

4

Disciplinas de serviço

Um importante componente na arquitetura dos sistemas que oferecem qualidade de serviço é a disciplina de serviço [19, 20, 21] a ser imposta às filas de armazenamento de pacotes. Estas disciplinas de serviço são regidas por algoritmos de escalonamento que por definição são conjuntos de instruções utilizadas pelas disciplinas de serviço para decidir qual pacote dos vários fluxos independentes de pacotes será encaminhado no enlace compartilhado de saída de um nó específico da rede.

O objetivo principal dos algoritmos de escalonamento é evitar que ocorram problemas relacionados com a alocação de recursos limitados da rede tais como banda ou memória de buffers de usuários, aplicações ou classes de serviços que competem entre si. Com estes algoritmos é possível gerenciar o modo como as classes de serviço acessam esses recursos limitados da rede através da decisão de qual pacote será trafegado na rede.

Independente da decisão tomada pelo algoritmo pode ocorrer congestionamento na rede quando a velocidade de chegada de pacotes for maior que a velocidade de processamento. Isso implica em aumento do atraso no recebimento dos pacotes, diminuição da taxa de transmissão, aumento do jitter, o que pode levar a perda dos pacotes. Todas essas conseqüências do congestionamento da rede são fatores indesejáveis sob o ponto de vista tanto dos usuários quanto da provedora de serviços de telecomunicações. É função também dos algoritmos de escalonamento minimizar os efeitos ocorridos devido ao congestionamento da rede e gerenciar as filas dos pacotes.

As disciplinas de serviço são implementadas por algoritmos que podem ser classificados como “Trabalhadores conservadores” (do inglês, “work-conserving”) ou “Trabalhadores não-conservadores” (do inglês, “non-work-conserving”) [22]. Os algoritmos “Trabalhadores conservadores” nunca se tornam ociosos quando há pacotes enfileirados para transmissão, porém os

“Trabalhadores não-conservadores” podem se tornar ociosos mesmo quando há pacotes em filas de baixas prioridades para transmissão. Os algoritmos “Trabalhadores não-conservadores” servem para controlar a característica de rajada dos fluxos de pacotes assim como controlar a variância do atraso (jitter). Os algoritmos utilizados nesse trabalho pertencem à classe dos algoritmos “Trabalhadores conservadores”.

Em geral as disciplinas de serviço devem ter as seguintes propriedades:

Limite de atraso: Aplicações interativas (como vídeo e áudio conferência) requerem um limite no atraso total percebido por um pacote na rede com base no atraso fim-a-fim. A disciplina decide a ordem na qual os pacotes são enviados no enlace de saída, e, portanto o atraso no enfileiramento dos pacotes em cada nó intermediário da rede.

Baixa complexidade de implementação: Com taxas de transmissão chegando a 40 Gb/s, torna-se crítico que todas as tarefas de processamento realizadas por roteadores, incluindo enfileiramento de pacotes na saída, possam ser efetuadas na ordem de nano segundos. A complexidade na escolha do próximo pacote a ser escalonado deve ser pequena, e em particular, é desejável que essa complexidade seja uma pequena constante independente do número de fluxos de pacotes. Igualmente importante, a disciplina de serviço deve ser propícia a uma eficiente implementação em nível de hardware, ou seja, simples o suficiente para implementação prática.

Justiça: A disciplina de serviço necessita prover alguma medida de isolamento entre múltiplos fluxos de pacotes que competem pela mesma banda compartilhada de saída. Em particular, cada fluxo deve receber sua justa parte da banda disponível e essa parte não deve ser afetada pela presença e (mau) comportamento de outros fluxos. Por exemplo, essa parte deve ser uma fração pré-alocada da banda que deve estar disponível ao fluxo independente da atividade de outros fluxos. A banda não-utilizada deve ser dividida entre os fluxos de tráfego ativos proporcionalmente aos seus requisitos.

Essa última questão é um tanto contraditória, pois apesar dos avanços da pesquisa nessa direção, não foi definida uma métrica aceita universalmente para

avaliá-la. Alguns autores utilizam o conceito de “Justiça Max-Min” (do inglês, “Max-min fairness”) [36] assim como outros questionam a veracidade de tal conceito alegando que o custo da utilização da rede por parte de um usuário afeta a utilização de outros fazendo com que esse custo tenha que ser também levado em consideração [40]. Neste trabalho o conceito de justiça será avaliado pela satisfação dos usuários, ou seja, se um usuário tem uma alta satisfação, independente da relação entre o serviço que ele contrata e o que ele recebe, ele estará satisfeito e sua percepção de justiça também.

O design de um algoritmo de escalonamento que tenha todas essas propriedades é reconhecido como uma tarefa complexa de ser resolvida [12]. Durante a evolução dos estudos das disciplinas de serviço de pacotes, observam-se duas abordagens principais: Disciplinas baseadas em prioridades (“Sorted-priority”) e em quadros (“Frame-based”).

Disciplinas baseadas em prioridades [23, 24, 25, 26, 27] designam um valor de prioridade para cada pacote para então serem transmitidos em ordem crescente de tais prioridades. Os pontos fortes em relação a essas disciplinas são suas boas propriedades de limites de atraso e justiça (“max-min fairness”), porém apresentam alta complexidade de implementação pois em cada instante de escalonamento, novas prioridades têm que ser calculadas para cada pacote consumindo tempo e memória de processamento. Geralmente a complexidade é da ordem do número de fluxos servidos pelo escalonador. Esse gargalo na complexidade de implementação, torna problemática a implementação prática, requerendo assim esquemas mais simples de escalonamento.

Nas disciplinas baseadas em quadros [28, 29], o tempo é dividido em quadros e os pacotes são inseridos nesses quadros sem exceder um máximo pré-definido. Os métodos dessas disciplinas são escaláveis já que o processamento dos pacotes tem tempo constante diferentemente das disciplinas baseadas em prioridades. Porém essas disciplinas tendem a oferecer um fraco limite de atraso e característica de rajada na saída.

Apesar do escalonamento de pacotes ser um conceito muito estudado em redes de computadores, há uma lacuna significativa entre escalonadores que oferecem desempenho comprovado e os que são implementáveis na prática.

Algumas disciplinas tentam se beneficiar desses dois atributos [29], porém muito há de se estudar para chegar a um consenso entre desempenho e escalabilidade.

A disciplina proposta nesta tese é uma disciplina baseada em prioridade onde por prioridade se entende o nível de insatisfação do usuário com o serviço recebido. Seu desempenho é comparado às disciplinas baseadas em prioridade “Self-Clocked Fair Queueing” [24] e as baseadas em quadros do tipo “Round Robin” e sua variação “Deficit Round Robin” [28].

A seguir, descreve-se o sistema aqui empregado em que tanto a disciplina proposta como as disciplinas alternativas acima listadas serão averiguadas. Também, cada uma destas disciplinas é detalhada assim como a função utilidade [10] utilizada na disciplina proposta.

4.1. Descrição do sistema:

O sistema proposto é ilustrado na figura abaixo e consiste de um modelo onde os roteadores de borda pertencente à rede de pacotes de uma Provedora de Serviço de Telecomunicações oferecem três classes de serviços, a saber, VoIP, FTP e HTTP a seus usuários.

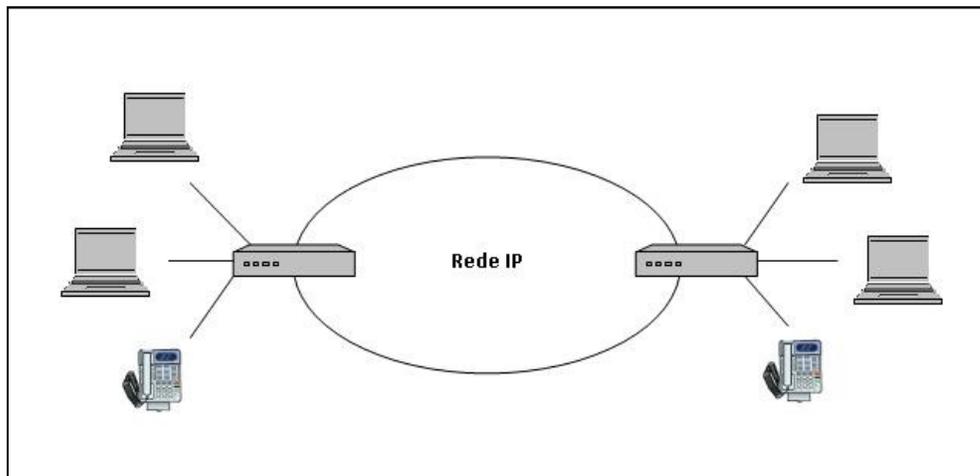


Figura 3 - Roteador de borda

Estes serviços também são diferenciados por tipos, ou seja, uma mesma classe de serviço pode oferecê-lo com parâmetros limites distintos, criando assim o conceito de tipo.

O sistema que descreve a atividade dos roteadores de borda consiste de “k” escalonadores iniciais e um escalonador mestre.

Cada usuário de uma determinada classe/tipo de serviço tem uma fila onde seus pacotes são inicialmente inseridos e que são administrados por um escalonador que foi denominado de “inicial”.

Este escalonador é uma entidade que obedece a um determinado algoritmo para determinar qual pacote de qual fila será escolhido e escoado. Então em um determinado instante de tempo, o escalonador analisa os pacotes de cada cabeça-de-fila que administra e de acordo com um algoritmo específico, escolhe qual pacote será escoado.

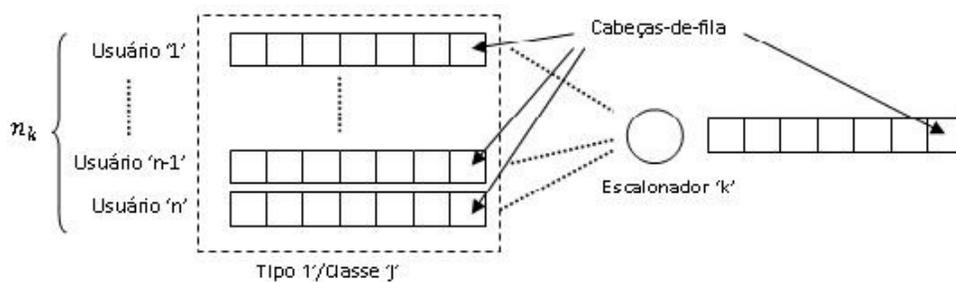


Figura 4 - Escalonador inicial ‘k’

No sistema proposto, existem dois níveis de escalonadores. No primeiro nível, existem tantos escalonadores quantos as classes/tipos de serviços. Esses escalonadores escolhem quais pacotes são escoados, analisando as filas de cada usuário de um mesmo tipo/classe de serviço. O termo “escoado” se referindo aos escalonadores de primeiro nível significa que o pacote escolhido será removido da fila de seu usuário e será inserido na fila de saída do escalonador que efetuou essa operação. Então cada escalonador também tem uma fila de pacotes.

No segundo nível, existe apenas um escalonador (denominado de “mestre”) que irá escolher entre os pacotes das filas dos escalonadores iniciais quais serão escoados. No contexto do escalonador mestre, este termo quer dizer que o pacote escolhido será removido da fila de seu escalonador inicial e será trafegado na rede. A escolha do pacote será baseada em estatísticas coletadas pelo monitor de tráfego.

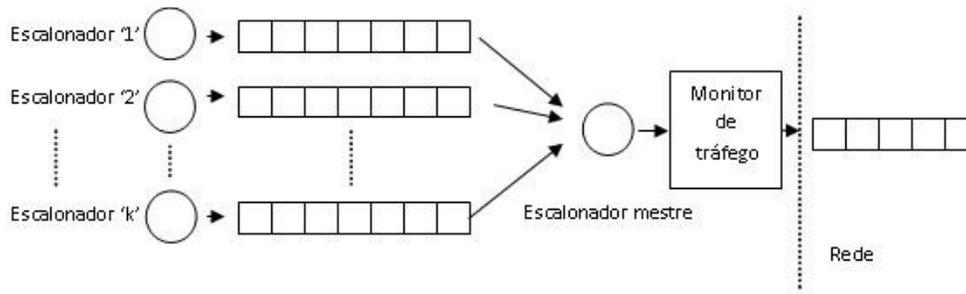


Figura 5 - Escalonador mestre

Os dois níveis de escalonadores configuram um sistema com escalonamento hierárquico que descrevem a atividade dos roteadores de borda, como mostra a figura seguinte.

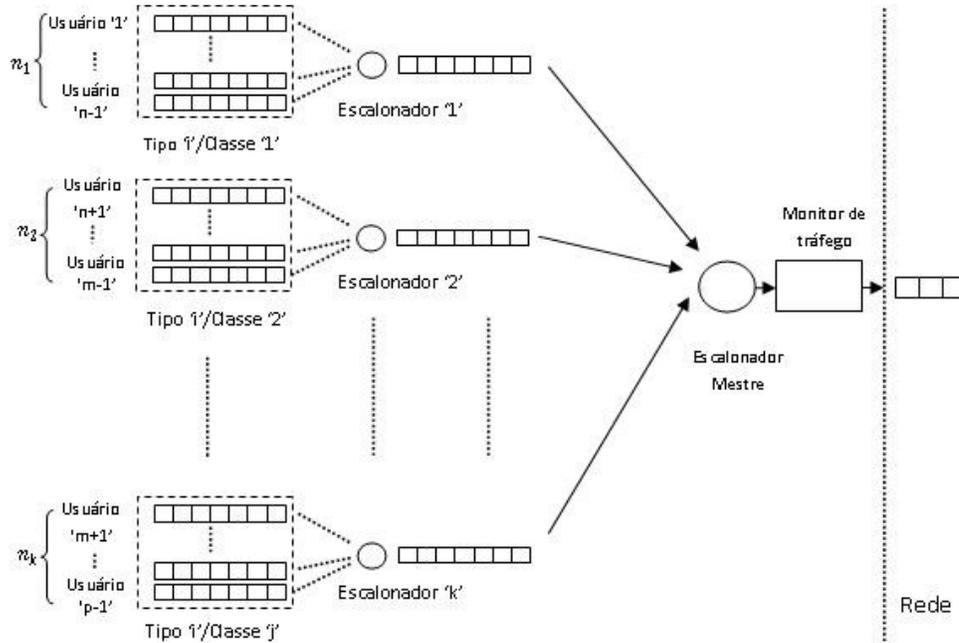


Figura 6 - Sistema dos roteadores de borda com escalonamento hierárquico

4.2. Funcionamento do sistema

Um usuário de um tipo/classe de serviço chega ao sistema competindo inicialmente com outros usuários de mesmo tipo/classe de serviço. Um usuário

“chegar ao sistema”, quer dizer inserir pacotes em sua fila para competir com os pacotes inseridos nas filas de usuários do mesmo tipo/classe de serviço. Quem vai decidir qual pacote de qual usuário será escolhido/escoado é o escalonador inicial correspondente. A regra de decisão usada é definida por um algoritmo de escalonamento. Será proposto mais adiante um algoritmo de escalonamento para comparação com outros algoritmos já existentes na literatura.

O pacote escolhido é removido da fila do usuário e inserido na fila de saída do escalonador inicial correspondente. Já no segundo nível do escalonamento, os pacotes que competirão por serviço serão os pacotes das filas dos escalonadores e não mais das filas dos usuários. Logo os pacotes de diferentes tipos/classes de serviço competirão por serviço nesse segundo estágio diferentemente do primeiro estágio quando os pacotes de um mesmo tipo/classe de serviço competiam entre si. O escalonador mestre de acordo com seu algoritmo de escalonamento escolhe entre os pacotes das cabeças-de-fila dos escalonadores, qual pacote será trafegado finalmente na rede.

O método de escolha dos pacotes é determinado pelo algoritmo de escalonamento. Como o próprio nome diz, algoritmo significa um conjunto de regras que são seguidas para chegar a um fim pré-estabelecido. No caso em questão é a definição de uma regra capaz de fazer uma escolha criteriosa a respeito do pacote a ser escoado dentre o conjunto de pacotes elegíveis.

O indicador a ser usado aqui nesta tese para determinar o pacote a ser escalonado foi trazido do domínio da economia e é chamado de “utilidade”. O conceito de utilidade [10] representa a satisfação ou prazer que os consumidores retiram do consumo de determinado bem ou serviço. Trazendo para o contexto do presente trabalho, é a satisfação que os usuários “sentem” com o serviço de telecomunicação contratado. Alguns autores já trouxeram este conceito para as telecomunicações [2-8], onde a utilidade dos usuários é relacionada com a probabilidade de aceitação do serviço, maximização de renda da provedora de serviço ou otimização da alocação de recursos. Nesse trabalho a utilidade é usada como regra de decisão sobre qual pacote de qual tipo/serviço de qual usuário será trafegado na rede visando à melhoria da alocação dos recursos e da satisfação dos usuários.

Porém esse indicador é difícil de ser medido fielmente para os usuários, pois utilidade é uma medida subjetiva dependendo de muitos aspectos impossíveis de serem analiticamente modelados tais como humor e sensibilidade.

Logo a utilidade no contexto do presente trabalho, é medida através de uma razão entre estatísticas normalizadas da rede medidas no instante de tempo da decisão sobre qual pacote será trafegado na rede (escalonador mestre). A forma como essa razão é calculada será detalhada posteriormente.

A medição dos parâmetros da rede de cada pacote é feita no monitor de tráfego após seu escalonamento pelo escalonador mestre e enviado ao coletor de estatísticas para calcular as estatísticas dos parâmetros medidos como mostra a figura a seguir.

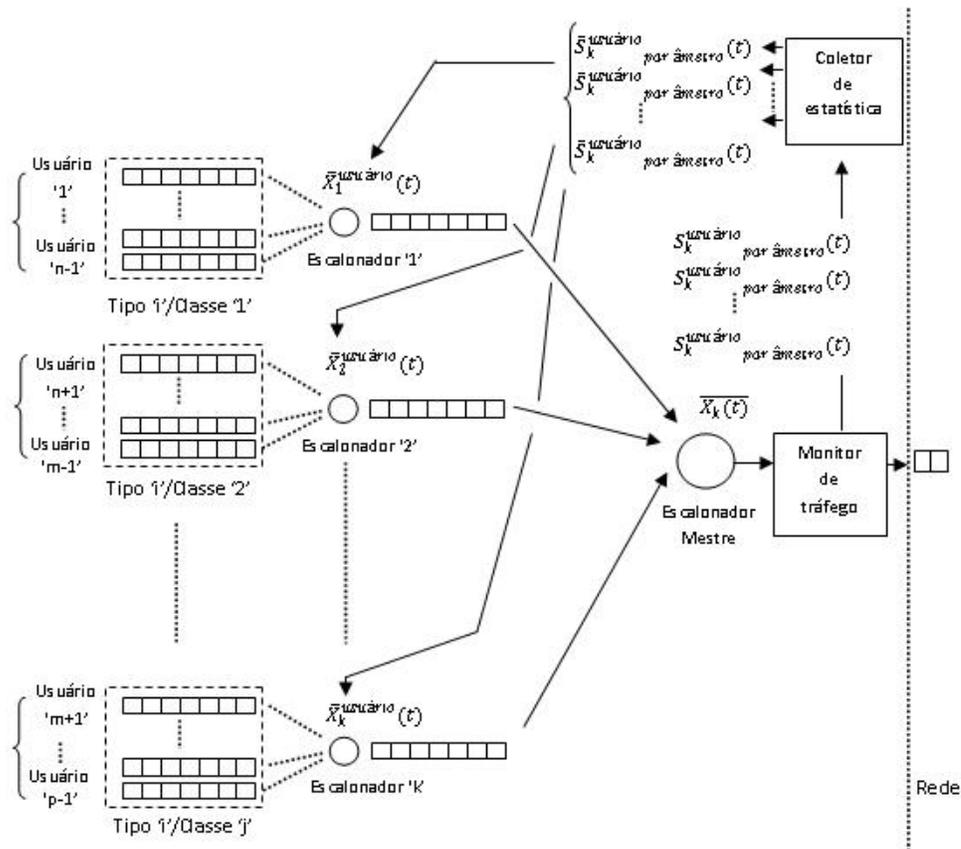


Figura 7 - Atualização dos escalonadores iniciais e mestre para cálculo da satisfação por usuários e por tipo/classe de serviço

Após coletadas as estatísticas, o coletor atualiza os parâmetros dos escalonadores iniciais para que estes possam calcular a utilidade média de cada usuário. Essas utilidades calculadas tanto pelos escalonadores iniciais quanto o escalonador mestre, são utilizadas para escalonar os pacotes dos usuários.

4.3. Modelagem de tráfego das classes de serviço

Como dito anteriormente, apenas três classes de serviço estão sendo consideradas neste trabalho: FTP, VoIP e HTTP e diversas fontes de tráfego têm sido utilizadas em inúmeros estudos onde a modelagem de tais tráfegos se faz necessária [30].

O modelo usado nesse trabalho, é o modelo de fonte de tráfego ON-OFF de acordo com [31]. Neste modelo o fluxo de pacotes de uma sessão é modelado como uma sucessão de períodos ativos (ON) e períodos de silêncio (OFF). A geração de pacotes (ou qualquer outra unidade de informação) ocorre apenas nos períodos de atividade. Normalmente assume-se que os períodos ON e OFF são independentes entre si e que suas durações são variáveis aleatórias com distribuição exponencial [31].

A seguir cada classe de serviço será detalhada assim como sua modelagem.

4.3.1. VoIP

Voz sobre IP, também chamado VoIP, telefonia IP, telefonia Internet, telefonia em banda larga e voz sobre banda larga é o roteamento de conversação humana usando a Internet ou qualquer outra rede de computadores baseada no Protocolo de Internet, tornando a transmissão de voz mais um dos serviços suportados pela rede de dados.

A proposta de modelagem do tráfego de VoIP seguida nesse trabalho está de acordo com [31]. O codificador de voz considerado é o AMR 12.2Kbps gerando pacotes de 244 bits.

Os parâmetros de tráfego estão descritos na tabela a seguir:

Parâmetro	Distribuição	Valor médio
Intervalo entre chegadas de usuários	Exponencial	Variável de acordo com o tipo
Sessão da chamada	Exponencial	{90} segundos
Tamanho do pacote	Fixo	30 bytes (~ 244 bits)
Tempo entre chegadas de pacotes	Fixo	20 ms
Tempo ativo de voz	Exponencial	2 segundos
Fator de atividade de voz	Fixo	0.5

Tabela 1 - Parâmetros de tráfego VoIP

Modelagem teórica:

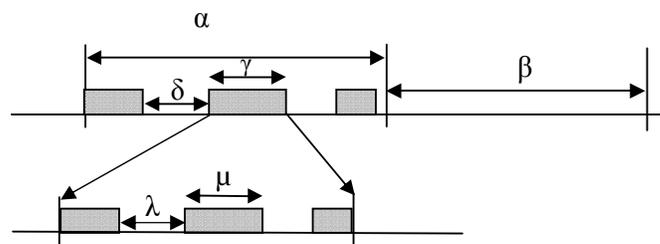


Figura 8 - Tráfego de VoIP teórico

α = Duração da sessão da chamada;

β = Intervalo entre chegadas dos usuários;

γ = Duração de atividade da voz;

δ = Tempo inativo;

μ = Duração do pacote;

λ = Tempo entre chegadas de pacotes;

s = Tamanho do pacote (Bytes);

r = Taxa de dados nominal (bps);

A = Fator de atividade: $A = \frac{\bar{\gamma}}{\gamma + \delta} = \frac{1}{1 + \delta/\bar{\gamma}}$.

A figura a seguir mostra um tráfego típico de VoIP gerado:

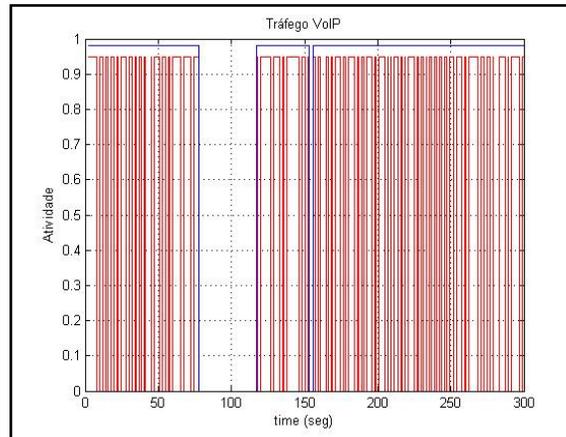


Figura 9 - Tráfego de VoIP gerado

A janela azul de duração ' α ' representa a duração da sessão da chamada, e a janela vermelha de duração ' γ ', representa os períodos de atividade da voz.

4.3.2. FTP

FTP significa "File Transfer Protocol" (Protocolo de Transferência de Arquivos) e é uma forma bastante rápida e versátil de transferir arquivos sendo uma das mais usadas na internet. Pode referir-se tanto ao protocolo quanto ao programa que o implementa.

O FTP é baseado no TCP [43], mas é anterior à pilha de protocolos TCP/IP, sendo posteriormente adaptado para o TCP/IP. É o padrão da pilha TCP/IP para transferir arquivos, sendo um protocolo genérico, independente de hardware e do sistema operacional, tendo em conta restrições de acesso e propriedades dos mesmos.

O tráfego FTP nesse trabalho é referenciado como uma transferência de arquivo entre usuários e servidores de uma prestadora de serviços de telecomunicação. Não está sendo aqui considerado o seu funcionamento sob o protocolo de transporte TCP, que é deixado assim como proposta para trabalhos futuros.

A modelagem aqui proposta do tráfego FTP é a mesma adotada em [31]. A tabela a seguir mostra os parâmetros e suas características utilizados no trabalho.

Parâmetro	Distribuição	Valor médio
Intervalo entre chegadas de usuários	Exponencial	Variável de acordo com o tipo
Tamanho do arquivo	Exponencial	{500} Kbytes

Tabela 2 - Parâmetros de tráfego FTP

Modelagem teórica:

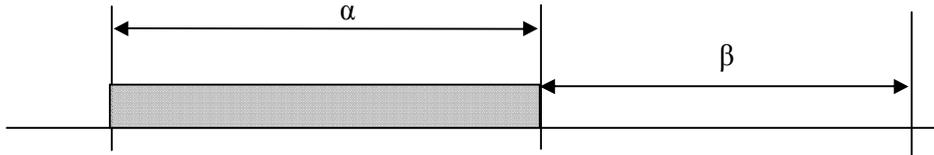


Figura 10 - Tráfego de FTP teórico

$$\alpha = \text{Tempo de serviço} = \bar{\alpha} = \frac{8.s}{r};$$

β = Intervalo entre chegadas de usuários;

= Tamanho médio do arquivo (bytes);

r = taxa de dados nominal (bps).

A figura a seguir mostra o tráfego de FTP gerado:

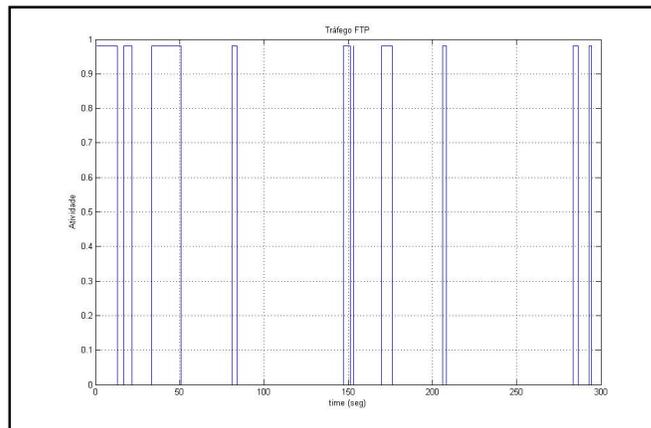


Figura 11 - Tráfego de FTP gerado

A janela azul de duração ‘ α ’ representa a sessão da transferência (ou equivalentemente o tamanho) do arquivo.

4.3.3. HTTP

O HyperText Transfer Protocol (HTTP) é um protocolo de aplicação responsável pelo tratamento de pedidos/respostas entre cliente e servidor na World Wide Web. Ele surgiu da necessidade de distribuir informações pela Internet e para que essa distribuição fosse possível, foi necessário criar uma forma padronizada de comunicação entre os clientes e os servidores da Web e entendida por todos os computadores ligados à Internet. Com isso, o protocolo HTTP passou a ser utilizado para a comunicação entre computadores na Internet e a especificar como seriam realizadas as transações entre clientes e servidores, através do uso de regras básicas. Em resumo, é o tráfego correspondente a navegação de um usuário pela internet e ao de acesso a e-mails via Web.

A proposta de modelagem do tráfego de HTTP seguida nesse trabalho está de acordo com a descrita em [31]. A tabela a seguir mostra os parâmetros e suas características utilizadas no trabalho.

Parâmetro	Distribuição	Valor médio	Variância
Intervalo entre chegadas de usuários	Exponencial	Variável de acordo com o tipo	-
Tempo de navegação/duração da sessão	Exponencial	{60} segundos	-
Tamanho mínimo do objeto (Ω_{\min})	Fixo	1000 Bytes	-
Tamanho máximo do objeto (Ω_{\max})	Fixo	2×10^5 Bytes	-
Parâmetro do tamanho do objeto (X)	Normal	0	1
Tempo de leitura entre objetos	Exponencial	{5} segundos	-

Tabela 3 - Parâmetros de tráfego HTTP

Durante o tempo de navegação dos usuários na internet, eles baixam páginas da web que serão consideradas como objetos. O tamanho do objeto é aleatório e aqui modelado pela seguinte equação:

$\Omega = \min(\Omega_{\max}, \Omega_{\min} + 10^{\mu + \sigma X})$; onde Ω_{\min} e Ω_{\max} são 10^3 e 2×10^5 bytes, respectivamente; X é uma variável aleatória gaussiana com média zero e variância

unitária; μ e σ são 3.605 e 0.7839 respectivamente, especificando a distribuição do tamanho da página web.

Modelagem teórica:

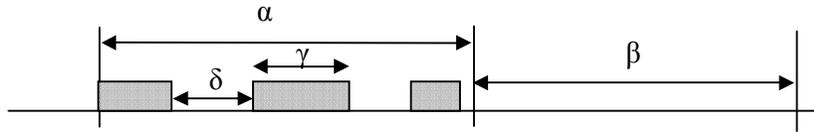


Figura 12 - Tráfego de HTTP teórico

α = Tempo de navegação/duração da sessão;

β = Intervalo entre chegadas de usuários;

γ = Duração da transferência dos pacotes de cada objeto;

δ = Tempo de leitura entre objetos;

A figura a seguir mostra o tráfego de HTTP gerado:

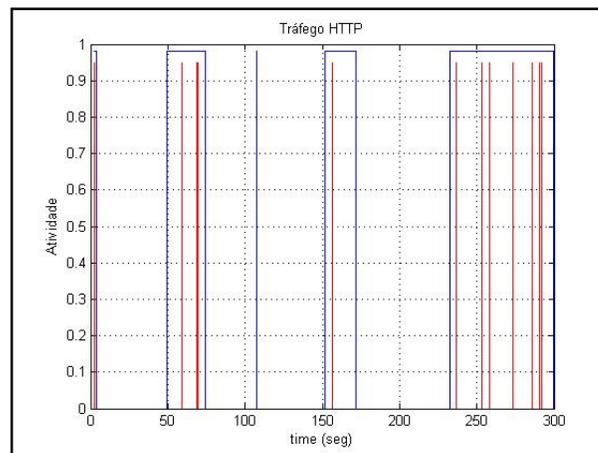


Figura 13 - Tráfego de HTTP gerado

A janela azul de duração ' α ' representa a navegação/duração da sessão, e a janela vermelha de duração ' γ ' representa a transferência dos pacotes de cada objeto.

A seguir é definida a disciplina baseada na Satisfação assim como a função utilidade [10], para determinar a satisfação dos usuários ao adquirir um serviço de telecomunicação, peça fundamental desta disciplina de serviço aqui proposta.

4.4. Disciplina de Serviço baseada na Satisfação

Antes de descrever a disciplina baseada na satisfação, será desenvolvido o conceito de satisfação utilizada nesse trabalho.

4.4.1. Função Utilidade

A função utilidade, como dito anteriormente, mede a satisfação dos usuários com o serviço de telecomunicação contratado e é uma medida subjetiva difícil de ser modelada analiticamente levando em conta apenas o ponto-de-vista dos usuários.

A função utilidade nesse trabalho pretende descrever a satisfação dos usuários baseado nos parâmetros técnicos da rede que vão determinar a qualidade do serviço contratado. Em resumo, quanto maior qualidade no serviço recebido, mais satisfeito o usuário ficará.

As estatísticas coletadas pelo monitor de tráfego são relativas aos parâmetros técnicos da rede que estão relacionadas ao conceito de Qualidade de Serviço. Como a utilidade tem a função de “descrever a satisfação dos usuários”, um usuário satisfeito com o serviço contratado é um usuário que obtém um serviço com parâmetros técnicos satisfatórios para seu tipo/classe de serviço contratado. Então nada mais justo do que definir utilidade baseado em parâmetros da rede que indiquem a qualidade do serviço contratado no instante do escalonamento dos pacotes.

Nesse trabalho a função utilidade é definida como o produto das estatísticas médias normalizadas de três parâmetros técnicos da rede. Esses parâmetros da rede são: taxa transmissão de dados, atraso (delay) e jitter (variância do atraso).

Mas criar uma função com esses dados em sua forma bruta é inadequado porque são grandezas com faixas dinâmicas diferentes, requerendo assim alguma forma de normalização dos mesmos.

O serviço de VoIP é mais sensível à atrasos e jitter do que os serviços de FTP e HTTP, logo os limites aceitáveis desses parâmetros para VoIP são menores do que os limites de FTP e HTTP.

Então a normalização é a razão entre a estatística média do parâmetro medido sob o valor nominal do parâmetro contratado no caso da taxa de transmissão de dados e o valor limite aceitável no caso do atraso e jitter. A tabela a seguir mostra tais valores.

A tabela a seguir mostra os parâmetros que foram adotados nesta tese para os cenários a serem simulados.

Classe 'i'/Tipo 'j' de serviço	Taxa de dados contratada (Kbps)	Atraso (ms)	Jitter (ms)
FTP (1/1)	2000	4000	1000
FTP (1/2)	3000	4000	1000
VoIP (2/1)	12.4	400	100
HTTP (3/1)	4000	4000	1000

Tabela 4 - Valores nominais/limites dos parâmetros técnicos da rede

A função utilidade média de cada usuário é, então, definida por:

$$\bar{X}_k^{usuário}(t) = \bar{X}_k^{usuário}_{atraso}(t) * \bar{X}_k^{usuário}_{taxa}(t) * \bar{X}_k^{usuário}_{jitter}(t);$$

As estatísticas médias normalizadas $\bar{X}_k^{usuário}_{atraso}(t)$, $\bar{X}_k^{usuário}_{taxa}(t)$ e $\bar{X}_k^{usuário}_{jitter}(t)$ são calculadas da seguinte forma:

$$\bar{X}_k^{usuário}_{atraso}(t) = 1 - \text{Min}\left(1, \frac{s_k^{usuário}_{atraso}(t)}{\text{AtrasoLimite}_{ij}}\right)^2;$$

Observe que quanto maior a razão entre o atraso medido e o limite, menos satisfeito o usuário estará e se o atraso medido for maior que o limite ele estará 100% “insatisfeito”, equivalendo à satisfação zero. Essa medida é inversamente proporcional à satisfação.

² Sendo “i” o tipo de serviço e “j” a classe de serviço

$$\bar{X}_k^{usuário}_{taxa}(t) = \text{Min} \left(1, \frac{\bar{S}_k^{usuário}_{taxa}(t)}{\text{TaxaContratada}_{ij}} \right);$$

Neste caso, quanto maior a razão entre a taxa medida e a contratada, mais satisfeito o usuário estará, pois estará recebendo uma taxa mais próxima (ou superior) à sua taxa contratada. Se a taxa recebida for maior que a contratada estará 100% satisfeito. Essa medida é diretamente proporcional à satisfação.

$$\bar{X}_k^{usuário}_{jitter}(t) = 1 - \text{Min} \left(1, \frac{\bar{S}_k^{usuário}_{jitter}(t)}{\text{JitterLimite}_{ij}} \right)$$

Aqui, a semelhança do caso “atraso”, quanto maior a razão entre o jitter medido e o limite, menos satisfeito o usuário estará e se o jitter medido for maior que o limite ele estará 100% “insatisfeito”, novamente equivalendo à satisfação zero. Essa medida é inversamente proporcional à satisfação.

As estatísticas médias dos parâmetros da rede são calculadas da seguinte maneira:

$\bar{S}_k^{usuário}_{atraso}(t)$: É calculado através da diferença entre o instante de geração do pacote e o instante em que é escolhido para escalonamento no escalonador mestre. O atraso médio é calculado como a razão entre a soma dos atrasos dos pacotes pela quantidade de pacotes trafegados na rede de uma mesma sessão de um mesmo usuário.

$\bar{S}_k^{usuário}_{taxa}(t)$: A taxa de dados oferecida no instante de escalonamento é calculada através da razão entre o produto do tamanho do pacote em bits da classe de serviço e o número de pacotes trafegados até o instante atual da mesma rajada de tráfego sobre a diferença entre o instante de tempo atual e o instante de início da atual rajada de tráfego. Rajadas de tráfego de uma mesma sessão de um mesmo usuário.

$\bar{S}_k^{usuário}_{jitter}(t)$: “Jitter” por definição é a medida da variância do atraso e é calculado como o valor absoluto da diferença do atraso atual em relação ao atraso médio dos pacotes anteriores da mesma sessão de um mesmo usuário.

Resumindo:

O monitor de tráfego mede os parâmetros da rede para o pacote escalonado e envia ao coletor de estatísticas que calcula as estatísticas dos parâmetros medidos. Essas estatísticas são enviadas aos escalonadores iniciais referentes ao pacote escalonado para que a função utilidade do usuário correspondente possa ser calculada.

O escalonador mestre calcula a função utilidade média de cada escalonador 'k' baseado nas utilidades médias dos usuários que pertencem a tal escalonador.

A função utilidade média de cada escalonador 'k' é definida por:

$$\bar{X}_k(t) = \left(\frac{1}{n_k}\right) * \sum \bar{X}_k^{usuário}(t);$$
 para todos os 'n_k' usuários do tipo 'i'/classe 'j' pertencentes ao escalonador 'k'.

Na sessão seguinte a disciplina de serviço que leva em conta essas utilidades calculadas será descrita em mais detalhes.

4.4.2. Descrição da disciplina empregada

A disciplina aqui proposta considera diferentes classes de serviço e tenta balancear a alocação dos recursos e a satisfação dos usuários baseado em parâmetros técnicos da rede medidos para cada classe de serviço considerada.

Esta disciplina opera baseada nos seguintes fundamentos: a cada fluxo de pacotes de um usuário é designado uma fila e a divisão da banda é baseada no inverso da utilidade dos usuários com o serviço recebido, ou seja, sua insatisfação.

O diferencial da disciplina proposta é que o peso dado a cada fila de pacotes não é determinado unicamente pelos requerimentos de banda e sim pelo inverso da satisfação média baseada em parâmetros técnicos da rede descrita pelo inverso da função utilidade.

A idéia subjacente dessa disciplina é a seguinte: uma provedora de serviços de telecomunicações deseja, pelo menos teoricamente, que todos os seus usuários estejam satisfeitos com o serviço recebido e uma das mais importantes válvulas propulsoras desse “desejo” é a disciplina de serviço, pois é ela quem vai decidir qual usuário será ofertado serviço em um determinado instante de tempo. Então quando todos os escalonadores calcularem a utilidade média de seus usuários e dos tipos/classes de serviço, não adianta priorizar os usuários que estejam com uma utilidade média alta, pois eles teoricamente já estão satisfeitos com o serviço. Os usuários ou tipos/classes de serviço que estejam com utilidade média baixa, se não forem priorizados, ficarão com uma utilidade média ainda mais baixa, fato indesejável a qualquer provedora de serviços de telecomunicações. Então os escalonadores deverão priorizar os usuários e tipos/classes de serviço que estejam com utilidades médias baixas, ou seja, aqueles que estejam com “maior insatisfação média”.

Como o sistema em questão utiliza um esquema de escalonamento hierárquico, a atribuição dos pesos para divisão de banda ocorrerá em dois pontos do sistema: nos ‘k’ escalonadores iniciais e no escalonador mestre. No primeiro estágio ocorrerá o escalonamento “intra-tipo/classe de serviço” e no segundo estágio “inter-tipo/classe de serviço”.

A atribuição dos pesos para os ‘k’ escalonadores iniciais se dá em dois estágios. Como o peso é definido pela insatisfação média dos usuários, nos primeiros instantes de tempo de funcionamento do sistema, não há estatísticas suficientes para descrever a insatisfação dos usuários com o serviço recebido. Logo é definido um limite chamado ‘Npks_th’, que indica quantos pacotes de cada usuário precisam ser trafegados para que o coletor possa coletar estatísticas suficientes para descrever a insatisfação média dos usuários. Durante esse período em que o coletor está apenas coletando estatísticas, aos usuários são atribuídas $\bar{X}_k^{usuário}(t) = 0$, ou seja, 100% de insatisfação e peso = 1 para cada usuário. No segundo estágio, quando o limite ‘Npks_th’ é ultrapassado, já existem estatísticas suficientes para determinar a insatisfação média dos usuários e o valor de $\bar{X}_k^{usuário}(t)$ é calculado de acordo com a função utilidade descrita anteriormente.

Então o ‘k-ésimo’ escalonador realiza o seguinte algoritmo:

1. Para cada usuário de seus ‘ n_k ’ usuários, calcula-se a insatisfação média por usuário ($1 - \bar{X}_k^{usuário}(t)$).
2. Se existe um usuário com maior insatisfação média em relação a todos os outros de sua classe/tipo de serviço, esse usuário é escolhido e o algoritmo continua no item oito. Se não, seguem os próximos passos;
3. Consideram-se apenas os usuários com as maiores insatisfações médias calculadas no item anterior. Suas insatisfações são somadas para criação da grandeza

$$\sum_{\forall "usuários\ mais\ insatisfeitos" \in k} (1 - \bar{X}_k^{usuário}(t)).$$

4. É então calculada uma nova insatisfação média normalizada para cada usuário considerado no item três, ou seja, razão entre a insatisfação média do usuário e a grandeza calculada referente à soma das insatisfações dos usuários mais insatisfeitos do tipo ‘i’/serviço ‘j’ pertencente ao escalonador ‘k’.
 5. Essas insatisfações médias normalizadas são somadas para preencherem todo o intervalo entre 0 a
- $$\sum_{\forall "usuários\ mais\ insatisfeitos" \in k} (1 - \bar{X}_k^{usuário}(t))$$
- dos usuários considerados no item três.
6. Gera-se um número aleatório entre 0 e 1;
 7. Checa-se em qual intervalo esse número aleatório está contido e escolhe o usuário que pertença a esse intervalo.
 8. O pacote do usuário escolhido é retirado de sua fila e inserido na fila do escalonador ‘k’ que o escolheu;
 9. Agenda o funcionamento do escalonador mestre.

A figura 14 ilustra o procedimento em que um único usuário tem a maior insatisfação média entre os usuários de seu mesmo tipo ‘i’/serviço ‘j’ e pertencente ao mesmo escalonador ‘k’.

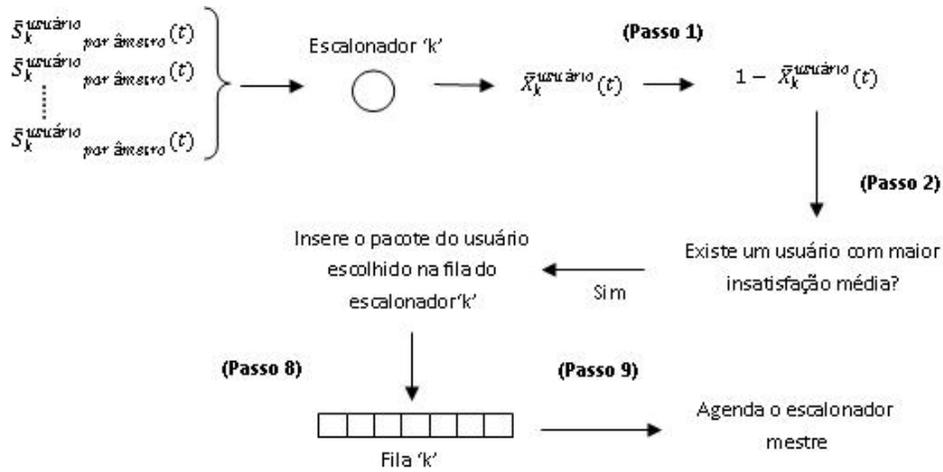


Figura 14 - Escalonamento no escalonador inicial 'k' (Caso 1)

A figura 15 ilustra o procedimento do escalonador inicial quando existe mais de um usuário com maior insatisfação média. No caso em particular mostrado na figura, todos os usuários do mesmo do tipo 'i'/serviço 'j' pertencente ao escalonador 'k', estão igualmente insatisfeitos.

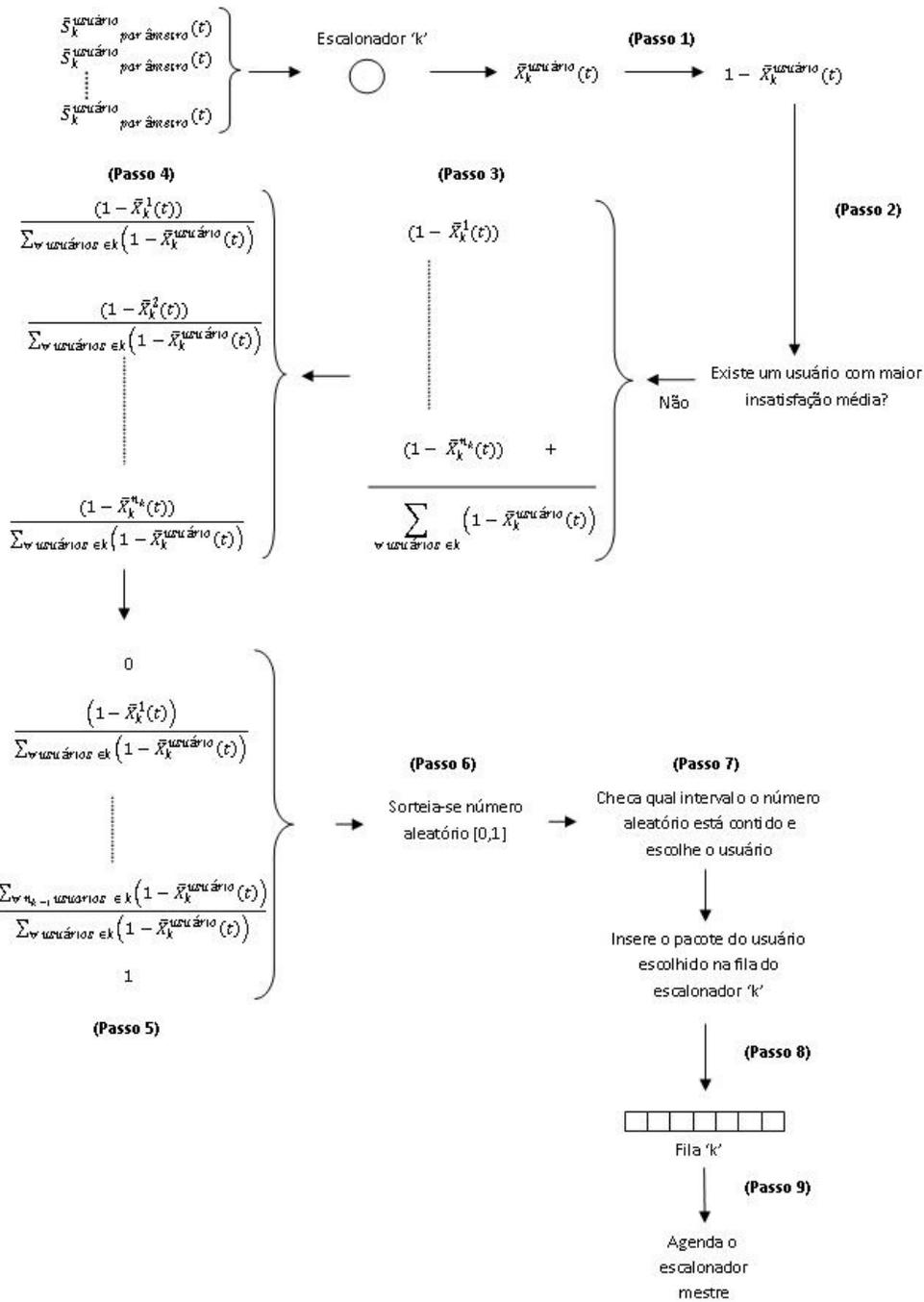


Figura 15- Escalonamento no escalonador inicial 'k' (Caso 2)

Depois das atribuições dos pesos, do escalonamento inicial dos pacotes e da inserção dos pacotes nas filas dos 'k' escalonadores, o escalonador mestre tem condições de implementar seu algoritmo para decidir qual pacote das 'k' filas será finalmente trafegado na rede.

Nesse estágio o escalonamento é “inter-tipo/classe de serviço”. Esse é parte que requer mais atenção, pois diferentes classes de serviços com diferentes requisitos de qualidade estão competindo por serviço, o que é uma tarefa crítica para o bom desempenho da rede.

Então o escalonador mestre realiza o seguinte algoritmo:

1. Calcula a insatisfação média por escalonador inicial ‘k’, ou seja, calcula a média das insatisfações médias dos usuários pertencentes à fila ‘k’:

$$1 - \bar{X}_k(t) = \frac{\sum_{\forall \text{usuários} \in \text{fila } k} (1 - X_k^{\text{usuário}})}{nr.\text{usuários} \in \text{fila } k}, p/k = 1, 2, \dots, k;$$

2. Se existir um escalonador ‘k’ com a maior insatisfação média dos usuários, o pacote deste escalonador é escolhido e segue o passo 8. Senão seguem os próximos passos;

3. Consideram-se os escalonadores (que se referem a um tipo/classe de serviço) com as maiores insatisfações médias. Suas insatisfações médias são somadas para criação da seguinte grandeza:

$$\sum_{\forall k \text{ escalonadores com maiores insatisfações médias}} (1 - \bar{X}_k(t))$$

4. As insatisfações médias de cada escalonador é normalizada em função da grandeza criada no item 3;

5. Essas insatisfações médias normalizadas são somadas para preencherem todo o intervalo entre 0 e

$$\sum_{\forall k \text{ escalonadores com maiores insatisfações médias}} (1 - \bar{X}_k(t)) \text{ dos escalonadores considerados no item três.}$$

6. Gera-se um número aleatório entre 0 e 1;
7. Checa-se em qual intervalo esse número aleatório está contido e escolhe o escalonador que pertença a esse intervalo.
8. Retira-se este pacote da fila escolhida, o monitor de tráfego recalcula os parâmetros técnicos da rede relacionados ao pacote e envia o pacote selecionado para a rede.

A figura 16 ilustra este procedimento do escalonador mestre quando há apenas um escalonador (tipo/classe de serviço) com maior insatisfação média.

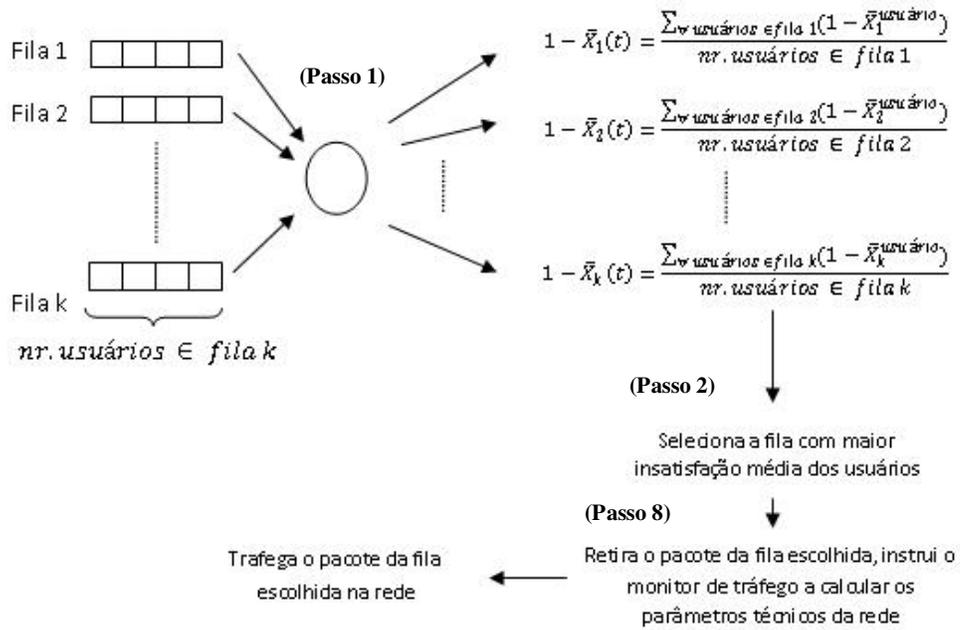


Figura 16 - Escalonamento no escalonador mestre (caso 1)

A figura 17 seguinte, ilustra o caso em que mais de um escalonador (tip/classe de serviço) apresenta a maior insatisfação média. No caso da figura, todos os escalonadores têm insatisfações médias iguais.

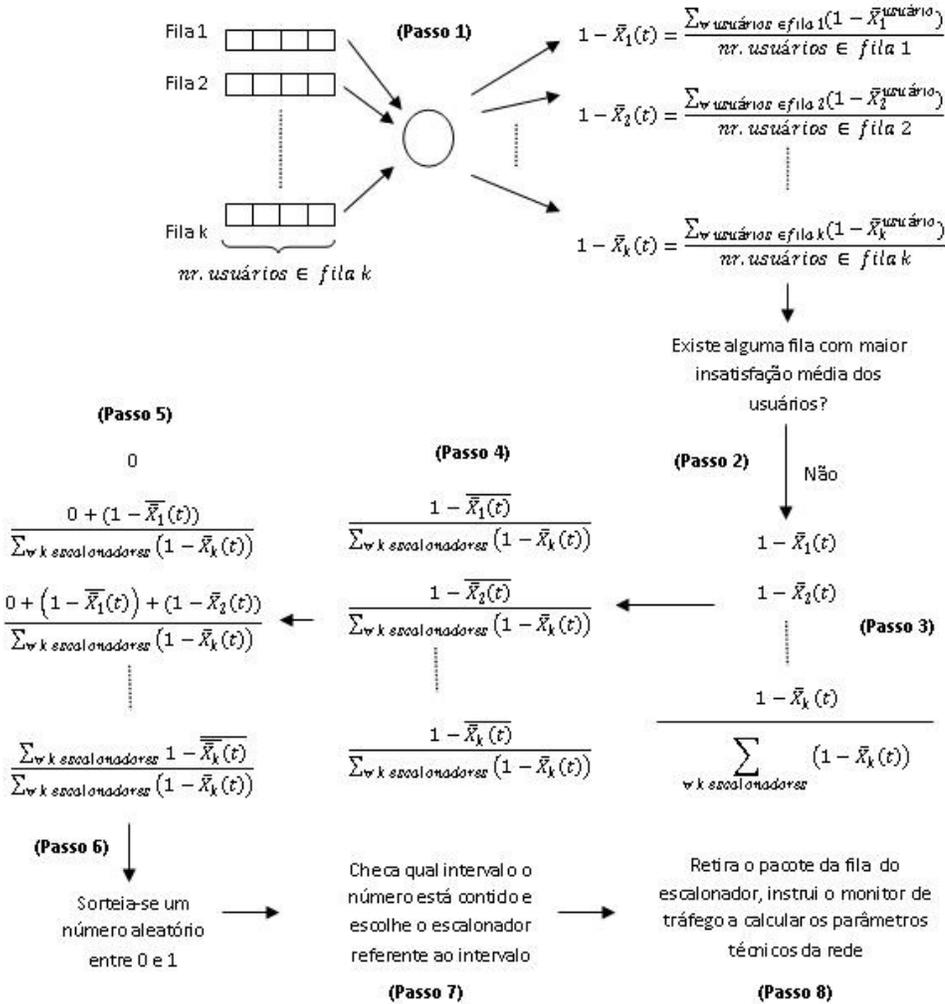


Figura 17 - Escalonamento no escalonador mestre (caso 2)

4.5. Disciplinas “Frame-Based”

As disciplinas do tipo “Frame-Based”, como já dito anteriormente, são as disciplinas que tomam a decisão a cerca do pacote a ser escolhido baseado em um tamanho máximo de “frame” pré-definido. Em geral, essas disciplinas tem baixa complexidade de implementação, não garantem limites de atraso e não costumam ser justas na alocação de recursos aos usuários. O “Round-Robin” tradicional é a seguir descrito assim como sua variação “Deficit Round Robin” [28].

4.5.1. Round Robin

“Round Robin” é uma das mais antigas e simples disciplinas de serviço. É largamente usado, e foi projetado especialmente para sistemas do tipo *time-sharing*.

Um escalonador que utiliza o escalonamento “Round Robin”, tem filas separadas para cada fluxo de tráfego. O algoritmo seleciona para escoamento um fluxo de tráfego ativo (que contém pacotes em sua fila) para que transmita no canal compartilhado de uma maneira repetitiva e periódica. Quando uma fila está vazia (sem pacotes), é dada a vez para a próxima fila para transmitir.

A figura a seguir mostra o funcionamento do algoritmo no sistema proposto.

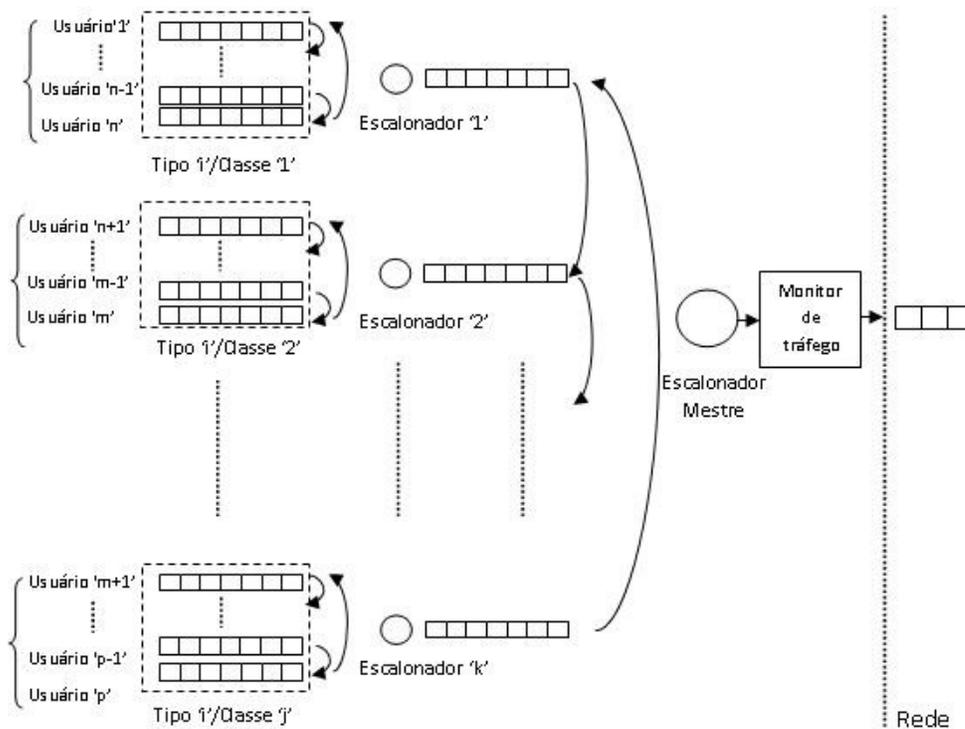


Figura 18 - Disciplina Round Robin

4.5.2. Deficit Round Robin

“Deficit Round Robin” (DRR) [28], é uma variação do esquema “Round Robin” que suporta pacotes de tamanhos variáveis sem precisar do conhecimento de seu tamanho médio. Nesse algoritmo, uma “janela de transmissão” é definida como se fosse um tamanho máximo de pacote. Cada vez que o escalonador seleciona uma particular fila, dela são removidos o maior número de pacotes cuja soma de tamanhos seja inferior ao da janela de transmissão. Estes pacotes assim selecionados são então transmitidos.

À semelhança do algoritmo Round Robin, este esquema é realizado de forma rotativa nas filas seguintes, lembrando apenas que quando uma fila tiver vazia, passa-se automaticamente para a seguinte.

Esse algoritmo, dependendo da implementação, pode ser configurado para priorizar certas classes de serviços ou prover uma reserva de banda justa entre diferentes classes de serviços.

Neste trabalho, a disciplina é configurada para prover bandas iguais às diferentes classes de serviço, ou seja, é determinada uma janela de transmissão igual para todas as classes. De certa forma o tráfego de VoIP acaba sendo priorizado em relação às outras classes de serviços por ser um serviço em tempo real com pacotes com “payloads” menores e por necessitar de métricas de QoS mais precisas.

Esse algoritmo é implementado no escalonador mestre, pois somente nesse estágio diferentes classes de serviços com diferentes tamanhos de pacotes estão competindo por serviço. Nos escalonadores iniciais, o Round-Robin tradicional é implementado, pois não há priorização de usuários da mesma classe de serviço e os pacotes tem o mesmo tamanho.

A tabela seguinte mostra o tamanho adotado nesta tese para os pacotes por classe de serviço e suas respectivas janelas de transmissão.

Classe de serviço	Tamanho do pacote (bytes)	Janela de transmissão (bytes)
VoIP	30	3000
FTP	1500	3000
HTTP	1000	3000

Tabela 5 - Janelas de transmissão dos pacotes por classes de serviço

Como mostra a tabela 6, bandas iguais são oferecidas para todas as classes de serviço aqui consideradas. Considerando assim, uma variante do “Round Robin” onde a banda é igualmente compartilhada entre as classes de serviço pois a janela de transmissão tem tamanhos iguais e se houverem pacotes nas filas suficientes para “encher” a janela, cada classe de serviço envia a mesma quantidade de dados (banda) por vez.

A figura a seguir mostra o funcionamento dessa disciplina no escalonador mestre já que nos escalonadores iniciais o “Round Robin” tradicional é utilizado.

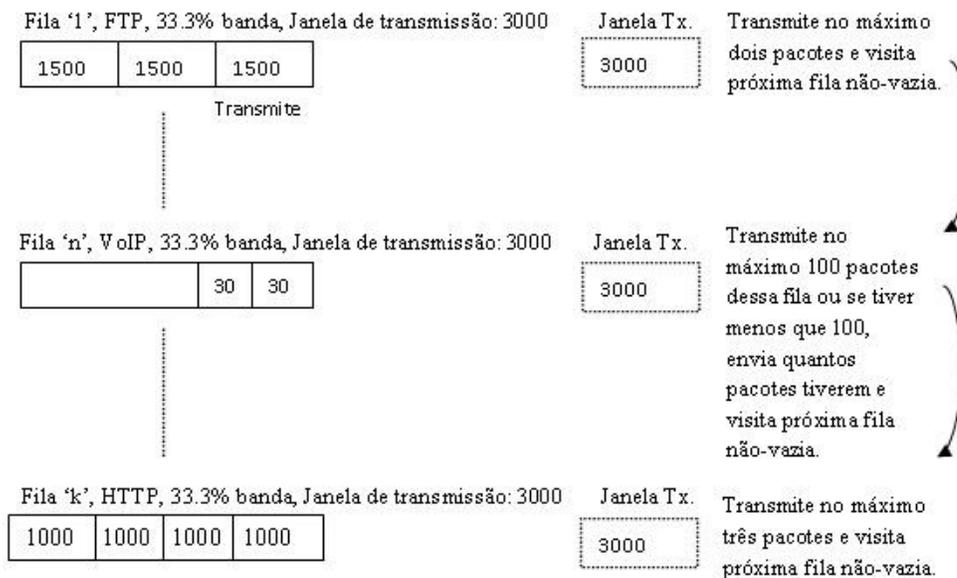


Figura 19 - Disciplina Deficit Round Robin 1

No caso acima, como a janela de transmissão suporta no máximo dois pacotes FTP, a disciplina transmite os dois pacotes e passa a vez para a próxima fila não-vazia (VoIP). Como há dois pacotes na fila, a disciplina transmite esses dois pacotes sendo que poderia transmitir no máximo cem pacotes e passa a vez para a próxima fila não-vazia.

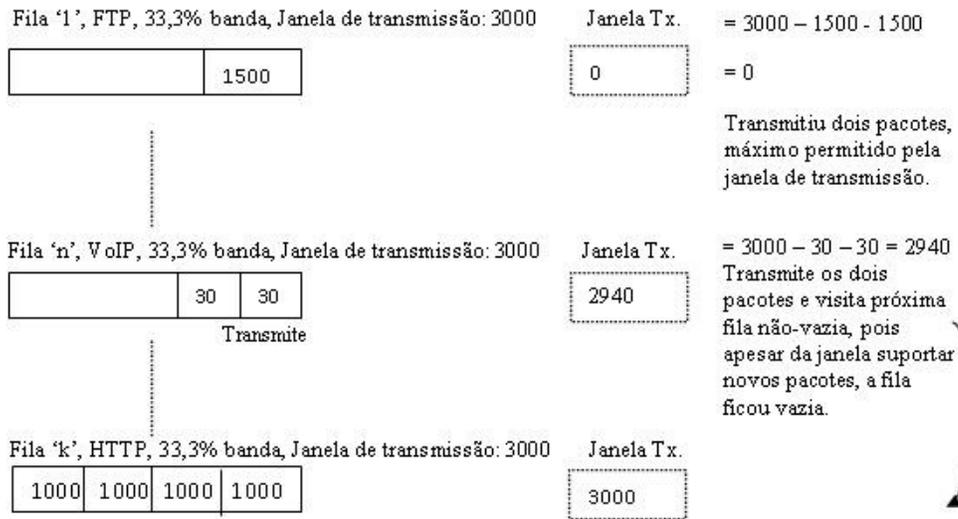


Figura 20 - Disciplina Deficit Round Robin 2

Quando o escalonador visita a fila de HTTP, é possível a transmitir no máximo três pacotes. Como há mais que três pacotes enfileirados, são transmitidos três pacotes e a próxima fila não-vazia é visitada.

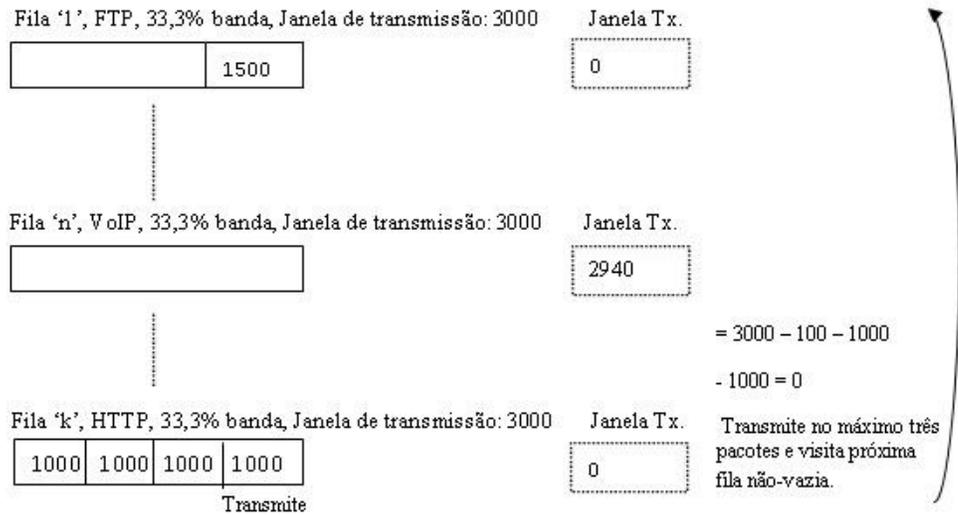


Figura 21 - Disciplina Deficit Round Robin 3

Quando todas as filas não-vazias são visitadas, o escalonador volta para a primeira fila não-vazia pertencente ao escalonador mestre. Quando há pacotes enfileirados, a janela de transmissão é atualizada somando o valor residual (caso a janela de transmissão não tenha sido totalmente utilizada na rodada anterior) com

o valor nominal da janela de transmissão (3000). Se não há pacotes enfileirados, a janela é atualizada com valor nulo.

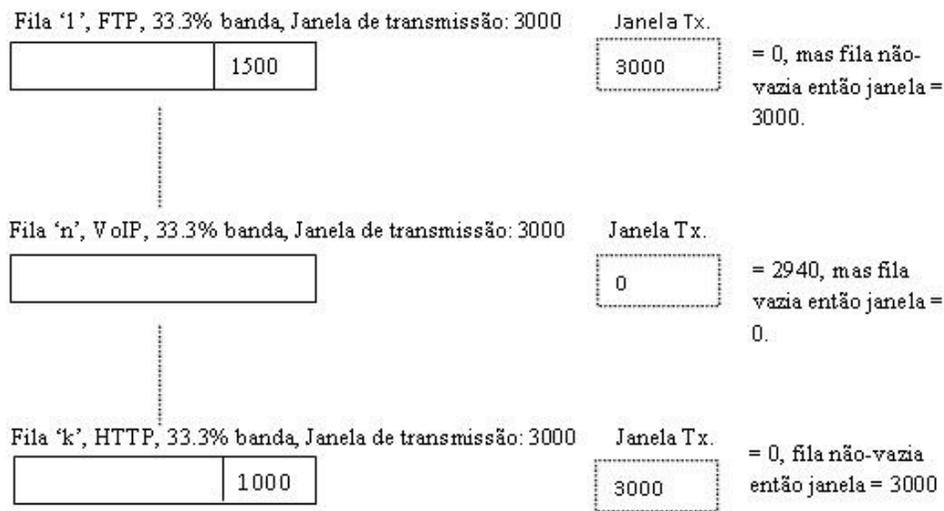


Figura 22 - Disciplina Deficit Round Robin 4

O valor da janela da transmissão depende da implementação do sistema, logo os valores acima podem ser modificados dependendo da classe de serviço que se queira priorizar.

4.6. Disciplinas “Sorted-Priority”

As disciplinas “Sorted-Priority”, como já dito anteriormente, são as disciplinas que tomam a decisão a cerca do pacote a ser escalonado baseado em prioridades dos mesmos. Para o detalhamento da disciplina “Self-Clocked Fair Queuing” [24], é necessário um entendimento da disciplina “Fluid-Flow Fair Queueing” [35] que é uma disciplina que utiliza um algoritmo ideal e não-implementável usado como referência para o desenvolvimento de outras disciplinas de serviço baseada em prioridades. A disciplina “Fluid-Flow Fair Queueing” é detalhada no Apêndice e a disciplina “Self-Clocked Fair Queuing” é detalhada a seguir.

4.6.1. Self-Clocked Fair Queuing (SCFQ)

A disciplina “Self-Clocked Fair Queuing” é um esquema alternativo muito mais simples que visa reduzir a complexidade no cálculo dos tempos virtuais e ainda fornece um desempenho desejável.

Essa disciplina também utiliza a noção de tempo virtual, que administra o progresso do sistema “FFQ”. A diferença é que esta, ao invés de usar uma definição analítica abstrata para o instante de tempo virtual, usufrui de uma quantidade que surge naturalmente durante o progresso do algoritmo para atualizá-lo.

Para compreensão da idéia, note que na disciplina “WFQ” o instante virtual de fim de serviço corresponde ao instante em que o pacote concluiria o serviço no sistema “FFQ”. Isto sugere que o instante virtual de fim de serviço em qualquer instante “ t ” poderia ser estimado pelo próprio instante virtual de fim de serviço do pacote em serviço naquele determinado instante. Levando isso em consideração, a disciplina “SCFQ” é definida através do seguinte algoritmo:

1. Em cada pacote que chega ao sistema, é guardado o instante de fim de serviço deste pacote antes que sejam inseridos em suas filas correspondentes. A cada instante de escalonamento, os pacotes são escalonados para serviço em ordem crescente de tais instantes de fim de serviço;
2. Para cada fluxo f de pacotes de um usuário/escalonador, os instantes de fim de serviço são iterativamente calculados como:

$$F(p_f^j) = \max\{v[A(p_f^j)], F(p_f^{j-1})\} + \frac{l_f^j}{\phi_f}, j \geq 1$$

Onde p_f^j = pacote j do fluxo f , $A(p_f^j)$ = instante de chegada do pacote p_f^j , $F(p_f^0) = 0$ e ϕ_f é o peso do fluxo f .

3. $v[t]$ é o instante virtual de fim de serviço no instante “ t ” que corresponde ao instante virtual de fim de serviço do pacote em serviço naquele instante:

$v[t] = F(p_f^j), s_t^j < t \leq d_t^j$ onde s_t^j e d_t^j denotam respectivamente, os instantes em que o pacote p_f^j começa e termina serviço;

4. Quando os escalonadores se tornam ociosos, isto é, todas as filas estão vazias, o algoritmo é reinicializado. Fixa-se novamente em zero o instante virtual de fim de serviço.