, de forma a analisar a influência destes parâmetros sobre a eficácia da política desenvolvida em respeitar os tempos de resposta estabelecidos nos contratos de cada classe, bem como a diferenciação entre elas.

Nesses gráficos são ilustradas as médias do tempo de resposta do sistema oferecidas ao longo do tempo para as classes de serviço com maior e menor dificuldade de atendimento, classes A e B, respectivamente. O eixo das abscissas informa o término de atendimento de uma requisição e o momento em que a média do tempo de resposta efetivamente oferecida àquele usuário é atualizada. O eixo das ordenadas representam tais médias. Os valores dos contratos que definem as classes de serviço (A e B) são representados pelas retas horizontais T_{c_A} e T_{c_B}, respectivamente.

Em cada um dos cenários analisados é realizado um comparativo entre o comportamento da política EBS e o do método proposto. É importante ressaltar que a utilização da política EBS sem o contrato virtual provoca inversões de prioridade em todos os gráficos

No cenário, apresentado pela Figura 5, há uma proporção de 90% de requisições da classe A e 10% de requisições da classe B com uma variação contratual de 5%. Por meio dos resultados obtidos é possível observar que a média do

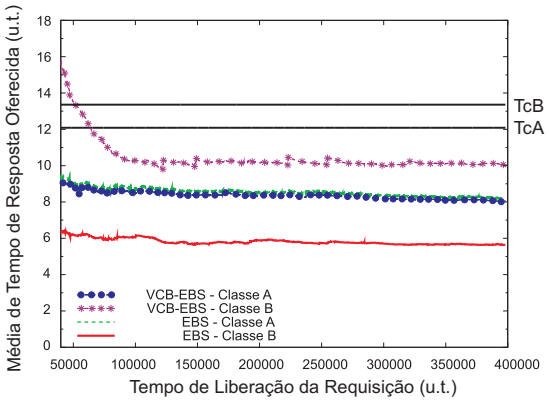

Figura 5. Representação do cenário 90%A-10%B com 5% de variação contratual, considerando-se o contrato virtual T_{vc_B}.

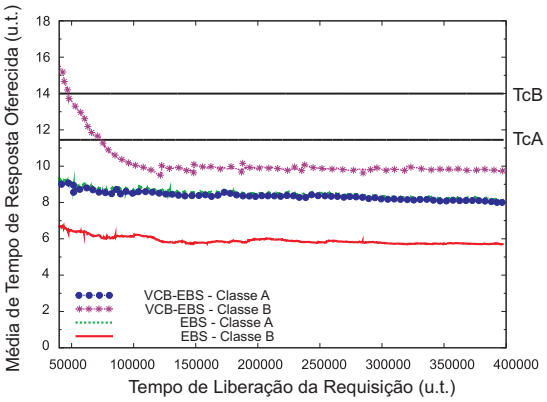

Figura 6. Representação do cenário 90%A-10%B com 10% de variação contratual, considerando-se um contrato virtual T_{vc_B}.

contrato virtual mostra-se bastante superior ao contrato da classe B. Entretanto, esse aumento é suficientemente necessário para elevar o tempo médio de resposta oferecido aos usuários da classe B, de forma que esse tempo seja superior ao tempo médio de resposta oferecido aos usuários da classe A, revertendo, então, a inversão de prioridade. Isso é possível visto que a média dos tempos de resposta efetivos dos usuários da classe B é bem inferior à sua média contratada.

Analisando o gráfico da Figura 6, pode-se observar que com 10% de variação contratual, o valor do contrato virtual vc_B é maior que o apresentado no gráfico da Figura 5. Portanto, o aumento nos tempos médios de resposta oferecidos aos usuários da classe B também é maior, visto que quanto maior o valor do contrato virtual vc_B, menor são as restrições dos usuários dessa classe.

No gráfico ilustrado pela Figura 7 também é possível observar que o aumento no tempo médio de resposta oferecido aos usuários da classe B é insuficiente para fazer com que o mesmo supere o contrato estipulado, mantendo, dessa forma, a QoS absoluta. Além disso, é possível observar que, como os contratos são cumpridos com determinada folga, o tempo médio de resposta de B não tem um crescimento acentuado e fica em torno da média entre o tempo médio de resposta e o contrato da classe A.

Devido às limitações de espaço, foram considerados tais cenários, porém, pôde-se observar que sob condições de cargas maiores, a utilização do método proposto apresentou bons resultados, revertendo os cenários em que ocorrem as inversões de prioridade.

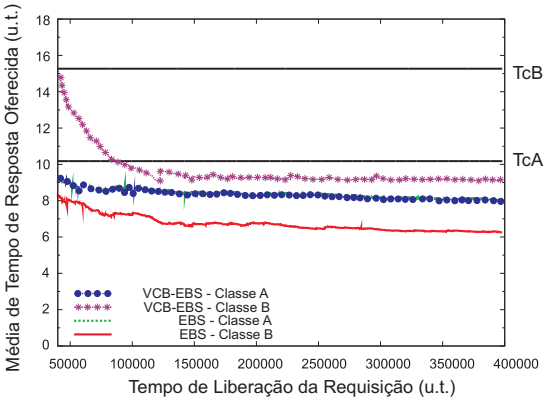

Figura 7. Representação do cenário 90%A-10%B com 20% de variação contratual, considerando-se um contrato virtual T_{vc_B}.

6 Conclusões

A principal contribuição deste trabalho é o desenvolvimento de um mecanismo de escalonamento adaptativo de requisições *Web*, com restrições de tempo real especificadas em termos de limites superiores para o tempo médio de resposta.

A abordagem introduzida é baseada em um contrato virtual adaptativo de forma a relaxar as restrições temporais dos usuários da classe B em situações de carga onde há uma porcentagem maior de requisições mais exigentes ($90\%A$ e $10\%B$), situações estas em que pôde-se observar as inversões de prioridade. Isso foi possível, visto que os tempos

médios de resposta efetivamente oferecidos aos usuários da classe de serviço B são bem menores que os estabelecidos em seus contratos.

Os experimentos realizados permitem concluir que a utilização do contrato virtual possibilita ao escalonador priorizar as requisições cuja QoS esteja próxima ao limite especificado, em detrimento de outras, cujo serviço, fornecido ao longo do tempo, tenha sido eventualmente realizado com qualidade superior a do nível contratado, revertendo, assim, o cenário de inversão.

Sendo assim, a especificação de QoS não indica apenas que, no decorrer do tempo, os tempos médios de resposta praticados no atendimento das requisições do referido usuário não serão maiores que o limite superior contratado, mas também permite estabelecer que determinados usuários serão atendidos com prioridade em relação aos demais, considerando, principalmente, a percepção dos usuários familiarizados ao modelo de atendimento preferencial da QoS relativa.

O modelo de sistema composto por duas classes pode ser estendido de forma a abranger n classes de serviços. Nesse caso, em cenários de ocorrência de inversões de prioridade, o contrato virtual adaptativo proposto pode ser definido tendo como referência um valor médio dos contratos.

Referências

[1] L. S. Casagrande, R. F. de Mello, R. Bertagna, J. A. Andrade Filho, and F. J. Monaco. Exigency-based real-time scheduling policy to provide absolute QoS for web services. *19th International Symposium on Computer Architecture and High Performance Computing - SBAC-PAD*, 0:255–262, 2007.

[2] X. Chen and J. Heidemann. Preferential treatment for short flows to reduce web latency. *Computer Networks: The International Journal of Computer and Telecommunications Networking*, 41(6):779–794, 2003.

[3] L. Eggert and J. Heidemann. Application-level differentiated services for web servers. *World Wide Web*, 2(3):133–142, 1999.

[4] J. C. Estrella, M. M. Teixeira, and M. J. Santana. Negotiation mechanisms on application level: a new approach to improve quality of service in web servers. *The 4th IEEE Workshop on Software Technologies for Future Embedded and Ubiquitous Systems, and 2nd International Workshop on Collaborative Computing, Integration, and Assurance. SEUS/WCCIA*, 0:255–260, April 2006.

[5] P. A. Fishwick. Simpackj: Simpack toolkit - version 1.0. disponível em: http://www.cise.ufl.edu/fishwick/simpackj, 2004.

[6] D. Henriksson, Y. Lu, and T. Abdelzaher. Improved prediction for web server delay control. In *Proceedings of the 16th Euromicro Conference on Real-Time Systems - ECRTS*, pages 61–68. IEEE Computer Society, 2004.

[7] K.-D. Kang, S. H. Son, and J. A. Stankovic. Differentiated real-time data services for e-commerce applications. *Electronic Commerce Research*, 3(1-2):113–142, 2003.

[8] V. Kanodia and E. W. Knightly. Ensuring latency targets in multiclass web servers. *IEEE Transactions on Parallel and Distributed Systems*, 14(1):84–93, 2003.

[9] C. Lu, T. F. Abdelzaher, J. A. Stankovic, and S. H. Son. A feedback control approach for guaranteeing relative delays in web servers. In *Proceedings of the Seventh Real-Time Technology and Applications Symposium - RTAS*, page 51. IEEE Computer Society, 2001.

[10] Y. Lu, T. Abdelzaher, C. Lu, L. Sha, and X. Liu. Feedback control with queueing-theoretic prediction for relative delay guarantees in web servers. In *Real-Time and Embedded Technology and Applications Symposium, 2003. Proceedings. The 9th IEEE*, pages 208–217, 27-30 May 2003.

[11] F. J. Monaco, M. Nery, and M. M. L. Peixoto. An orthogonal real-time scheduling architecture for responsiveness QoS requirements in SOA environments. In *Proceedings of the ACM symposium on Applied computing - ACM/SAC*, pages 1–6, New York, NY, USA, 2009. ACM. to be published.

[12] K. Nichols, S. Blake, F. Baker, and D. Black. RFC 2474: Definition of the differentiated services field (DS Field) in the IPv4 and IPv6 headers. *Internet RFC, Internet Engineering Task Force - IETF*, 1999.

[13] W. Pan, D. Mu, H. Wu, and L. Yao. Feedback control-based QoS guarantees in web application servers. In *Proceedings of the 10th IEEE International Conference on High Performance Computing and Communications - HPCC*, pages 328–334, Los Alamitos, CA, USA, 2008. IEEE Computer Society.

[14] M. L. M. Peixoto, R. Tott, M. Nery, and F. J. Monaco. Arquitetura de escalonamento ortogonal de tempo-real para garantias de QoS em servidores web. In *Workshop em Desempenho de Sistemas Computacionais e de Computação - WPerformance*, pages 18–37. Anais do XXVIII Congresso da SBC, 2008.

[15] L. Sha, T. Abdelzaher, K.-E. Arzén, A. Cervin, T. Baker, A. Burns, G. Buttazzo, M. Caccamo, J. Lehoczky, and A. K. Mok. Real time scheduling theory: A historical perspective. *Real-Time Syst.*, 28(2-3):101–155, 2004.

[16] M. M. Teixeira, M. J. Santana, and R. H. C. Santana. Using adaptive priority controls for service differentiation in QoS-enabled web servers. In *International Conference on Computational Science - ICCS*, volume 3036 of *Lecture Notes on Computer Science*, pages 537–540, Cracóvia, Polônia, 2004. Springer.

[17] M. M. Teixeira, M. J. Santana, and R. H. C. Santana. Servidor web com diferenciação de serviços: Fornecendo qos para os serviços da internet. In *XXIII Simpósio Brasileiro de Redes de Computadores (SBRC)*, pages 1–14, Fortaleza, CE, 2005.

[18] J. Wei and C. Xu. A self-tuning fuzzy control approach for end-to-end QoS guarantees in web servers. In *Proceedings of the 13th International Workshop Quality of Service - IWQoS*, volume 3552 of *Lecture Notes in Computer Science*, pages 123–135, Passau, Germany, 2005. Springer.

[19] X. Xiao and M. Ni, L. Internet QoS: A big picture. *IEEE Network*, 13(2):8–18, 1999.