

4

Simulador

Uma das etapas integrantes deste trabalho foi o desenvolvimento de uma ferramenta de simulação de Redes de Comunicações Móveis. O objetivo deste simulador foi verificar o desempenho dos algoritmos propostos nesta tese para Controle de Admissão de Chamadas de usuários de múltiplas classes. Optou-se pelo desenvolvimento de um simulador ao invés de utilizar outros já existentes no mercado tais como o NS₂, pela necessidade de dominar completamente o processo de simulação, uma vez que o objetivo é trabalhar apenas com o controle de admissão de usuários e não com a rede completa. Por outro lado, ao longo do desenvolvimento do simulador, observou-se que a eliminação completa dos problemas apresentados pelo simulador revelou-se um trabalho bem maior do que o esperado.

Para verificar a confiabilidade dos resultados obtidos pelo simulador foram testados vários casos que apresentam solução analítica. Os resultados analíticos foram comparados aos resultados simulados e ambos os resultados são coerentes entre si conforme se observa nos resultados apresentados neste capítulo.

O programa simulador foi desenvolvido em linguagem C++ com a preocupação de manter o código de programação em módulos. Tal procedimento permitirá que a ferramenta seja explorada em trabalhos de pesquisa futuros sem a necessidade do domínio de todo o programa. Além disto, o desenvolvimento da ferramenta foi feito de uma forma geral, incluindo a previsão para diversos módulos que não foram integralmente explorados. Nesta tese, foram explorados os aspectos do simulador relativos ao Controle de Admissão de Usuários de Múltiplas Classes.

4.1

Descrição dos Módulos do Simulador

A figura 4.1 apresenta o fluxograma do simulador, nela estão apresentados os módulos que são descritos a seguir e as suas conexões. O programa

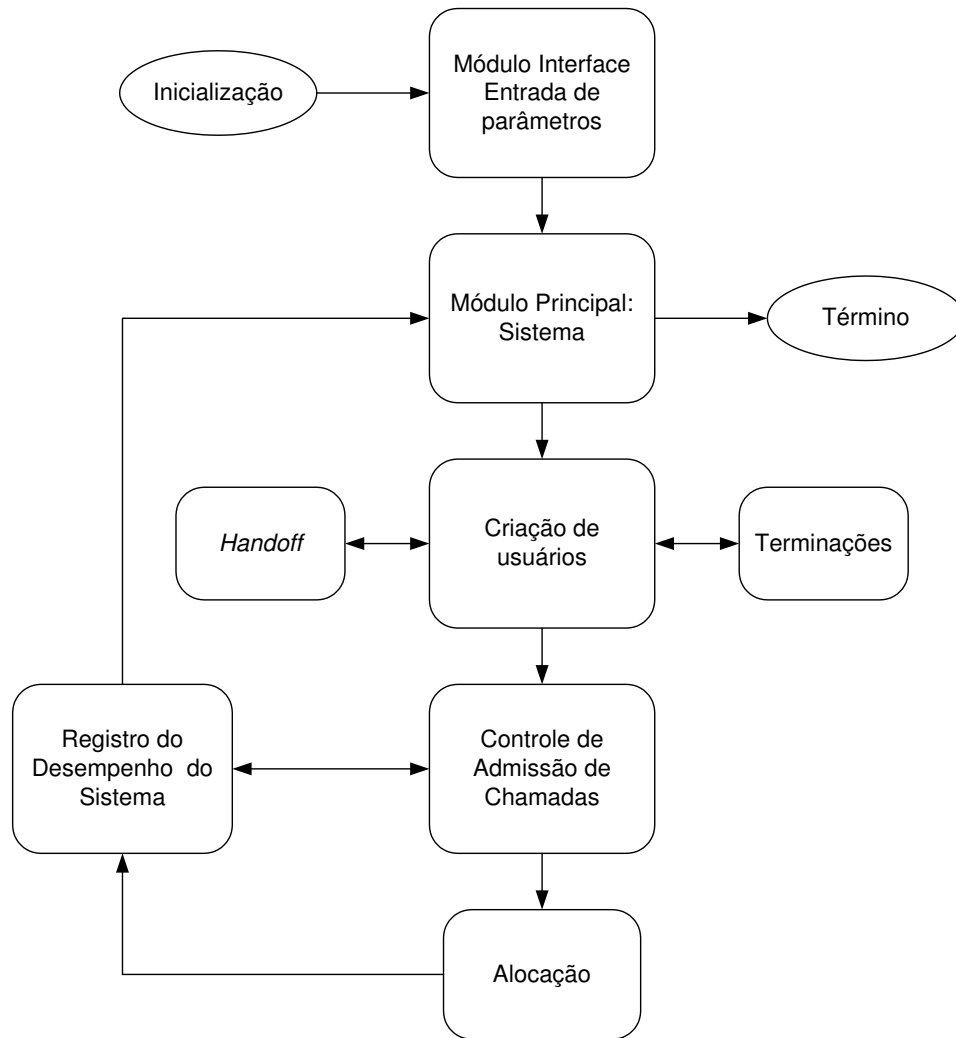


Figura 4.1: Fluxograma do Simulador.

começa com a inicialização e a entrada de parâmetros. O Módulo principal contém o *loop* com os comandos que se repetem ao longo de toda a simulação. Os usuários são criados e são identificados como usuários novos ou como usuários necessitam fazer *handoff*. Identificados os recursos que estão disponíveis e os usuários que necessitam de recursos, o programa passa ao Controle de Admissão de Chamadas que consulta os parâmetros de Desempenho do Sistema e define, segundo seus critérios, quais usuários serão encaminhados do módulo alocação. O resultado da alocação é registrado. São terminadas as chamadas correspondentes ao intervalo de tempo tratado na iteração e seus recursos são liberados. O módulo principal reinicia nova iteração até que todos os usuários tenham sido atendidos, ou não, e a simulação é encerrada.

4.1.1

Inicialização

O módulo inicia as matrizes e vetores do sistema. Algumas matrizes e vetores são fixos e não são alterados durante a execução, enquanto outros são destinados a armazenar informações constantemente atualizadas.

O simulador trabalha com uma célula em forma de hexágono, dividida em três setores iguais, conforme a figura 4.2.

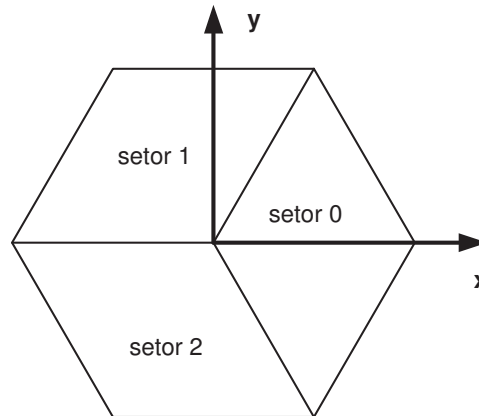


Figura 4.2: Célula hexagonal setorizada.

4.1.2

Interface

No Módulo Interface há a definição de todos os parâmetros de entrada da simulação. Dentre outros, são definidos: a intensidade de tráfego, a taxa de mobilidade, as classes de usuários e qual algoritmo CAC será simulado.

Além dos parâmetros de entrada, este módulo também é o responsável por gerar os arquivos numéricos de saída aonde são impressos os resultados da simulação.

Os resultados emitidos pelo programa são o Percentual de Bloqueio de chamadas novas, P_{BN} , o Percentual de Falha de chamadas em *handoff*, P_{FH} e a Utilização dos recursos do sistema, U para cada classe de usuários.

4.1.3

Principal

O módulo principal é um *loop* com a variável tempo. A cada nova iteração são criados os usuários novos e de *handoff* e é feita a verificação

de quais usuários terminarão sua chamada dentro do intervalo de tempo considerado pela iteração em execução no programa.

São verificados quais são os usuários envolvidos no intervalo de tempo correspondente à iteração e esses são ordenados. Para os usuários novos, a ordem se dá pelo o instante de início da chamada; para os usuários de *handoff*, pelo instante de entrada no sistema e, para os que terminam o serviço, pelo instante de término.

A alocação dos usuários, novos ou de *handoff*, é feita de acordo com o critério adotado pelo algoritmo de controle de admissão que está sendo testado na simulação.

Após a alocação de canais há a contabilização dos parâmetros de desempenho. As tarefas são executadas em módulos separados que são chamados através do Módulo Principal.

4.1.4

Criação de usuários

Conforme o modelo amplamente aceito e adotado na literatura, utiliza-se a distribuição de Poisson para simular a chegada de novas chamadas. No caso da simulação ser realizada com uma única célula, os usuários em *handoff* também são criados da mesma forma que os novos. O tempo de retenção do recurso varia de acordo com o modelo de tráfego que está sendo simulado para o tempo de permanência do usuário na célula, de acordo com o desenvolvimento feito no apêndice A.

O simulador desenvolvido neste trabalho oferece a possibilidade de alocação de mais de um recurso, Unidade de Banda, *UB*, a cada usuário.

As classes possuem diferentes necessidades de banda. Um usuário que tenha alocado para si várias Unidades de Banda, corresponde a uma taxa maior de transmissão de dados. O sistema proposto tem a capacidade de até dez classes diferentes, cada uma com até dez *UB*.

Cada classe de usuário é definida no início da simulação juntamente com sua necessidade de banda; seu tempo médio de duração da chamada e o percentual de usuários que pertencerão à classe.

Na geração de cada usuário são feitos sorteios para definição da classe a qual este pertence; se é usuário novo ou em *handoff* e em qual ponto do mapa este iniciará sua chamada, conforme a figura 4.3. A intensidade de solicitação de recursos é previamente definida na entrada de dados da execução.

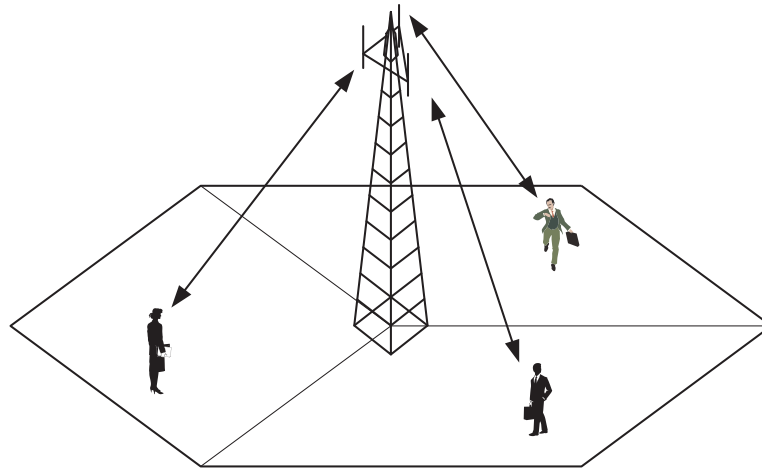


Figura 4.3: Usuários na Célula setorizada.

4.1.5 Handoff

O módulo de *Handoff* monitora quais são os usuários que necessitam de *handoff* dentro do intervalo de iteração. Sua identificação é armazenada para que possa ser consultada pelo Controle de Admissão.

Em relação à prioridade das chamadas em *handoff*, o sistema apresenta a reserva de canais de guarda. O número inicial de canais de guarda por setor é definido nos parâmetros de entrada do programa. Este número pode ser fixo ou dinâmico de acordo como algoritmo de canais de guarda escolhido.

4.1.6 Terminações

O módulo é completamente independente e trata apenas da terminação do usuário. A tomada de decisão da terminação é realizada pelos outros módulos. Diversas razões podem levar um usuário a ser terminado; podem ser terminados os usuários cujo tempo de chamada termina, usuários que sofreram bloqueio por excesso de tentativas falhas de efetuar *handoff* ou, em caso de sistemas que consideram interferência, aqueles cuja razão sinal/ruído não é suficiente para manter a chamada.

Ao executar a terminação do usuário, todos os recursos que estavam sendo utilizados pelo usuário são liberados.

4.1.7

Controle de Admissão de Chamadas, CAC

Os usuários novos, gerados na iteração vigente, e os usuários que necessitam *handoff* são encaminhados ao Controle de Admissão para que esse módulo decida se o usuário será aceito ou não.

Um dos principais objetivos do simulador neste trabalho de tese é testar diversos algoritmos de Controle de Admissão de Chamadas. Para tal, no início da simulação há a definição sobre qual dos algoritmos de Controle de Admissão será aplicado.

A cada intervalo de iteração todos usuários são ordenados de acordo com o instante de tempo de início de chamada, para os novos, e de instante de solicitação de *handoff*. O instante de solicitação do recursos de cada usuário é definido no início da simulação de acordo com a distribuição de Poisson.

O Controle de Admissão varre a lista de usuários a serem tratados, identifica se é novo ou *handoff*, a classe e a necessidade de recursos de cada um. Cada usuário é tratado individualmente, o CAC consulta os parâmetros de desempenho e decide se o usuário deve ser encaminhado à alocação ou não.

4.1.8

Alocação

O Módulo Alocação é chamado pelo Módulo de Controle de Admissão de Chamadas para alocar recursos disponíveis ao usuário indicado. O simulador opera com o sistema de Alocação Fixa de Canais.

4.2

Registro do Desempenho do Sistema

O tráfego e o desempenho do sistema são permanentemente monitorados através do registro de todas as tentativas de chamadas e de seus respectivos sucessos ou falhas de alocação. Baseado nestes registros, são obtidos os valores instantâneos de percentuais de bloqueio de novas chamadas, B_{BN} , de falha de *handoff*, B_{FH} e de perfil de tráfego, θ_{medido} . As medidas de desempenho são monitoradas e comparadas aos requisitos de QoS para a tomada de decisão do Controle de Admissão de Chamadas.

4.2.1

Identificação e Registro da Chamada

A cada tentativa de alocação no setor, a chamada é identificada, se é de *handoff* ou nova e, para o caso de sistemas de múltiplas classes, a que classe pertence a chamada.

Há uma janela de registro para cada tipo de chamada. Para usuários de uma única classe são necessárias duas janelas de registro: chamadas novas e chamadas em *handoff*. Para o caso de M classes são necessárias $2M$ janelas de registro.

Identificado o tipo da chamada, ocorre o registro da tentativa de alocação e do seu sucesso ou falha.

4.2.2

Janela de Medidas

A janela de medidas é o nome dado ao vetor que comporta os registros de sucessos e falhas de alocação no sistema.

Deslizamento

Para efetuar o registro, após a identificação do usuário a respectiva janela é deslocada da posição (0) até que a última medida, a mais velha, seja perdida. A figura 4.4 apresenta a janela de medidas e o deslizamento dos registros cada vez que se pretende fazer um novo registro na posição (0).

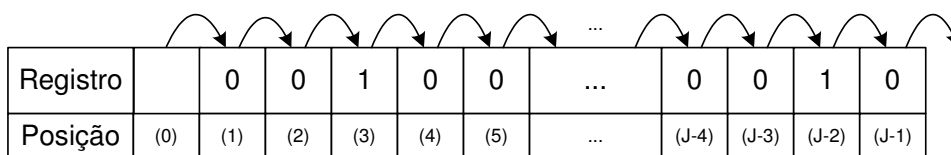


Figura 4.4: Janela de Medidas.

A solicitação de chamada é encaminhada para a tentativa de alocação. Então, é verificado se houve sucesso, 0, ou falha, 1, de alocação. O resultado da tentativa de alocação, 0 ou 1, é registrado na posição (0) da janela.

O tamanho total da janela é J . São mantidas as medidas das J últimas tentativas de alocação de cada tipo de chamada, na posição (0) até a posição correspondente ao registro mais antigo ($J - 1$).

Baseado nas informações de sucesso e falha de alocação registradas nas janelas, é possível calcular a medida instantânea dos parâmetros de desempenho do sistema. Da janela relativa aos usuários novos obtém-se a medida instantânea da probabilidade de bloqueio de chamadas novas P_{BN} , chamada de B_{BN} . Da mesma forma, da janela relativa aos usuários em *handoff* obtém-se a medida instantânea da probabilidade de falha de chamadas em *handoff*, P_{FH} , chamada de B_{FH} . Estas medidas são usadas pelos algoritmos de CAC para a tomada de decisão sobre a alocação dos usuários.

O método de obtenção das medidas é foi testado verificando se o tamanho da janela é adequado comparando do valor medido e o valor já previamente conhecido gerado na simulação. Além disto, atribui-se um peso maior aos registros mais recentes, através de uma média ponderada com coeficientes que possuem um decaimento exponencial.

O tamanho da janela é um parâmetro importante na obtenção da medida pois, a utilização de uma janela muito grande acarretará em demora excessiva para se tomar uma atitude na admissão dos usuários. Por outro lado, uma janela demasiado pequena tende a provocar atitudes precipitadas, por não refletir uma medida estatística confiável do problema.

Para definir o tamanho da janela vamos observar um exemplo.

Exemplo

Supondo que a duração da chamada é em média $\frac{1}{\mu} = 120s$. A intensidade de solicitação de recursos normalizada, ρ , definida na equação (3-11), será considerada de 0.5 a 0,9.

$$\rho = \frac{\lambda}{\mu N_c} \Rightarrow \lambda = \rho N_c \mu$$

O número total de canais em cada setor é $N_c = 30$ canais. Portanto, a taxa total de chegada de usuários ao setor é dada por λ , que para um ρ igual a 0.5, corresponde a 450 usuários/hora.

$$\lambda = \rho N_c \mu = 0.5 \times 30 \frac{1}{120} = 0,125u/s = 7,5u/min = 450u/h$$

Para ρ igual a 0,9, corresponde a 810 usuários/hora.

$$\lambda = \rho N_c \mu = 0,9 \times 30 \frac{1}{120} = 0,225u/s = 13,5u/min = 810u/h$$

Escolhendo uma janela de 100 registros, supondo o caso em que a mobilidade θ é igual a um, teremos igual quantidade de usuários novos e de handoff no sistema. Então, no caso em que ρ é igual a 0.5, para preencher a janela de usuários novos serão necessários, em média, menos de 30 minutos.

Tamanho da Janela em Função do Requisito de QoS

Do exemplo anterior, observamos que quanto maior a janela maior a precisão da medida obtida, porém mais demorado se torna preencher a janela. Além disto, se a janela é muito grande, pode haver medidas muito antigas e se o sistema varia ao longo do tempo, as condições podem ter mudado desde os registros mais antigos.

Neste trabalho foi decidido que a janela será uma função de seu respectivo parâmetro de QoS denominado Q . O tamanho J , escolhido é dado em 4-1 de tal forma que sempre haja, em média, cerca de cinco falhas de alocação registradas na janela.

$$J = 5 \frac{1}{Q} \quad (4-1)$$

Na tabela 4.1 são apresentados os valores de Q e o respectivo tamanho de janela adotado.

Tabela 4.1: Tamanhos de janelas adotados para cada valor de Q .

Q	$1/Q$	J
0.05	20	100
0.04	25	125
0.03	33	165
0.01	100	500

4.2.3

Cálculo da Medida

Os registros feitos na janela são divididos em cinco grupos iguais, de tamanho $\frac{J}{5}$. Para cada grupo de registros é feita a soma dos valores de todos registros encontrados nas $\frac{J}{5}$ posições. Cada soma é chamada de S_i .

A medida B é tomada pela média ponderada dos registros, com pesos X_i , ($i = 0, \dots, 4$), conforme a equação 4-2.

$$B = \frac{(X_0 \times S_0) + (X_1 \times S_1) + (X_2 \times S_2) + (X_3 \times S_3) + (X_4 \times S_4)}{20 \times (X_0 + X_1 + X_2 + X_3 + X_4)} \quad (4-2)$$

São adotados pesos maiores para as medidas mais recentes, considerando um decaimento exponencial para o esquecimento das medidas mais antigas.

$$X_0 > X_1 > X_2 > X_3 > X_4 \quad (4-3)$$

4.3

Conclusão e Validação do Simulador

A verificação da confiança dos resultados obtidos no Simulador desenvolvido neste trabalho é feita pela simulação de casos que também possuem solução analítica. No capítulo anterior foi apresentada a solução por Cadeias de Markov para o caso de sistemas que utilizam um número fixo de canais de guarda, CG .

Os casos a serem comparados são: número fixo de canais de guarda: $1CG$, $0.5CG$ e $0CG$, e razão entre chegada de usuários de *Handoff* e Novos dada por $\theta = \frac{\lambda_H}{\lambda_N} = 5$. Esses casos foram simulados em linguagem C++ e seus resultados são comparados aos resultados analíticos, gerados no programa em *MATLAB*® que calcula os valores de Percentual de Bloqueio de Chamadas Novas, P_{BN} , segundo a equação (3-36); Percentual de Falha de Chamadas em *Handoff*, P_{FH} , equação (3-38). As comparações dos resultados de P_{BN} e de P_{FH} estão nas figuras 4.5 e 4.6 onde se observa que os resultados do simulador conferem com os resultados analíticos. O intervalo de confiança considerado é de 95%.

Portanto, concluímos que os resultados obtidos pelo simulador são confiáveis, pois conferem perfeitamente com os resultados analíticos para os casos em que há a solução analítica exata. Como os algoritmos dinâmicos operam com valores de canais de guarda dentro das faixas que foram testadas pelo simulados, espera-se que este apresente um comportamento semelhante e confiável.

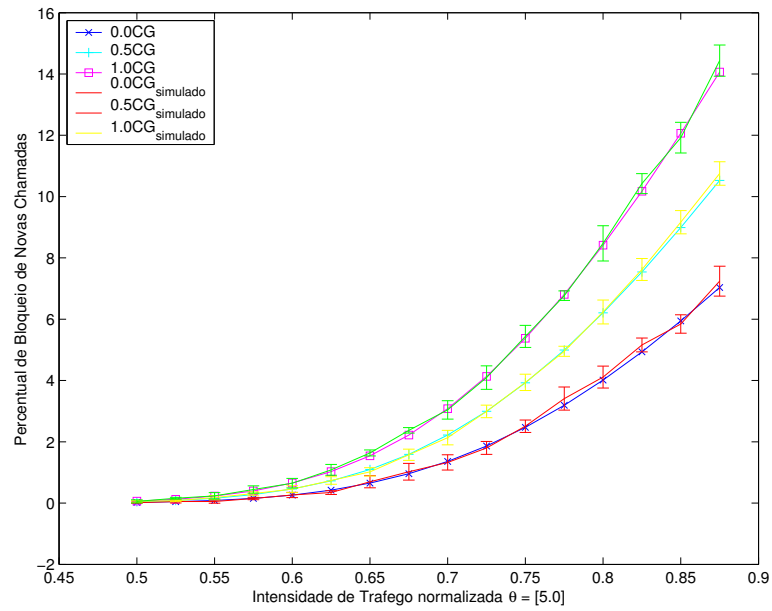


Figura 4.5: Comparação de resultados simulados e analíticos de Percentual de Bloqueio de Chamadas Novas versus Intensidade de Tráfego normalizada para $\theta=5$.

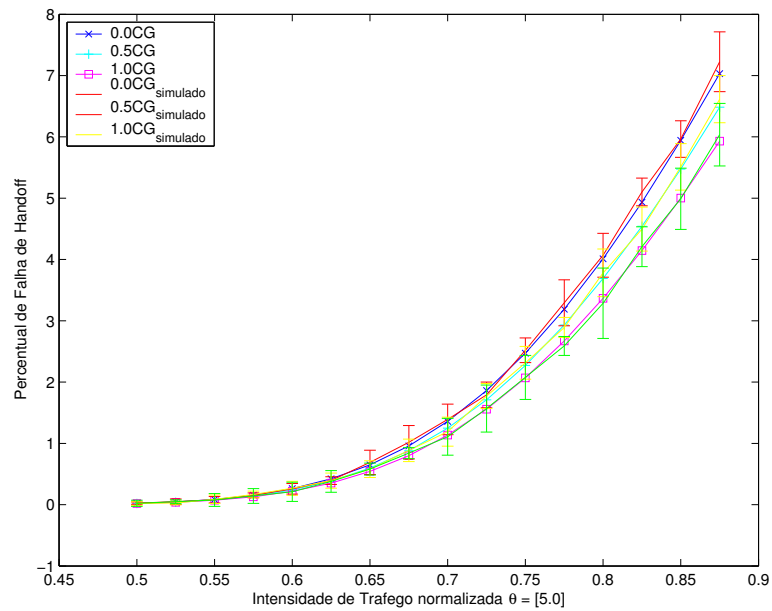


Figura 4.6: Comparação de resultados simulados e analíticos de Percentual de Falha de Chamadas em *Handoff* versus Intensidade de Tráfego normalizada para $\theta=5$.