

PONTIFÍCIA UNIVERSIDADE CATÓLICA DE MINAS GERAIS

Programa de Pós - Graduação em Informática

Cristiano Lopes Martins Borges

**CONTROLE DE ADMISSÃO DE CHAMADAS PARA REDES UMTS UTILIZANDO
LÓGICA NEBULOSA EM CASCATA**

Belo Horizonte

2012

Cristiano Lopes Martins Borges

**CONTROLE DE ADMISSÃO DE CHAMADAS PARA REDES UMTS UTILIZANDO
LÓGICA NEBULOSA EM CASCATA**

Dissertação apresentada ao Programa de Pós-Graduação em Informática da Pontifícia Universidade Católica de Minas Gerais, como requisito parcial para obtenção do título de Mestre em Informática.

Orientadora: Fátima de Lima Procópio Duarte
Figueiredo

Belo Horizonte

2012

FICHA CATALOGRÁFICA

Elaborada pela Biblioteca da Pontifícia Universidade Católica de Minas Gerais

B732c Borges, Cristiano Lopes Martins
Controle de admissão de chamadas para redes UMTS utilizando lógica nebulosa em cascata / Cristiano Lopes Martins Borges. Belo Horizonte, 2012. 78f.: il.

Orientadora: Fátima de Lima Procópio Duarte Figueiredo
Dissertação (Mestrado) – Pontifícia Universidade Católica de Minas Gerais. Programa de Pós-Graduação em Informática.

1. Sistemas de comunicação sem fios. 2. Sistemas de telecomunicação. 3. Sistemas de comunicação móvel. 4. Sistemas difusos. 5. Qualidade em serviços. I. Figueiredo, Fátima de Lima Procópio Duarte. II. Pontifícia Universidade Católica de Minas Gerais. Programa de Pós-Graduação em Informática. III. Título.

Cristiano Lopes Martins Borges

**CONTROLE DE ADMISSÃO DE CHAMADAS PARA REDES UMTS UTILIZANDO
LÓGICA NEBULOSA EM CASCATA**

Dissertação apresentada ao Programa de Pós-Graduação em Informática da Pontifícia Universidade Católica de Minas Gerais, como requisito parcial para obtenção do título de Mestre em Informática.

Prof^a. Fátima de Lima Procópio Duarte Figueiredo – Orientadora (PUC Minas)
Doutora em Ciência da Computação - UFMG

Prof. José Ferreira de Rezende (UFRJ)
Doutor em Ciência da Informação – UPMC, França

Prof. Luis Enrique Zárate Galvez – Coorientador (PUC Minas)
Doutor em Engenharia Metalúrgica e de Minas - UFMG

Belo Horizonte, 27 de junho de 2012.

AGRADECIMENTOS

Agradeço a Deus pelas bênçãos e por colocar tantos anjos em meu caminho.

À minha mãe Marlene por seu apoio incondicional, pelas orações e pela certeza que sempre estaria ali.

À minha esposa Ivie, por sua lealdade e amor, sempre me dando forças para buscar algo mais. Obrigado por acreditar em mim.

Aos meus familiares e de uma maneira especial, minhas irmãs Andréa e Daniela, pelo carinho, incentivo e momentos de descontração.

Aos meus grandes amigos Patric, Gabriel, Cláudio Luiz, Elias, Ledimir, Michele Farnese, Joailson, Titõe, Fábio Henrique, Etinho, Igor Silveirinha, Rogério, Mateus Lopes, Mateus Lobato, Ivan Silveirinha, Dadaça, Simone Molina, Uillian, Valéria, David, Oberdan e ao Deputado Eolo, que contribuíram diretamente para a concretização deste trabalho.

Agradeço à minha orientadora Fátima, pela sabedoria em potencializar minhas qualidades e dar o melhor tratamento para meus defeitos. Obrigado por contribuir para minha evolução como ser humano, aprendi muito com você.

Ao Luiz Zárte por sempre colaborar e fazer excelentes considerações no desenvolver deste trabalho.

De uma maneira muito especial, agradeço à Anna Tostes, por ter compartilhado inúmeros conhecimentos e demonstrado sua grandeza com simplicidade.

Aos demais grandes amigos que sempre estiveram ao meu lado, torcendo por mim, que indiretamente também contribuíram turma do futebol de areia, Família Crediprata, amigos do Sicoob Central Cecremge, entre outros.

Quereria agradecer àquele que me ensinou a terminar todo projeto iniciado e nunca, em nenhuma hipótese desistir. Graças aos seus ensinamentos tive forças para continuar, após sua partida. Ficou mais difícil, mas valeu a pena continuar, pois acredito que esta conquista te deixaria orgulhoso. Sinto sua falta, meu pai.

RESUMO

É notável o franco crescimento de dispositivos móveis capazes de oferecer serviços como navegação na web, videoconferência, envio de e-mails, além de chamadas de voz. No Brasil, todo esse conjunto de serviços é suportado pelas redes de celulares UMTS, também conhecidas como 3G, de terceira geração. Um dos desafios das redes 3G é a garantia de Qualidade de Serviço (QoS), medida pela disponibilidade da rede e pela satisfação do usuário. O Controle de Admissão de Chamadas (CAC) é um ponto crítico na garantia de qualidade de serviço (QoS), pois decide sobre a aceitação ou bloqueio de uma nova requisição de chamada. Um exemplo é o CAC-RDF, que implementa, através de Lógica Nebulosa, bloqueio dinâmico de chamadas de acordo com o nível de congestionamento da rede. Mas o CAC-RDF possui um alto índice de bloqueio em processos de *handover*. Este trabalho apresenta uma solução para reduzir os bloqueios de processos de *handover* através da implementação de uma estrutura de Lógica Nebulosa em cascata, sobre o CAC-RDF, reavaliando bloqueios, caso sejam *handovers*. Com esta arquitetura em cascata, é possível um ganho em termos de processamento computacional, pois, a saída da primeira estrutura de Lógica Nebulosa é reaproveitada como entrada da segunda estrutura de Lógica Nebulosa. Os resultados mostram que a proposta apresentada reduz, de fato, o bloqueio de *handovers*, sem afetar o desempenho e a qualidade da rede como um todo.

Palavras-chave: CAC. *Fuzzy*. UMTS. QoS.

ABSTRACT

It is remarkable how rapidly growing of mobile devices are able to offer services like web browsing, video conferencing, sending e-mails, and voice calls. In Brazil, this whole chain of services is supported mainly by UMTS, also known as 3G networks, third-generation cellular. One of the challenges of 3G networks is to guarantee Quality of Service (QoS), measured by the network availability and user satisfaction. The Call Admission Control (CAC) is critical in ensuring quality of service (QoS), it decides on the acceptance or blocking a new call request. An example is the CC-RDF which implements by means of Fuzzy Logic block, dynamic call in accordance with the level of network congestion. But the CAC-RDF has a high rate of blockage in the process of handover, then, this paper presents a solution to reduce blockages handover processes through the implementation of a Fuzzy Logic structure cascade on the CAC-RDF, reassessing blocks, if handovers. With this architecture you can gain in terms of computing power, there is output of the first structure of Fuzzy Logic is reused as input of the second structure of Fuzzy Logic.

Keywords: CAC. *Fuzzy*. UMTS. QoS.

LISTA DE FIGURAS

FIGURA 1 - Escopo da solução	25
FIGURA 2 - Arquitetura das redes UMTS	27
FIGURA 3 - Requisitos de QoS na rede UMTS.....	31
FIGURA 4 - Mecanismo de Lógica Nebulosa.....	34
FIGURA 5 - Controle de Admissão de Chamadas – CAC.....	35
FIGURA 6 - Processo de <i>Handover</i>	36
FIGURA 7 - Tipos de CACs.....	36
FIGURA 8 - Arquitetura CAC-RD	38
FIGURA 9 - Arquitetura CAC-RDF	38
FIGURA 10 - Bloqueio por Limites Estáticos do CAC-RD.....	39
FIGURA 11 - Arquitetura CAC-RDFF	42
FIGURA 12 - Visão simplificada do CAC-RDFF	43
FIGURA 13 - Diagrama de processos do CAC-RDFF.....	44
FIGURA 14 - Sete passos	46
FIGURA 15 - Função de pertinência do Estado da Rede.....	48
FIGURA 16 - Função de pertinência da Largura de Banda Requisitada.....	49
FIGURA 17 - Função de pertinência da Saída	50
FIGURA 18 - Regras de Inferência	51
FIGURA 19 - Representação da integração do Mecanismo LN com RRM	53
FIGURA 20 - Simulação dos resultados da LN	54
FIGURA 21 - Ilustração da simulação dos resultados da LN	55

LISTA DE GRÁFICOS

GRÁFICO 1 - Métodos de Defuzzificação	52
GRÁFICO 2 - Análise de bloqueios SHO entre CAC-RDF x CAC-RDFF	62
GRÁFICO 3 - Análise serviço <i>Background</i>	62
GRÁFICO 4 - Análise serviço <i>Interactive</i>	63
GRÁFICO 5 - Análise serviço <i>Conversational</i>	64
GRÁFICO 6 - Análise serviço <i>Streaming</i>	64
GRÁFICO 7 - Análise de aceites SHO entre CAC-RDF x CAC-RDFF – A	65
GRÁFICO 8 - Análise de aceites SHO entre CAC-RDF x CAC-RDFF – B	66
GRÁFICO 9 - Aceitação SHO <i>Conversational</i>	67
GRÁFICO 10 - Aceitação SHO <i>Streaming</i>	67
GRÁFICO 11 - Aceitação SHO <i>Interactive</i>	68
GRÁFICO 12 - Aceitação SHO <i>Background</i>	68
GRÁFICO 13 - Total de chamadas SHO (Bloqueadas e Aceitas).....	69
GRÁFICO 14 - QoS – Atraso <i>Conversational</i>	70
GRÁFICO 15 - QoS – Jitter <i>Conversational</i>	70
GRÁFICO 16 - QoS – Vazão <i>Conversational</i>	71
GRÁFICO 17 - QoS – Atraso <i>Streaming</i>	71
GRÁFICO 18 - QoS – Vazão <i>Streaming</i>	72
GRÁFICO 19 - QoS – Vazão <i>Interactive</i>	72
GRÁFICO 20 - QoS – Vazão <i>Background</i>	73

LISTA DE TABELAS

TABELA 1 - Cenário das simulações	56
TABELA 2 - Mapa dos resultados das simulações: <i>Conversational</i>	58
TABELA 3 - Mapa dos resultados das simulações: <i>Streaming</i>	59
TABELA 4 - Mapa dos resultados das simulações: <i>Interactive</i>	60
TABELA 5 - Mapa dos resultados das simulações: <i>Background</i>	61

LISTA DE SIGLAS

3G - Terceira Geração

3GPP - *Third Generation Partnership Project*

BS - Estação Rádio Base

CAC-J - Controle Admissão de Chamadas de Josephine Antoniou (2004)

CAC-RDF - CAC-RD - *Fuzzy*

CAC-RDFF - CAD-RD - *Fuzzy Cascata*

CAC-RDN - CAC-RD - Neural

CI - Inteligência Computacional

CN - *Core Network*

EU - Equipamento do Usuário

E-UMTS - *Enhanced Universal Mobile Telecommunication System*

GPRS - *General Packet Radio Service*

IP - *Internet Protocol*

LF - *Lógica Fuzzy*

LN - *Lógica Nebulosa*

NRT - *Not Real Time*

NS-2 - *Network Simulator versão 2*

NSHO - *Não Handover suave*

QC - Queda de Chamada

QoS - Qualidade de Serviço

RT - Tempo Real

SHO - *Handover suave*

SMS - *Short Message Service*

SQL - Linguagem de Consulta Estruturada

TDMA - *Time Division Multiplexing Access*

TQC - Taxa de Queda de Chamada

UMTS - *Universal Mobile Telecommunication System*

VoIP - Voz sobre IP

SUMÁRIO

1 INTRODUÇÃO	23
1.1 Definição do Problema.....	25
1.2 Objetivos	26
1.3 Justificativa.....	26
2 REFERÊNCIAL TEÓRICO.....	27
2.1 Redes UMTS	27
2.2 Taxa de queda de chamada.....	28
2.3 QoS.....	29
2.4 Lógica Nebulosa.....	32
2.5 Controle de Admissão de Chamada (CAC)	35
3 METODOLOGIA DE PESQUISA.....	41
4 REPRESENTAÇÃO DE LÓGICA NEBULOSA EM CASCATA	42
4.1 Escopo da modelagem	46
4.2 Escolha das variáveis linguísticas	47
4.3 Definição das funções de pertinência	47
4.4 Definição das regras de inferência	50
4.5 Definição do método de defuzzificação	51
4.6 Implementação do mecanismo de LN	52
4.7 Validação dos resultados	53
5 SIMULAÇÕES E ANÁLISES DO CAC-RDFF	56
5.1 Cenário das simulações	56
5.2 Mapa das simulações.....	58
5.3 Resultados de bloqueios de SHO	61
5.4 Resultados de aceites de SHO.....	65
5.5 Resultados de QoS	69
6 CONCLUSÕES E TRABALHOS FUTUROS	74
REFERÊNCIAS.....	76

1 INTRODUÇÃO

A crescente demanda por serviços de comunicação gerada pela massificação de dispositivos móveis implica em um grande desafio na garantia de Qualidade de Serviço (QoS) em uma rede celular.

Existe também o desafio para as operadoras em sempre ampliar, ou pelo menos manter, a base de assinantes. Uma das formas para diferenciar o serviço oferecido e manter fiel o usuário é prestar serviço de qualidade, de forma a atender plenamente suas necessidades. Oferecer disponibilidade de recursos e atingir a satisfação dos usuários deve ser uma preocupação constante das operadoras de telefonia celular.

Conforme o número de usuários e de serviços aumenta, maior é a dificuldade de se decidir sobre a aceitação ou o bloqueio das solicitações de serviços. O problema é como decidir quais aplicações aceitar ou bloquear, de modo que se assegure a garantia de QoS e o melhor aproveitamento dos recursos da rede. Por isso, devem existir mecanismos que permitam esse controle dos serviços requisitados na rede.

Controle de Admissão de Chamadas (CAC) é um mecanismo de QoS, que decide pela aceitação ou pelo bloqueio de uma solicitação de serviço na rede, considerando a disponibilidade de recursos. A aceitação de serviços sem critérios, pode ocasionar um mau aproveitamento de recursos, o que é inaceitável no contexto de garantia de QoS.

Mobilidade é o principal aspecto a ser considerado em uma rede celular. É desejável que o usuário se locomova tendo garantia da qualidade dos serviços utilizados. O procedimento que permite a locomoção entre estações rádio base da rede UMTS, sem perda de conexão do serviço utilizado pelo usuário, é chamado *Handover* Suave (SHO). No momento do processo de SHO, é feita uma solicitação de serviço para a estação rádio base destino. O CAC é o módulo responsável por aceitar ou bloquear essa solicitação, de acordo com a disponibilidade de recursos da estação rádio base em questão.

O CAC pode utilizar diversas técnicas, tais como: limite de aceitação de serviços, reserva de recursos e priorização de aplicações. Os limites de aceitação são baseados em parâmetros da rede e podem ser estáticos ou dinâmicos. Quando

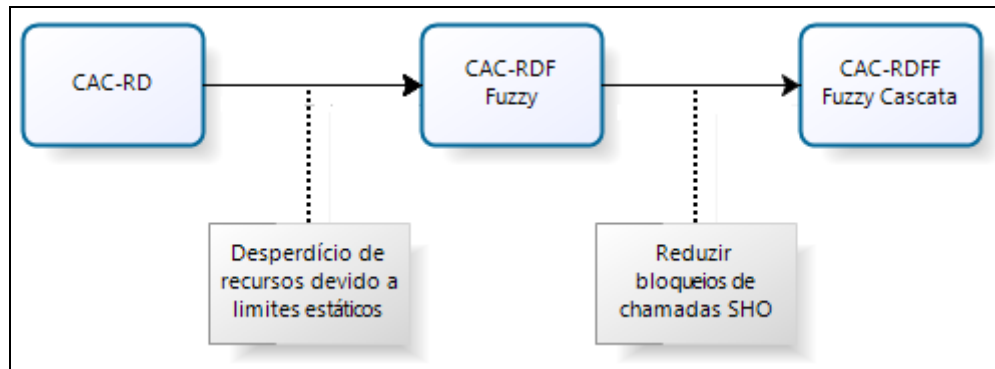
o limite é atingido, o CAC bloqueia novas solicitações. A reserva de recursos também pode ser estática ou dinâmica. Na reserva estática, o mais comum é os CACs guardarem alguns canais para aplicações prioritárias, por exemplo, partindo da classe mais prioritária para a menos prioritária.

Existem duas categorias de CACs: convencional e não convencional. Os CACs convencionais utilizam técnicas determinísticas ou estocásticas, tais como a priorização de serviços, a reserva de recursos, o empréstimo de canais e a utilização de limites estáticos ou dinâmicos (GHADERI; BOUTABA, 2006). Um exemplo de CAC convencional é o Controle de Admissão de Chamadas, baseado em Reserva de recursos e Diagnóstico da rede (CAC-RD) apresentado em Storck (2007), que é baseado em reserva de recursos, diagnóstico da rede e na imposição de limites estáticos de aceitação de aplicações menos prioritárias.

Em contrapartida, os CACs não convencionais utilizam de Inteligência Computacional (Rede Neural Artificial, Lógica Nebulosa, etc) associado às técnicas tradicionais (ENGELBRECHT, 2002). Esses mecanismos são capazes de inserir comportamento considerado inteligente em ambientes dinâmicos, por isso, são capazes de aprender, adaptar, generalizar, abstrair e associar novas situações.

Um exemplo de CAC não convencional é o CAC-RD *Fuzzy* (CAC-RDF), proposto em Ribeiro (2010), que é uma melhoria do CAC-RD com a utilização de Lógica Nebulosa (LN). Seu objetivo é melhorar o aproveitamento dos recursos da rede através do bloqueio dinâmico de novas chamadas. O CAC-RDF foi desenvolvido para melhorar o CAC-RD no quesito aproveitamento de recursos, já que o CAC-RD, em certas condições de tráfego, desperdiça recursos devido ao emprego de limites estáticos de utilização da rede. Como resultado, o CAC-RDF viabilizou, com sucesso, limites dinâmicos para bloqueios de novas chamadas, sendo possível utilizar LN para melhorar a garantia de QoS para aplicações prioritárias. Contudo, o CAC-RDF possui um alto índice de bloqueio de processos SHO.

Figura 1 - Escopo da solução



Fonte: Elaborado pelo autor

Com o objetivo de reduzir os bloqueios de processos de *handover* no CAC-RDF, este trabalho propõe a inclusão de um novo controle para um melhor tratamento das chamadas *handover* (SHO), no CAC-RDF. Assim, foi incluída uma nova camada de lógica nebulosa na arquitetura do CAC-RDF, a fim de cobrir essa questão. Uma segunda estrutura baseada em LN foi desenvolvida, em cascata, com o objetivo de maximizar o número de aceitação de novos processos SHO.

A diminuição dos bloqueios de SHO através de LN em cascata motivou o desenvolvimento do trabalho. Essa solução permite o tratamento de incertezas (imprecisões) recorrentes no cenário de admissão de solicitações de serviços em ambientes de rede celular. Além disso, a LN possui uma capacidade linguística que permite modelar a complexidade do tráfego por meio de valores linguísticos (baixo, médio, alto). Para avaliar estatisticamente o trabalho, foram realizadas simulações no módulo E-UMTS (ANTONIOU, 2004), desenvolvido no simulador Network Simulator – Versão 2 (NS-2).

1.1 Definição do Problema

O CAC-RDF apresentou vantagens em relação ao CAC-RD. Por exemplo, quando foi garantida QoS para todas as classes, o CAC-RDF aceitou todas as chamadas (com priorização de serviço), enquanto o CAC-RD bloqueou todas as aplicações *Background* após 40% de utilização, mesmo que a rede estivesse ociosa.

Sua principal vantagem é o melhor aproveitamento dos recursos da rede. Os resultados mostraram que o CAC-RDF tem uma melhor aceitação das aplicações prioritárias, respeitando os parâmetros de QoS. Mas, em Ribeiro (2010), não se

avaliou os impactos em serviços SHO, que deveriam ser serviços priorizados. Esse foi um dos trabalhos futuros indicados.

Serviços SHO são prioritários, pois como o *handover* deve ser suave, não pode ocorrer perda de conexão e, além disso, essa transição entre estações rádio base deve ser feita com garantia de QoS. O problema que motivou este trabalho foi justamente a necessidade de se reduzir os bloqueios para essas aplicações SHO.

1.2 Objetivos

Este trabalho tem como principal objetivo propor melhoria para o processo de SHO de redes UMTS para diminuir o número de bloqueios desta classe de serviço.

Os objetivos específicos deste trabalho são:

- a) Propor mecanismo de LN sobre o CAC-RDF (cascata) para melhorar o processo de SHO e, conseqüentemente, diminuir o número de bloqueios;
- b) Priorizar a satisfação dos usuários em relação à disponibilidade da rede;
- c) Realizar simulações e testes de modo a comprovar a eficiência da solução proposta, baseado em métodos comprovados estatisticamente.

1.3 Justificativa

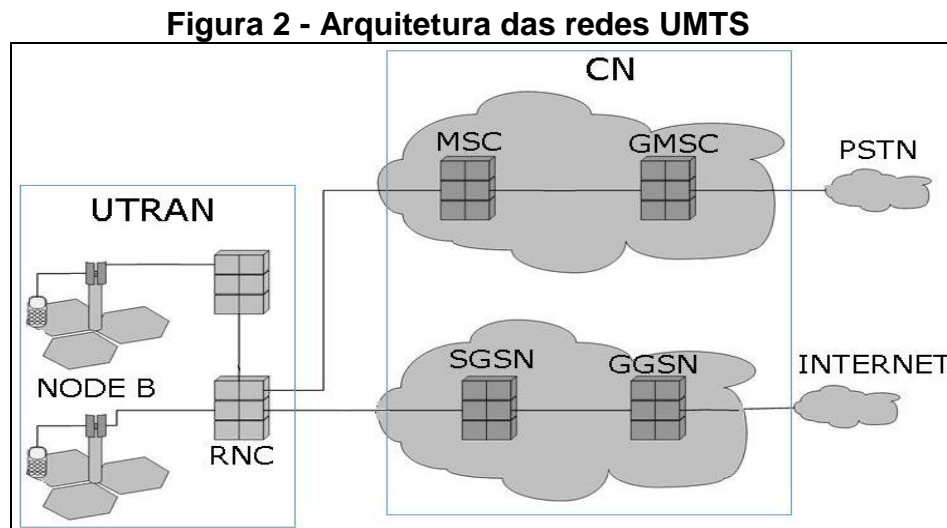
A justificativa deste trabalho baseia-se nos excelentes resultados do CAC-RDF apresentados no trabalho de Ribeiro (2010), quando comparados aos parâmetros de QoS definidos pelo 3GPP (GENERATION..., 2011b), grupo que desenvolve as especificações da tecnologia UMTS. Mas Ribeiro (2010) não avaliou o impacto em serviços SHO, que deveriam ser serviços priorizados, então, resolver este problema e dar um melhor tratamento para serviços SHO torna-se relevante para a evolução do CAC-RDF.

A abordagem do mecanismo de LN em Cascata se justifica quando se leva em consideração o ganho computacional quando a saída do primeiro mecanismo de LN é aproveitada como entrada no segundo mecanismo de LN, caracterizando assim o mecanismo de LN em Cascata.

2 REFERÊNCIAL TEÓRICO

2.1 Redes UMTS

O UMTS é o padrão 3G estabelecido como evolução para o *Global System for Mobile Communications* (GSM) e foi desenvolvido pelo 3GPP. Conforme Figura 2 ilustra os principais componentes físicos de uma rede UMTS. A rede UMTS é composta de duas partes, o *Core Network* (CN) e a *UMTS Terrestrial Radio Access Network* (UTRAN).



Fonte: Adaptada de MIKUSZEWSKI, 2009

A UTRAN corresponde à interface aérea da rede, enquanto o CN fica responsável por prover o acesso à internet ou à rede pública de telefone. A conexão entre o UTRAN e o CN é efetuada através da Interface Iu.

Segundo Manner (2003), dentre os componentes da CN podemos destacar o papel da comutação por circuito feito pelo *Mobile Switching Center Server* (MSC Server) que opera com o GPRS MSC Server (GMSC) que por sua vez é responsável pela interface com a RTPC (Rede de Telefonia Pública Comutada), representada (Figura 2) por *Public Switched Telephone Network* (PSTN).

Ainda sobre os componentes da CN, existe o papel da comutação por pacote realizada pelo *Serving GPRS Support Node* (SGSN), que é um roteador com funções adicionais de autenticação, autorização e bilhetagem de usuários. O SGSN comunica com o *Gateway GPRS Support Node* (GGSN) que também é um roteador

responsável pela interface com redes IP de dados representada na Figura 2 pela internet.

Dentro do contexto da UTRAN, existem componentes denominados Nodes B, que acomodam os equipamentos de rádio conectados ao *Radio Network Control* (RNC) respectivo, os quais são responsáveis pelas conexões físicas e lógicas entre os equipamentos dos usuários e o CN. Para cada Node B é estabelecida uma área de cobertura, tecnicamente chamada de célula. Segundo Figueiredo (2004), as solicitações de serviços são encaminhadas para o Node B, onde são tratadas no Controle de Admissão de Chamadas (CAC) presente no *Radio Network Controller* (RNC). Independente de qual seja a tecnologia implementada no CAC, conceitualmente ele decide se aceita ou bloqueia a nova solicitação de serviço. Quando se leva em consideração a necessidade de Qualidade de Serviço (QoS), torna-se fundamental implementar técnicas que possibilite que o CAC faça uma melhor gestão dos recursos.

2.2 Taxa de queda de chamada

Um dos principais fatores de qualidade de serviço que deve ser continuamente observado pelas operadoras de telefonia móvel celular é a interrupção indesejada de chamadas nos processos de Handover, este fenômeno é conhecido como queda de chamada (QC). A QC é um fator primordial na satisfação do usuário, pois, caso o usuário experimente um número excessivo de QC, provavelmente ele irá optar por outra operadora (HALONEN; ROMERO; MELERO, 2003).

A taxa de queda de chamada (TQC) é definida como a razão entre o número total de chamadas interrompidas de forma indesejada e o número total de chamadas que foram iniciadas em uma rede, e, mais especificamente, em uma determinada célula. Assim, a TQC contabiliza a porcentagem de vezes em que houve o desligamento indesejado da chamada em relação ao total de chamadas que foram originadas em uma determinada célula.

Reduzir a TQC tem impacto direto para o usuário e para a operadora, principalmente se essa redução for alcançada sem que haja necessidade de novos investimentos em infraestrutura, o que reduz sensivelmente os custos com a operação e a instalação da rede (MWANSA; HANRAHAN, 2004).

Para a operadora, a redução na TQC tem forte apelo financeiro por três motivos. Primeiro, quanto maior o tempo de duração da ligação telefônica, maior a arrecadação da operadora, pois, geralmente a tarifação é por minuto. Segundo, tem impacto na satisfação do usuário e, portanto, a manutenção deste em sua base de assinantes. Terceiro, evita multas com o descumprimento das metas de qualidade na prestação de serviço que são definidas e monitoradas pela Agência Nacional de Telecomunicações (HALONEN; ROMERO; MELERO, 2003).

2.3 QoS

QoS é capaz de distinguir tipos de serviços e/ou de tráfego, de forma que as aplicações possam verificar diversos comportamentos para diferentes classes de tráfego. Para fazer uma análise de QoS, é necessário observar o que é mais importante para cada aplicação ou classe de serviço. O tráfego é todo por classes de serviços. Operar com QoS significa dispor das melhores condições que a rede viabiliza. Cada classe precisa avaliar um determinado parâmetro de rede para obter uma avaliação de QoS mais precisa. (STEINMETZ; WOLF, 1997, p.211).

A análise de QoS nas redes UMTS é feita pelas classes de serviços definidas pelo grupo 3GPP (GENERATION..., 2011b), que são: *Conversational*, *Streaming*, *Interactive* e *Background*. Essas classes representam tipos de aplicações com requisitos distintos.

Cada uma dessas classes possui suas exigências nos parâmetros de desempenho da rede:

- a) *Conversational* = baixa tolerância a atrasos. Exemplo: em aplicativos de voz;
- b) *Interactive* = baixa tolerância a vazão baixa. Exemplo: acesso ao banco de dados, navegação web, jogos em rede;
- c) *Background* = baixa tolerância à perda de pacotes. Exemplo: e-mail, SMS;
- d) *Streaming* = baixa tolerância a jitter alto. Exemplo: multimídia, vídeo sob demanda.

É necessário medir alguns parâmetros de QoS para conhecer o comportamento de uma rede. Através dessa medição é possível ver se estão acontecendo congestionamentos na rede, determinar a vazão da rede em um

determinado instante, verificar o tempo que os pacotes levam para chegar ao seu destino e assim fazer as melhorias necessárias para que o comportamento da rede seja melhorado.

Os principais parâmetros utilizados para medir a QoS da rede são:

- a) *Delay* ou atraso é o tempo gasto para que um pacote seja enviado do transmissor até o receptor. Esse parâmetro sofre influências dos meios de comunicação e de congestionamentos na rede;
- b) Jitter ou variação do atraso é a diferença de atraso entre pacotes enviados e recebidos na rede;
- c) Número de pacotes perdidos que está relacionado à confiabilidade da transmissão;
- d) *Throughput* ou vazão é a taxa efetiva de pacotes transmitidos por uma unidade de tempo.

Na release 23.207 da 3GPP (GENERATION...,2011a), são definidas as medidas de QoS para cada classe de serviço. Com essas definições, é possível avaliar se um determinado usuário tem qualidade de serviço ou não.

Conforme demonstra (Figura 3), extraída do referido release, apresenta os requisitos dos principais parâmetros de QoS de cada classe de serviço: *delay*, jitter e *throughput*.

Figura 3 - Requisitos de QoS na rede UMTS

Classe	Aplicação	Demanda	Atraso		Jitter	Vazão
<i>Conversational^a</i>	Conversa de voz	Alta	Preferencial Limite	< 150 ms < 400 ms	< 1 ms	4-13 kbps
	Mensagem de voz	Alta	Receber Gravar	< 1 s < 2 s	< 1 ms	4-13 kbps
	Videoconferência	Baixa	Preferencial Limite	< 150 ms < 400 ms	–	32-384 kbps
	Telnet	Baixa	Limite	< 250 ms	–	32 kbps
<i>Streaming</i>	<i>Streaming</i> de áudio de alta qualidade	Baixa		< 10 s	< 1 ms	32-128 kbps
	Video <i>one-way</i>	Baixa		< 10 s	< 2 s	32-384 kbps
	Transferência de dados	Alta		< 10 s	–	< 8 kbps
	Imagem	Baixa		< 10 s	–	< 800 kbps
	Telemetria (monitoramento)	Baixa		< 10 s	–	< 28,8 kbps
<i>Interactive^b</i>	Telemetria (controle)	Baixa		< 250 ms	–	< 28,8 kbps
	Navegação na web	Alta		< 4 s ^c	–	> 20 kbps
	Jogos interativos	Média		< 250 ms	–	< 23 kbps
	E-mail (acesso do servidor)	Alta		< 4 s	–	< 20 kbps
<i>Background</i>	Fax	Alta		< 30 s	–	9,6 kbps
	E-mail (servidor para servidor)	Alta		3 horas	–	0,074 kbps
	<i>Usenet</i>	Média		24 horas	–	1 kbps
	SMS	Média		< 30 s	–	2,8 kbps

Fonte: GENERATION..., 2011a

O atraso (*delay*) é o intervalo de tempo em que o pacote percorre a rede até atingir seu destino. A release 23.207 (GENERATION...,2011a) especifica que esse atraso não deve ser superior a 400 ms (milissegundos) para *Conversational*, 10 s para *Streaming* e 4 s para *Interactive*, lembrando que estas 3 classes são aplicações de Tempo Real – *Real Time* (RT). Já para a aplicação Não Tempo Real – *Not Real Time* (NRT), o máximo de atraso deve ser 30 s para aplicações *Background*.

O jitter é a medida de variação do atraso (KUROSE; ROSS, 2005), ou seja, a diferença sequencial dos atrasos de pacotes. É importante que o jitter seja reduzido ao nível mais baixo possível, sendo seu limite máximo aceitável de 1 ms para *Conversational* e 2 s para *Streaming*, únicas classes de serviços sensíveis ao jitter.

A vazão (*throughput*) é a taxa efetiva de transmissão, medida em *bits* por segundo. Esta medida está relacionada ao tempo que o dado gasta para ir do transmissor ao receptor. Cada aplicação possui uma determinada vazão, sendo 3-14 kbps para *Conversational*, 32-384 kbps para *Streaming*, maior que 20 kbps para *Interactive* e 2.8 kbps para *Background*.

Enfim, a utilização de QoS favorece muito a qualidade das aplicações, principalmente as aplicações em tempo real como a classe *Conversational*. Uma grande diferença entre uma aplicação de dados e uma aplicação de voz é que uma aplicação de voz é bem mais sensível ao atraso. Desta forma, a qualidade de serviço da rede depende das necessidades da aplicação, ou seja, do que ela requisita da rede a fim de que funcione bem e atenda, por sua vez, às necessidades do usuário. Por fim, estes requisitos são traduzidos em parâmetros indicadores do desempenho da rede.

2.4 Lógica Nebulosa

Segundo Simões e Shaw (2007), é possível que um operador humano seja capaz de controlar diversos sistemas sem compreender a matemática, ou todos os detalhes envolvidos. Uma pessoa possui inteligência capaz de manipular variáveis de entrada que influenciem as saídas do processo. Esta fundamentação levou ao enfoque do conceito de Inteligência Artificial na teoria de processos (complexos). A inteligência artificial compreende uma família de ferramentas para tratar problemas complexos.

A tomada de decisão é uma expressão utilizada para ilustrar situações em que uma entidade deve escolher um caminho a fim de solucionar um problema. Em termos computacionais, é muito complexo saber se a decisão que se toma é correta ou não. A Inteligência Artificial viabiliza a passagem desta decisão meramente lógica para uma decisão intuitiva e bem elaborada (STASZCZAK; LIZELAINE; MOLETTA, 2004).

Dentro do contexto da rede UMTS, é necessário tomar decisões constantemente sobre o Controle de Admissão de Chamadas (CAC). Estas decisões estão relacionadas com os recursos da rede no momento da chamada. Nesta circunstância, esta decisão se torna importante, pois desta determinará a QoS. Se essa decisão for errada, as consequências serão negativas, portanto há a necessidade de que essa decisão seja bem tomada.

Até o presente momento, existe um grande descompasso entre a capacidade criativa dos seres humanos e a possibilidade de solução que as máquinas computacionais proporcionam, devido ao fato de que as pessoas raciocinam de forma incerta, imprecisa, difusa ou nebulosa, enquanto que as máquinas e

computadores são movidos por raciocínio preciso e binário. A eliminação de tal restrição faria com que as máquinas fossem inteligentes, isto é, pudessem raciocinar da mesma maneira imprecisa, como os seres humanos. Tal forma de raciocínio é chamada em inglês por *Fuzzy* e será aqui utilizada como sinônimo de incerto, impreciso, difuso ou nebuloso (STASZCZAK; LIZELAINE; MOLETTA, 2004).

O objetivo deste trabalho é propor a aplicação da Lógica *Fuzzy* (que a partir deste momento será tratada por seu sinônimo Lógica Nebulosa), no processo de tomada de decisão do CAC da rede UMTS.

Veja um exemplo de um mecanismo de LN que infere qual atleta de basquete é melhor a partir de informações lingüísticas, exemplo retirado de Engelbrecht (2002).

Suponha dois jogadores, Peter e Carl. Ao assumir as seguintes funções de pertinência:

$$\mu_{\text{alto}}(\text{Peter}) = 0.9 \text{ e } \mu_{\text{bom_atleta}}(\text{Peter}) = 0.8$$

$$\mu_{\text{alto}}(\text{Carl}) = 0.9 \text{ e } \mu_{\text{bom_atleta}}(\text{Carl}) = 0.5$$

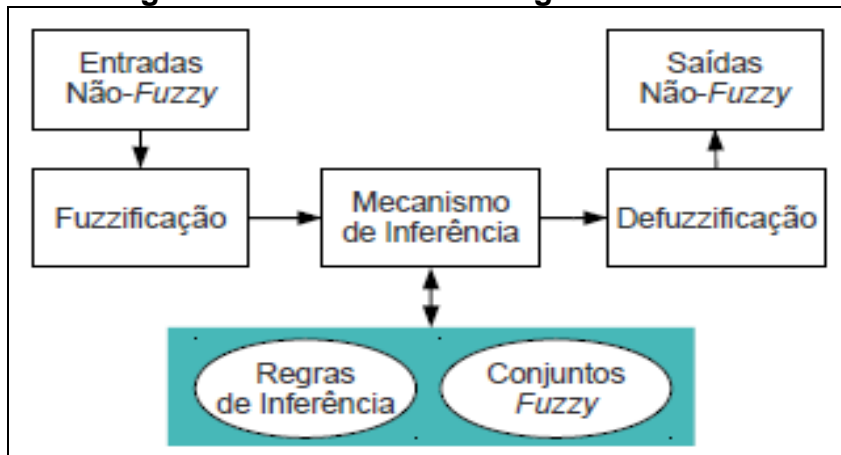
E sabendo que um bom atleta de basquete, além de ser um bom, deve ser alto, pode-se concluir que Peter é melhor atleta que Carl. Isso devido a:

$$\mu_{\text{bom_atleta_basquete}}(\text{Peter}) = \min\{0.9, 0.8\} = 0.8$$

$$\mu_{\text{bom_atleta_basquete}}(\text{Carl}) = \min\{0.9, 0.5\} = 0.5$$

Esse tipo de regra, em que um bom atleta de basquete deve ser alto e um bom atleta, é definida em um mecanismo de LN. Para esse pequeno exemplo, a resolução é trivial. Contudo, problemas do mundo real envolvem mais variáveis e a solução se torna mais complexa, o que exige uma solução computacional.

Figura 4 - Mecanismo de Lógica Nebulosa



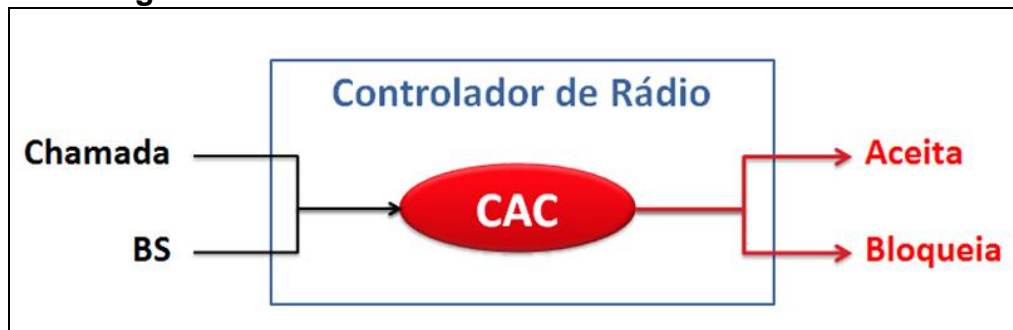
Fonte: Adaptada de ENGELBRECHT, 2002

Um mecanismo de LN possui a capacidade de, a partir de um conjunto de regras e de um conjunto de situações ativadas (conjunto *fuzzy*), inferir uma ação. Conforme (Figura 4) mostra a arquitetura de um mecanismo de LN. Entre seus componentes estão o processo de fuzzificação, defuzzificação, um mecanismo de inferência, um conjunto de regras de inferência e conjuntos *fuzzy*. Juntos, as regras e os conjuntos *fuzzy* formam uma base de conhecimento, que será consultada pelo mecanismo de inferência para calcular a saída do sistema. O processo de fuzzificação é responsável por converter os valores reais de entrada (entradas não-*fuzzy*) para valores *fuzzy* (variáveis linguísticas). Após essa etapa, o mecanismo de inferência, a partir das regras definidas, calcula a saída. Depois disso, ocorre o processo de defuzzificação, em que as saídas *fuzzy* são convertidas para valores reais (não-*fuzzy*). Entre os algoritmos de defuzzificação estão a Média dos Máximos e o Centro de Gravidade (RAO; RAO, 1995). Na Média dos Máximos, a saída é obtida tomando-se a média entre os dois elementos extremos no universo que correspondem aos maiores valores das funções de pertinência. Já no Centro de Gravidade, a saída é o valor no universo que divide a área sob a curva da função de pertinência em duas partes iguais. Este trabalho utilizará o algoritmo Centro de Gravidade, onde o valor numérico de saída corresponde ao valor da abscissa que divide pela metade a área da função de pertinência, gerada pela combinação das regras de inferência.

2.5 Controle de Admissão de Chamada (CAC)

De acordo com Silva e Costa (2008), uma chamada é qualquer solicitação para utilizar recursos da rede para serviços como voz, vídeo, SMS e web. Cada chamada possui suas características e requisitos que devem ser atendidos para ter o mínimo de Qualidade de Serviço (QoS). Quando uma chamada solicita utilização de recursos de rede, ela pode ser aceita ou bloqueada dependendo das condições da rede.

Figura 5 - Controle de Admissão de Chamadas – CAC

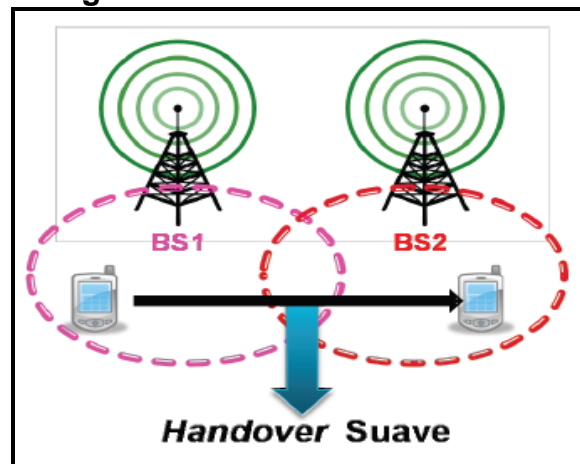


Fonte: Adaptada de RIBEIRO, 2010

Conforme demonstra (Figura 5), o mecanismo que gerencia a aceitação ou bloqueio de chamadas é conhecido como Controle de Admissão de Chamadas (CAC) e seu bom funcionamento é fundamental para o desempenho da rede.

O principal parâmetro na avaliação de um CAC é a garantia de QoS, pois se o CAC não decidir adequadamente sobre a aceitação ou bloqueio de uma nova requisição de chamada, ocorrerá congestionamento ou desperdício de recursos. Além da satisfação do usuário com a utilização do serviço, existem outros parâmetros a serem respeitados para mensurar a qualidade do serviço. Os parâmetros de QoS são o atraso, o jitter e a vazão por usuário, estes parâmetros são especificados pelo 3GPP (GENERATION..., 2011b).

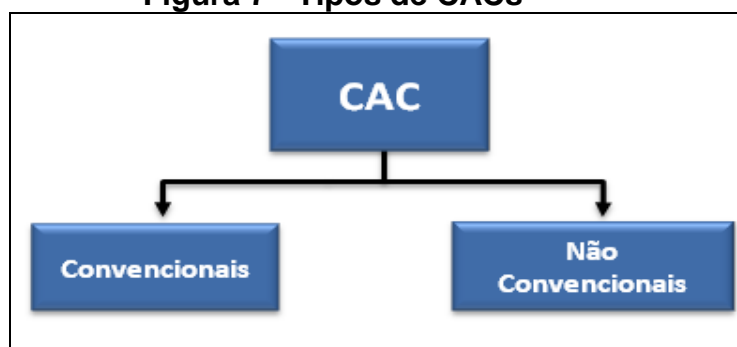
Figura 6 - Processo de Handover



Fonte: REIS, 2010

Conforme demonstrado (Figura 6), existe também um tipo especial de solicitação para utilizar a rede, que é o *handover* suave (SHO). Segundo Reis (2010), *handover* é a passagem do dispositivo móvel de uma célula para outra. O *handover* pode ocorrer de duas maneiras: hard (abrupto), onde há interrupção do sinal com perda da conexão e soft (suave), onde a mudança é feita sem perda do sinal. Isso ocorre, pois existem dois caminhos ativos e dois fluxos correspondentes, pelo menos por algum tempo, para o dispositivo portátil. Neste trabalho será considerado apenas o *handover* suave (SHO).

Figura 7 - Tipos de CACs



Fonte: Adaptado de GHADERIE et al., 2006

Com relação aos CACs, estes podem ser categorizados em:

- a) CAC Convencional;
- b) CAC Não Convencional.

Os CACs convencionais são os que não utilizam Inteligência Computacional (CI), enquanto os CACs Não Convencionais se apóiam nas vantagens que a CI pode oferecer neste contexto de admissão de chamadas. As vantagens dos CACs convencionais são a alta eficiência, a baixa complexidade e o baixo *overhead* (RIBEIRO; FIGUEIREDO, 2011).

O CAC-J foi proposto por Antoniou (2004) e trata-se de um CAC convencional, ele controla a admissão de novas chamadas considerando apenas o controle de potência.

Já os CACs não convencionais, são adaptativos e estáveis. Novas implementações com CI para CACs foram propostas pela comunidade acadêmica, uma solução interessante foi o trabalho de Ribeiro (2010), onde foi criado um CAC-RD Neural (CAC-RDN), a partir de uma Rede Neural Artificial (RNA) treinada, em substituição a arquitetura de um CAC convencional, conhecido como CAC-RD. Esta solução permitiu representar o comportamento do próprio CAC-RD, viabilizando a expansão do número de usuários no cenário de simulação.

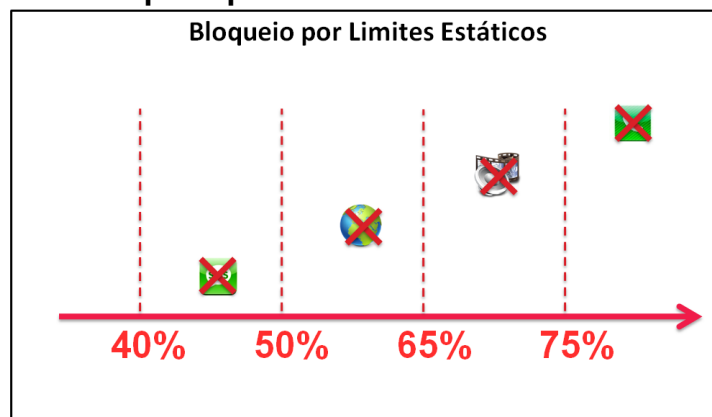
As técnicas possíveis de serem aplicadas em CACs não convencionais são:

- a) Rede Neural Artificial;
- b) Lógica Nebulosa;
- c) Algoritmos Genéticos;
- d) Estatística.

Vários esforços estão sendo feitos pela comunidade acadêmica no sentido de agregar valor aos CACs utilizando LN, segundo Pedrycz W. Vasilakos (2000), os CACs baseados em LN conseguem maximizar a utilização de recursos pela implantação de limites dinâmicos.

Reserva de Canais do CAC-RD, este procedimento originou o problema do não tratamento de chamadas SHO enfrentado pelo CAC-RDF, pois, após a desativação deste módulo de Reserva de Canais todas as chamadas SHO passaram a receber o mesmo tratamento das demais chamadas, sem distinção. A intenção do CAC-RDF foi tratar as chamadas de acordo com sua prioridade em termos de classe de serviços, aplicando os limites dinâmicos, sem considerar se a chamada é do tipo SHO.

Figura 10 - Bloqueio por Limites Estáticos do CAC-RD



Fonte: Elaborado pelo autor

A ineficiência dos limites estáticos do CAC-RD pode ser melhor visualizada (Figura 10), pois, quando a rede está livre e só ocorrem chamadas menos prioritárias, o CAC irá aceitar essas chamadas menos prioritárias somente até certo limite de utilização (por exemplo: o limite *Background* é 40 por cento). Logo, o CAC-RD bloqueia novas chamadas sem verificar se a QoS está dentro dos limites aceitáveis. O comportamento desejável seria a aceitação de chamadas até que a QoS não seja mais garantida, do contrário como é feito, ocorre desperdício de recursos.

A idéia do CAC-RDF é o aprimoramento do módulo de bloqueio por limites do CAC-RD, com a utilização de limites dinâmicos. A proposta é que os limites se ajustem ao estado e a QoS da rede. Se a rede estiver ociosa, não ocorrerão bloqueios, mas, se uma classe de serviço não tiver a QoS garantida, o CAC-RDF irá preferir as classes de serviços prioritárias e conseqüentemente, irá bloquear as demais. Assim que uma chamada em curso é finalizada, os recursos liberados por estas chamadas são atribuídos para novas chamadas, seguindo as prioridades

estabelecidas. A idéia do CAC-RDF é aceitar chamadas sem distinção, enquanto não houver evidências de congestionamento.

Este CAC-RDF consegue utilizar com eficiência a vantagem da Lógica Nebulosa no tratamento de incertezas e, conseqüentemente, viabilizar limites dinâmicos na aceitação de novas chamadas. Todas as chamadas aceitas pelo CAC-RDF respeitam os parâmetros de QoS definidos pelo 3GPP (GENERATION..., 2011b). A tomada de decisão dinâmica maximizou a aceitação de novas chamadas, evitando os desperdícios de recursos que ocorria quando os limites eram estáticos.

Conforme apresentado neste capítulo, existem várias técnicas de CACs apresentadas na literatura. Dentre todos levantados, o CAC-RDF é o que apresenta resultados mais interessantes, mas deixa em aberto o problema no tratamento das chamadas SHO.

3 METODOLOGIA DE PESQUISA

Abaixo, são descritas as etapas que foram seguidas para alcançar o objetivo deste trabalho:

- a) Estudos sobre o CAC-RDF;
- b) Proposta de um mecanismo de Lógica Nebulosa para criação de um CAC-RDFF: aplicação deste mecanismo de LN em cascata para resolver o problema do *handover*;
- c) Aplicação do procedimento de LN em cascata no CAC-RDF, propondo o CAC-RDFF;
- d) Avaliação do CAC-RDFF, através de simulações e comparações com o CAC-RDF;
- e) Análise estatística dos resultados obtidos;
- f) Levantamento de conclusões, vantagens e desvantagens da abordagem.

As simulações do CAC-RDF e CAC-RDFF foram realizadas no módulo E-UMTS, desenvolvido no NS-2. Nesse módulo, já se encontra implementado o CAC-RDF. Para simular o tráfego, foram criadas fontes para cada classe de serviço, sendo quatro fontes: uma para a classe de NRT (*Background*) e uma para cada classe de RT (*Conversational*, *Streaming* e *Interactive*).

Neste trabalho, na representação das classes, sempre são consideradas as aplicações descritas abaixo:

- a) Conversa de voz para a classe *Conversational* (RT);
- b) Vídeo one-way para *Streaming* (RT);
- c) Web-browsing para *Interactive* (RT);
- d) SMS para *Background* (NRT).

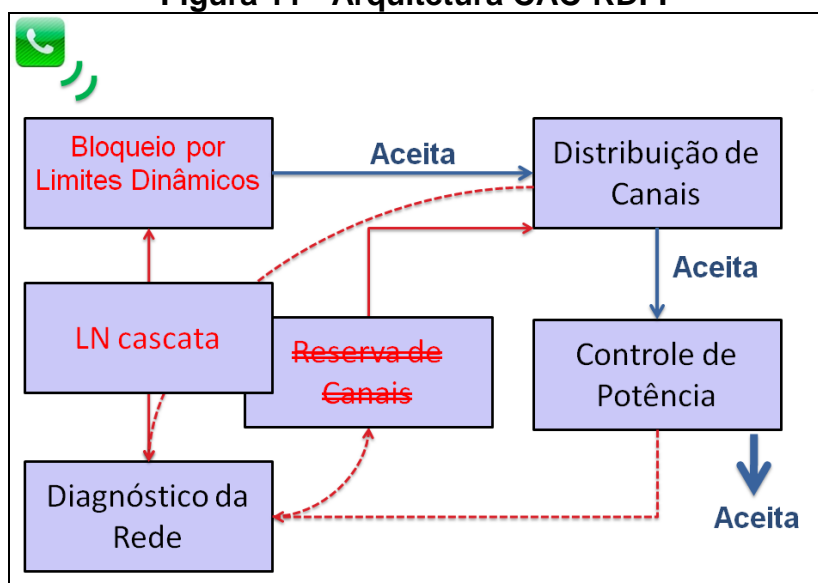
4 REPRESENTAÇÃO DE LÓGICA NEBULOSA EM CASCATA

É admitido que não seja possível atingir em CACs uma baixa complexidade, alta adaptabilidade e alta estabilidade sem aumentar o *overhead*, esta regra vale para CACs convencionais. Baseado nestas características, a aplicação de um processamento inteligente torna-se uma solução viável (RIBEIRO, 2010).

De acordo com Simões e Shaw (2007), a Lógica Nebulosa é ideal para o tratamento de incertezas, logo, sua aplicação em CACs torna-se interessante, pois nestes cenários não existe garantia de estabilidade, ou seja, a Lógica Nebulosa pode modelar as incertezas do estado e da QoS da rede para decidir a admissão de uma nova chamada.

O objetivo central deste trabalho foi a proposta de uma solução para o problema em aberto do CAC-RDF, que se refere ao não tratamento de chamadas SHO. Conforme já explicado, a desativação do módulo de Reserva de Canais no CAC-RDF originou o problema do não tratamento de chamadas SHO, pois, após a desativação deste módulo de Reserva de Canais, todas as chamadas SHO passaram a receber o mesmo tratamento das demais chamadas, sem distinção. A intenção do CAC-RDF foi tratar as chamadas de acordo com sua prioridade em termos de classe de serviços, aplicando os limites dinâmicos, sem considerar se a chamada é do tipo SHO.

Figura 11 - Arquitetura CAC-RDFF

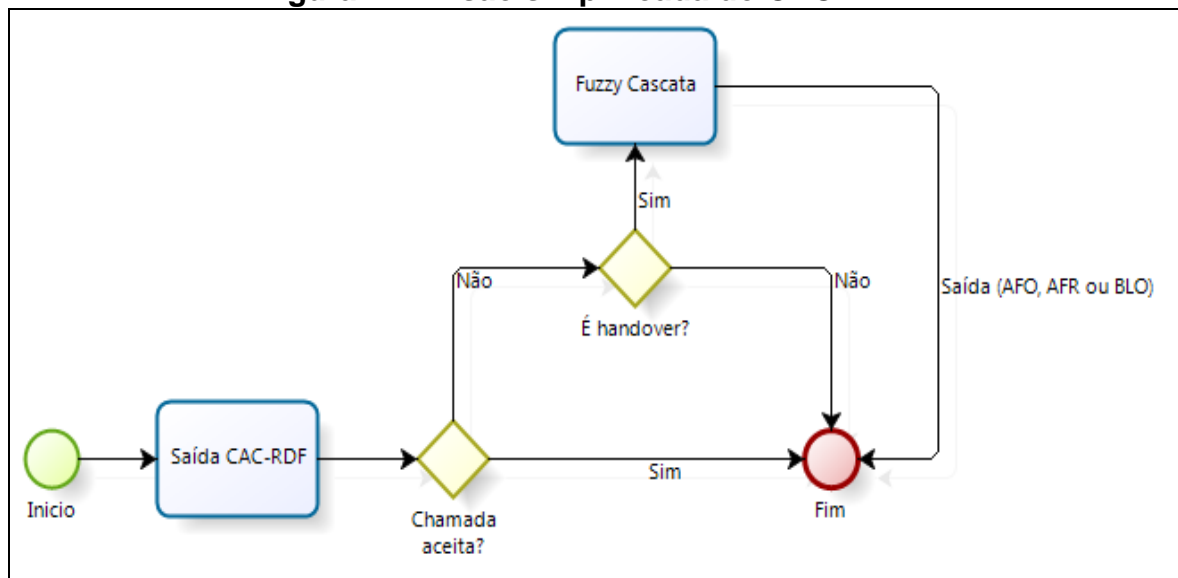


Fonte: Adaptado de RIBEIRO, 2010

Como pode ser visualizada (Figura 11), a Figura ilustra a arquitetura do CAC-RDFF, onde foi desativado o módulo de Reserva de Canais, originário do CAC-RD. Este módulo de Reserva de Canais do CAC-RD calcula, dinamicamente, o percentual de canais que devem ser reservados para usuários de SHO. O cálculo, baseado em Lindemann et al, 2003, propõe reserva entre 3% e 70% dos canais para SHO. O cálculo do percentual de canais reservados é uma média das probabilidades de falhas de *handovers* e dos bloqueios de novas chamadas, multiplicado pelo valor antigo deste percentual (FIGUEIREDO, 2004). Logo, apesar do módulo de Reserva de Canais ser uma técnica funcional, a demanda de processos de SHO é obscura no sentido de novas chamadas.

Para modelar esta incerteza sobre a aceitação da chamada *handover*, de acordo com o serviço requisitado e a utilização da rede, foi apresentada esta solução que cria um Mecanismo de LN em cascata, em que são estabelecidos níveis de aceitação como: aceita fracamente (com uma possível redução da largura de banda requisitada) e aceita fortemente (com largura de banda requisitada total). Enfim, o desafio deste trabalho foi integrar as áreas de Inteligência Computacional e Computação Móvel.

Figura 12 - Visão simplificada do CAC-RDFF



Fonte: Elaborado pelo autor

Acompanhando (Figura 12), percebe-se a abordagem deste trabalho segue a utilização do CAC-RDF e, caso a chamada seja bloqueada, é feito um teste lógico

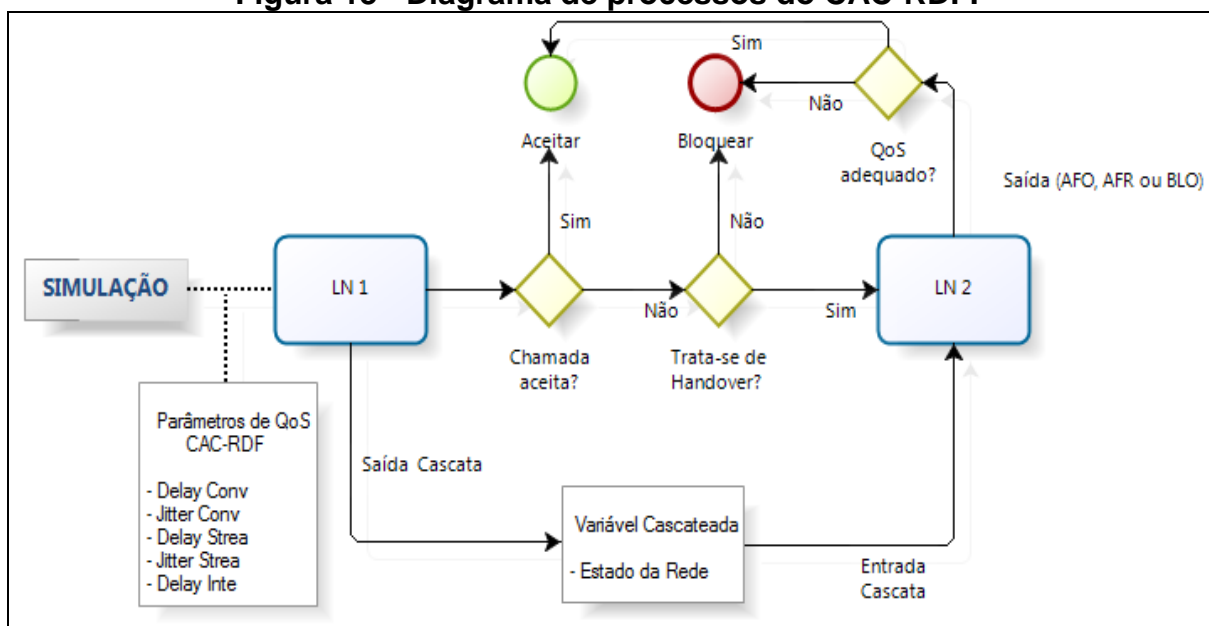
para verificar se a chamada é do tipo *handover*. Se for, é acionada uma segunda camada LN em cascata, onde uma das entradas é a saída da primeira camada LN.

Como saída desta segunda camada LN, existem três possibilidades, listadas a seguir:

- Aceitação forte, onde é garantido o atendimento dos requisitos de QoS;
- Bloqueio;
- A aceitação fraca, onde não é garantido o atendimento dos requisitos de QoS.

A aceitação fraca (sem garantia de QoS) tem o foco na priorização da satisfação do usuário em relação à disponibilidade da rede, pois, estamos tratando chamadas *handover*. Supõe-se, nesse caso, que o usuário prefere continuar com seu serviço, mesmo que tenha algum prejuízo tal como o aumento de latência, ao invés de ter seu serviço finalizado de maneira abrupta.

Figura 13 - Diagrama de processos do CAC-RDFF



Fonte: Elaborado pelo autor

Conforme mostra (Figura 13), a primeira estrutura de Lógica Nebulosa do CAC-RDF possui como variáveis de entrada os seguintes parâmetros de QoS: atraso e jitter para *conversational*; atraso e jitter para *streaming*; atraso de *interactive*. A classe *background* não precisa entrar neste contexto de QoS, porque é a classe menos prioritária e a primeira a ser bloqueada em caso de

congestionamento na rede. A variável de saída desta primeira estrutura de Lógica Nebulosa determina o Estado da Rede. Na segunda estrutura de Lógica Nebulosa, incorporada pelo CAC-RDFF, as variáveis de entrada são Estado da Rede e Largura de Banda Requisitada (LBR). Através da variável Estado da Rede, a Lógica Nebulosa em cascata é modelada, pois esta variável é a saída da primeira estrutura de Lógica Nebulosa e também a entrada da segunda, ocasionando um ganho computacional pela não necessidade de se efetuar cálculos sobre os parâmetros que definem o status da rede. Como variável de saída da segunda estrutura de Lógica Nebulosa, foi definida a decisão do CAC, que pode ser uma aceitação forte (AFO) que garante QoS adequado para a classe de serviço (baseado nos requisitos definidos pelo 3GPP), aceitação fraca (AFR) que não garante QoS ou bloqueio (BLO) da chamada do tipo *handover*.

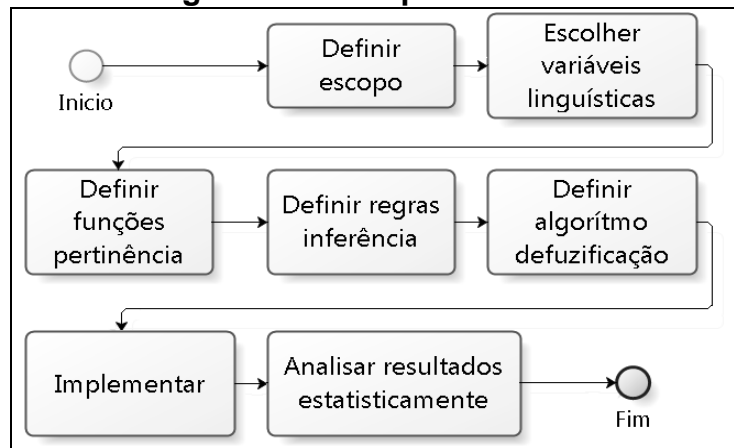
Com esta solução de LN em cascata, foi aperfeiçoada a decisão das solicitações de SHO, pois a LN permite o tratamento de incertezas (imprecisões) recorrentes no cenário de admissão de solicitações de serviços em ambientes de rede UMTS.

Além disso, a LN possui uma capacidade linguística, permitindo modelar a complexidade do tráfego com valores linguísticos (baixo, médio, alto). Isso auxilia na modelagem das variáveis e relacionamentos. Esta característica de modelagem, que permite manipulações de variáveis linguísticas e regras de inferência, é que fazem da LN uma ótima teoria para tratamento de incertezas.

Assim como a camada LN do CAC-RDF, a segunda camada LN do CAC-RDFF foi implementada no agente *Radio Resource Management* (RRM), que fica no RNC, onde está implementado o CAC, no módulo E-UMTS do NS-2.

A partir deste ponto, vamos tratar da implantação desta segunda camada LN, que foi baseada na metodologia apresentada em Ribeiro (2010) para desenvolvimento do mecanismo de LN.

Este trabalho seguiu os sete passos (Figura 14), descritos a seguir e detalhados, conforme utilização neste trabalho, nas próximas subseções.

Figura 14 - Sete passos

Fonte: Elaborado pelo autor

- a) Definir situações onde poderia aceitar chamadas *handover* que foram bloqueadas no primeiro mecanismo de LN;
- b) Escolher as variáveis linguísticas (variáveis de entrada e saída do mecanismo de LN);
- c) Definir as funções de pertinência das variáveis linguísticas;
- d) Definir as regras Nebulosas para o mecanismo de LN;
- e) Escolher algoritmo de defuzzificação;
- f) Implementar o mecanismo de LN como módulo no CAC;
- g) Testar a eficiência da solução.

4.1 Escopo da modelagem

O primeiro passo do procedimento envolve um estudo em profundidade dos impactos a QoS da rede, se ocorrer a aceitação de uma chamada *handover* que foi tecnicamente bloqueada no primeiro mecanismo de LN. O CAC-RDF possui 4 limites para iniciar bloqueios de chamadas, um para cada classe de serviço. Será respeitada a seguinte prioridade de classes abordada no CAC-RDF: *Conversational*, *Streaming*, *Interactive* e *Background*. A classe de maior prioridade é a *Conversational* e a de menor prioridade é a *Background*.

4.2 Escolha das variáveis linguísticas

O segundo passo do procedimento envolve a escolha das variáveis linguísticas. Esta etapa é crítica para o desempenho do mecanismo de LN. Para manter a mesma idéia do CAC-RDF, a variável de saída é a decisão do módulo. Ou seja, como saída são admitidos os seguintes valores, já mencionados anteriormente:

- a) Aceita fortemente: a chamada *handover* é aceita com garantia de QoS.
- b) Aceita fracamente: a chamada *handover* é aceita, mas sem garantia de QoS;
- c) Bloqueio: a chamada *handover* é bloqueada.

É válido ressaltar que se trata de um segundo mecanismo de LN, que só é acionado se houver bloqueio de chamadas *handover* no primeiro mecanismo de LN. Caso contrário, o fluxo é seguido normalmente sem que o segundo mecanismo de LN seja acionado.

No que diz respeito às variáveis de entrada, além do Estado da Rede que nada mais é do que a saída do primeiro mecanismo de LN, temos uma segunda variável que se refere à largura de banda requisitada pela classe de serviço. Este procedimento do controle de admissão atuará sempre que uma nova chamada for bloqueada no primeiro mecanismo de LN e se tratar de uma requisição de *handover*, com isso, o segundo mecanismo de LN irá verificar a largura de banda requisitada comparando-a com o estado da rede, logo, esta variável é importante para aferir o impacto nos parâmetros de QoS, caso ocorra aceitação da chamada *handover* previamente bloqueada no primeiro mecanismo de LN.

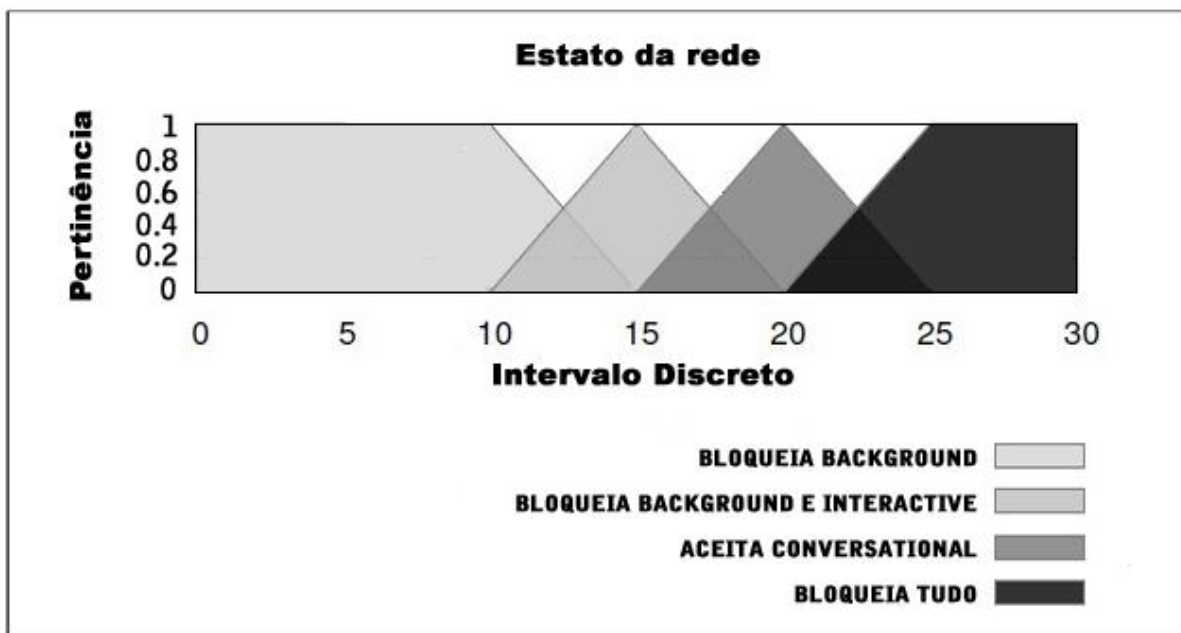
4.3 Definição das funções de pertinência

O terceiro passo do procedimento consiste na definição das funções de pertinência das variáveis, que representam os valores reais em números linguísticos. Todas as variáveis, inclusive as variáveis de saída, possuem função definida.

As funções de pertinência definem valores linguísticos para valores determinísticos em um universo de discurso, baseado no grau de pertinência, ou seja, baseado no quanto se pode afirmar e ter certeza do evento em questão. Este grau de pertinência varia de zero a um. Zero ocorre quando assumimos que a

variável não possui aquele determinado valor linguístico e um quando a variável definitivamente possui aquele determinado valor linguístico. Se, por exemplo, a variável possui grau de pertinência 0,8 para o rótulo baixo e de 0,4 para o rótulo médio, podemos dizer que esta variável é mais baixa do que média. O gráfico (Figura 15) mostra as funções de pertinência do Estado da Rede.

Figura 15 - Função de pertinência do Estado da Rede



Fonte: Elaborado pelo autor

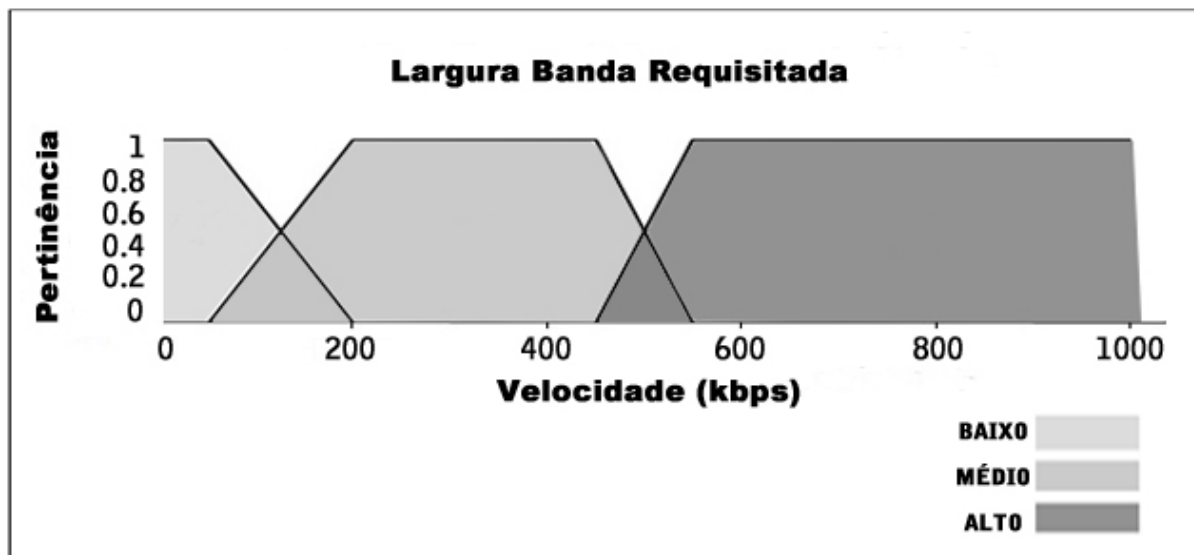
O gráfico (Figura 15) é a saída do primeiro mecanismo de LN, que serve de entrada para o segundo mecanismo de LN. A única particularidade é o aproveitamento de apenas 4 das 5 regiões que representam as variáveis de saída possíveis do primeiro mecanismo de LN. A região não aproveitada é a que aceita todas as chamadas, pois, nesta circunstância, a chamada *handover* será aceita, o que torna sem efeito o segundo mecanismo de LN, pois o objetivo deste é minimizar os bloqueios de chamadas *handover*. Logo, as demais regiões representadas na função de pertinência referem-se a:

- Bloqueia serviços *Background*;
- Bloqueia *Background* e *Interactive*;
- Bloqueia *Background*, *Interactive* e *Streaming*;
- Bloqueia todas as classes.

O objetivo desta função de pertinência é mensurar o estado da rede, aproveitando o processamento do primeiro mecanismo de LN, maximizando o desempenho na tomada de decisão. O objetivo desta abordagem no CAC-RDF foi modelar a incerteza da decisão de bloqueios de chamadas, contidas nos limites estáticos do CAC-RD. De acordo com Ribeiro (2010), para transformar para um valor discreto essa incerteza na utilização de uma BS (*Base Station*), ao invés de utilizar percentual e limites estáticos, seu universo de discurso foi definido como o intervalo discreto 0-30, dividido entre cada um dos estados da variável. Entende-se como incerteza a forma da função de pertinência dos valores linguísticos e as regiões de interseção.

A segunda função de pertinência refere-se à Largura de Banda Requisitada, que objetiva a modelagem de incerteza da demanda por banda, pois, para cada classe de serviço é alocada uma largura para atender aos parâmetros de QoS. O gráfico (Figura 16) mostra as funções de pertinência da Largura de Banda Requisitada.

Figura 16 - Função de pertinência da Largura de Banda Requisitada



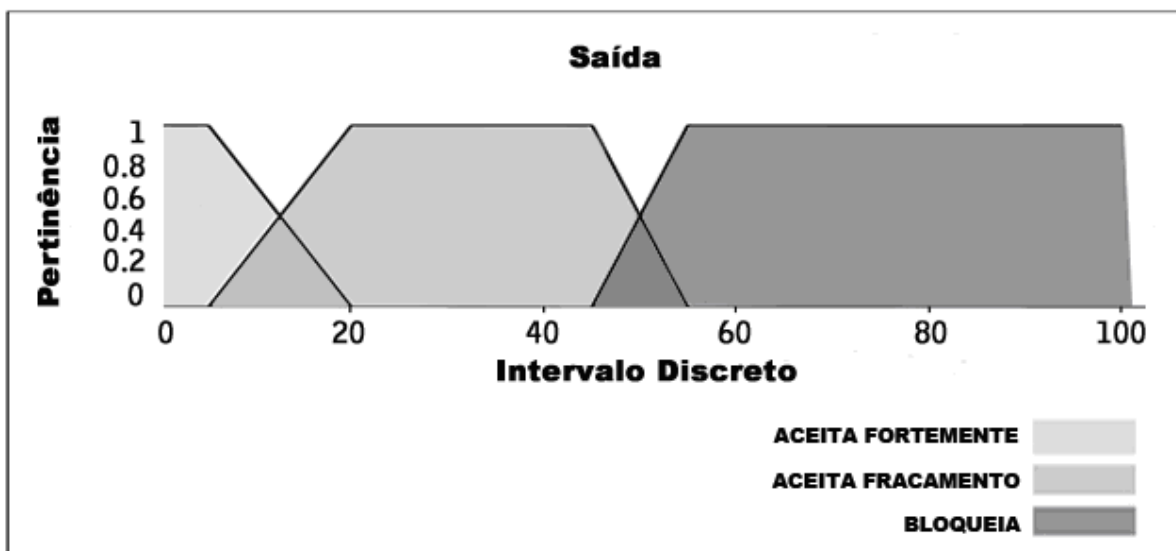
Fonte: Elaborado pelo autor

A terceira função de pertinência consiste em modelar as incertezas da saída, que pode ocorrer em:

- a) Bloqueio da chamada *handover* ;
- b) Aceitação fraca (onde não são garantidos os parâmetros de QoS definidos pela 3GPP);
- c) Aceitação forte (onde são garantidos os parâmetros de QoS definidos pela 3GPP).

Para descrever dados de interesse assumindo valores finitos, o universo de discurso desta função de pertinência foi definido com o intervalo discreto 0-100. O gráfico (Figura 17) mostra as funções de pertinência da Saída.

Figura 17 - Função de pertinência da Saída



Fonte: Elaborado pelo autor

4.4 Definição das regras de inferência

Finalizando as funções de pertinência, o passo 4 consiste na definição das regras de inferência. É válido ressaltar a necessidade de um especialista para esta etapa. A principal função do especialista é definir as regras SE-ENTÃO que determinam o comportamento da Lógica Nebulosa.

De acordo com Ribeiro (2010), no mecanismo de LN do CAC-RDF foram criadas 27 regras de inferência, que são todas as combinações possíveis das variáveis. O segundo mecanismo de LN, que gera o CAC-RDFF é composto de 12 regras de inferência, que são todas as combinações possíveis de Estado da Rede e

Largura de Banda Requisitada (LBR). A idéia destas regras é minimizar o bloqueio de chamadas *handover*, objetivando a garantia de QoS.

Figura 18 - Regras de Inferência

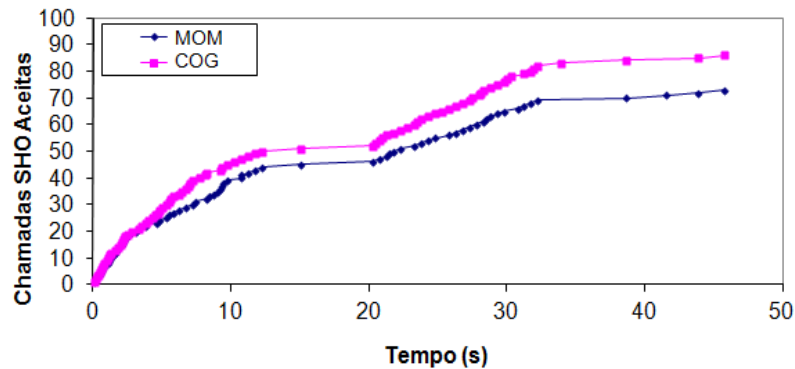
Regras de Inferência					Legenda
#	EST		LBR	SAÍDA	
1	BB	^	B	AFO	^ Operador "E"
2	BB	^	M	AFR	B Baixo
3	BB	^	A	AFR	M Médio
4	BI	^	B	AFO	A Alto
5	BI	^	M	AFR	BB Bloqueia <i>background</i>
6	BI	^	A	AFR	BI Bloqueia <i>interactive , background</i>
7	BS	^	B	AFO	BS Bloqueia <i>streaming , interative , background</i>
8	BS	^	M	AFR	BC Bloqueia tudo (inclusive <i>conversational</i>)
9	BS	^	A	BLO	AFR Aceita fracamente
10	BC	^	B	AFR	AFO Aceita fortemente
11	BC	^	M	BLO	BLO Bloqueia
12	BC	^	A	BLO	EST Estado da rede
					LBR Largura de banda requisitada

Fonte: Elaborado pelo autor

4.5 Definição do método de defuzzificação

O quinto passo do procedimento define o processo de defuzzificação. Existem vários métodos de defuzzificação, como, por exemplo, Média dos Máximos (MOM) e Centro de Gravidade (COG). O método de defuzzificação utilizado neste trabalho é o método Centro de Gravidade, que se baseia no cálculo do centro de gravidade da função de associação. De acordo com Engelbrecht (2002), calcula-se a área da curva da variável linguística de saída produzida pela máquina de inferência, e acha-se o índice correspondente que divide essa área à metade.

No estágio do defuzzificador, a variável difusa produzida pela máquina de inferência é transformada em variável numérica (determinística) que atuará no processo de forma a regulá-lo.

Gráfico 1 - Métodos de Defuzzificação

Fonte: Elaborado pelo autor

Conforme comprovação apresentada (Gráfico 1), o método de Defuzzificação Centro de Gravidade (COG) foi em média 17,81% mais eficiente que o método de Defuzzificação Média dos Máximos (MOM) para o cenário do segundo mecanismo de LN. Não foram analisadas alternativas para o método de defuzzificação do primeiro mecanismo de LN (original do CAC-RDF) que também utiliza Centro de Gravidade (COG), pois não era a proposta deste trabalho.

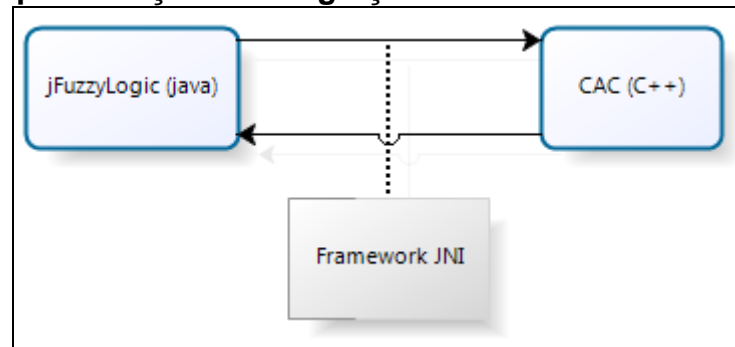
4.6 Implementação do mecanismo de LN

O sexto passo do procedimento é a implementação do mecanismo de LN. Este mecanismo de LN deve ser integrado ao módulo RRM (*Radio Resource Manager*), que por sua vez é o responsável pelo gerenciamento dos recursos de rádio da rede; é lá que se encontra o CAC. Esta implementação foi dividida em duas partes: uma no RRM – CAC enquanto a outra se encontra na biblioteca *jFuzzyLogic*, com a modelagem escrita em FCL. A Figura 19 ilustra os módulos utilizados.

Apesar da existência de várias implementações de mecanismos de LN na literatura, como por exemplo: o *fuzzyTech*; o *Awifuzzy* e o próprio *jFuzzyLogic*, a escolha do *jFuzzyLogic* se justifica pela vantagem dele ter código aberto, gratuito, portátil (Java) e implementa todo o mecanismo de LN com a Linguagem de Controle Fuzzy (FCL), padrão desenvolvido pelo *International Electrotechnical Commission* (IEC) 61131-7 (COMMISSION apud RIBEIRO; FIGUEIREDO; ZÁRATE 2011). FCL é uma linguagem específica de domínio, ou seja, só possui características relacionadas à Lógica Nebulosa. Utiliza-se FCL para definir a modelagem do mecanismo de LN, que pode ser lida e interpretada por qualquer outra linguagem,

permitindo um ganho de eficiência na manutenção do mecanismo de LN, pois, para qualquer modificação que seja necessária na modelagem de controle, basta alterar o arquivo FCL relativo ao controle. Neste trabalho, a modelagem da LN foi escrita em FCL.

Figura 19 - Representação da integração do Mecanismo LN com RRM



Fonte: Elaborado pelo autor

Levando em consideração a necessidade de o CAC enviar valores de entrada para o mecanismo de LN e receber a resposta do mesmo com a saída do mecanismo de LN, foi utilizado o *Java Native Interface* (JNI) que é um framework que permite chamar e/ou executar funções Java por aplicações escritas em outras linguagens. Logo o impasse do *jFuzzyLogic* estar em Java e o RRM em C++, foi solucionado.

4.7 Validação dos resultados

O sétimo e último procedimento é a realização da validação do mecanismo de LN. Isto implica que, dados os valores de entrada, deve ocorrer uma saída esperada, ou seja, coerente com o contexto definido pelo especialista. Se os valores de saídas forem incoerentes, a modelagem deverá ser revista.

As ferramentas mais completas de Lógica Nebulosa, existentes atualmente no mercado, são de domínio de empresas privadas e estão disponíveis apenas em versões de demonstração ou compra. Existem algumas implementações gratuitas e de código aberto, porém essas ferramentas são na sua grande maioria incompletas, sem ambientes gráficos, de difícil aprendizado, com pouca ou nenhuma documentação. Após uma revisão bibliográfica sobre a lógica nebulosa e usabilidade de *software*, foi adotada a ferramenta *InFuzzy* (POSSELT; FROZZA; MOLZ, 2011),

uma ferramenta gráfica para modelagem de aplicações de Lógica Nebulosa, dotada de recursos de simulação. Todos os arquivos necessários para o funcionamento encontram-se no instalador desta ferramenta, sendo eles: ferramenta *InFuzzy* e DLLs (*Dynamic-link library*) necessárias.

Nesta ferramenta *InFuzzy*, existe um módulo de simulação com *trace* que permite que, a cada simulação, sejam acompanhados os procedimentos realizados para a obtenção dos resultados, conforme ilustração (Figura 20):

Figura 20 - Simulação dos resultados da LN

```

===== FUZIFICAÇÃO =====
Valor Entrada "Estado_da_Rede" = 20,00
    Pertinência "BB" = 0,00
    Pertinência "BI" = 0,00
    Pertinência "BS" = 1,00
    Pertinência "BC" = 0,00

Valor Entrada "LBR" = 80,00
    Pertinência "Baixo" = 0,80
    Pertinência "Medio" = 0,20
    Pertinência "Alto" = 0,00

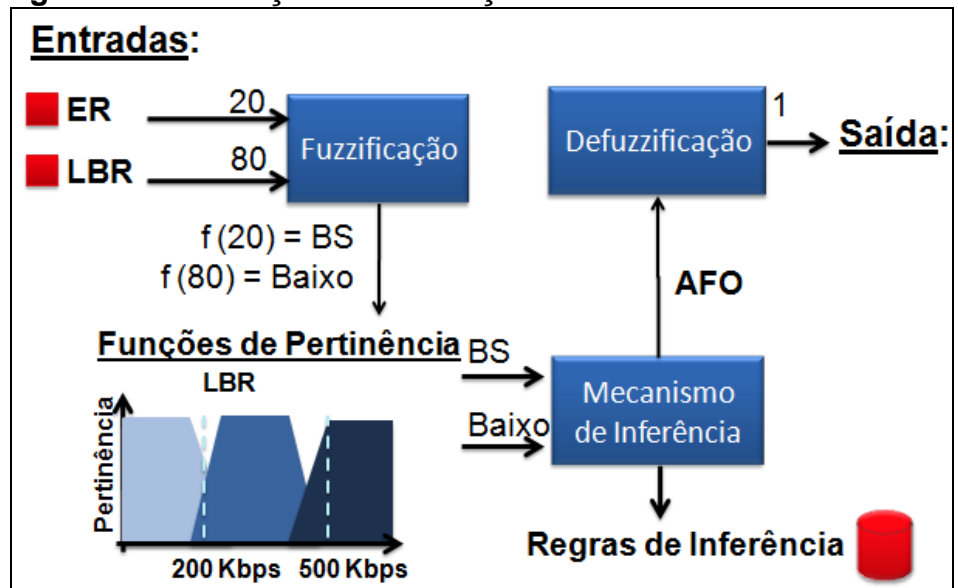
Valor Saída "SAIDA"
    Pertinência "AFO" = 0,80
    Pertinência "AFR" = 0,20
    Pertinência "BLO" = 0,00

===== DEFUZIFICAÇÃO =====
Defuzificação pelo método do Centro da Gravidade
  
```

Fonte: Elaborado pelo autor

Nesta simulação, foram atribuídos os valores numéricos de entrada para as variáveis Estado da Rede (atribuído o valor 20) e LBR (atribuído o valor 80). Após o processo de fuzzificação foi associado o valor de pertinência 1 para o rótulo BS da variável linguística Estado da Rede e foi associado o valor de pertinência 0.8 para o rótulo Baixo e 0.2 para o rótulo Médio da variável linguística LBR. Neste caso, podemos dizer que o valor desta variável é mais Baixo do que Médio. Após aplicação das regras de inferência e do processo de defuzzificação pelo método Centro de Gravidade, foi gerado o valor numérico 1 que aponta para a aceitação forte (AFO). Os passos desta simulação estão conforme a ilustração (Figura 21).

Figura 21 - Ilustração da simulação dos resultados da LN



Fonte: Elaborado pelo autor

5 SIMULAÇÕES E ANÁLISES DO CAC-RDFF

5.1 Cenário das simulações

Para realizar simulações e testes, de modo a comprovar a eficiência da solução proposta, foi utilizado o módulo E-UMTS por Antoniou (2004), desenvolvido no simulador *Network Simulator – Versão 2 (NS-2)*, conforme mencionado anteriormente. Com base em resultados de simulação, foram utilizados métodos estatísticos para comprovar a eficiência da solução proposta.

Todas as simulações realizadas neste trabalho seguiram os mesmos parâmetros, variando-se apenas o tempo de simulação e o número de usuários. O módulo E-UMTS possui três cenários implementados:

- a) Ambiente de escritório;
- b) Centro de negócios na cidade;
- c) Ambiente urbano.

Para este trabalho, foi escolhido o cenário urbano, cujos parâmetros são mostrados na Tabela 1. A máquina utilizada para as simulações foi Intel Dual Core E5700 CPU de 3 GHz e 4GB de memória. Devido à esta limitação computacional, a topologia utilizada possui apenas três células, sendo que cada célula possui antenas tri-setoriais. Deste modo, têm-se nove setores em três células, permitindo um melhor alcance de sinal.

Tabela 1 - Cenário das simulações

Parâmetros	
Modelo de propagação de rádio	<i>Hata - COST 231</i>
Probabilidade do usuário ser ativo	<i>Busy Hour Call Attempts (BHCA)</i>
Ambiente de operação	<i>Outdoor</i>
Modelo de mobilidade	<i>Gauss-Markov</i>
Velocidade da mobilidade	50 km/h (13,89 m/s) ou aleatória
Topologia	1 célula → BSs trissetoriais (120°/setor)
<i>Conversational</i> (12,2 kbps)	42% dos usuários
<i>Streaming</i> (768 kbps)	16% dos usuários
<i>Interactive</i> (384 kbps)	18,50% dos usuários
<i>Background</i> (144 kbps)	23,50% dos usuários

Fonte: Dados da pesquisa

Pela Tabela 1, constata-se que o modelo de propagação de rádio do cenário simulado é o *Hata*, que utiliza a frequência, a distância, a cobertura da BS e do usuário para estimar a perda de propagação de rádio. O modelo utilizado para calcular a probabilidade de o usuário estar ativo é o *Busy Hour Call Attempts*. A probabilidade de um usuário ficar ativo baseia-se no registro de tentativas de chamadas telefônicas do projeto SEACORN. O percentual de usuários ativos varia de 11% a 15% na simulação (ANTONIOU, 2004). O modelo de mobilidade utilizado foi o *Gauss-Markov*, que define a mobilidade para cada usuário em uma determinada área. Este modelo *Gauss-Markov* foi adotado por causa do tamanho da célula (ambiente macro-celular), gerando movimento dinâmico dos nodos na rede. A velocidade de mobilidade dos usuários no modelo *Gauss-Markov* é de 50 km/h, equivalente a 13,89 m/s, ou aleatória. A topologia possui BSs trissetoriais, porque a soma desta cobertura individual de 120° atende a mesma cobertura de 360° de uma *omni* com uma maior abrangência (ANTONIOU, 2004).

A divisão de tráfego não foi alterada no CAC-RDFF, logo, a taxa de utilização de cada classe foi definida como: 42% para os 12,2 kbps de *conversational*, 16% para os 768 kbps de *streaming*, 18,50% para os 384 kbps de *interactive* e 23,50% para os 144 kbps de *background* (única classe NRT) (STORCK, 2007). Para representar cada classe, foram selecionadas as aplicações mais críticas de cada classe, a saber:

- a) Conversa de voz para a classe *Conversational* (RT);
- b) Vídeo *one-way* para *Streaming* (RT);
- c) *Web-browsing* para *Interactive* (RT);
- d) Serviços SMS para *Background* (NRT).

Com o propósito de comprovar a melhoria no processo de SHO, foram feitas comparações com o CAC-RDF. Foram realizadas simulações com o CAC-RDF e CAC-RDFF com o tempo de 200 segundos, onde o cenário foi composto por 210 usuários com mobilidade entre as 3 BSs, a distância é de 800 m entre as BSs.

5.2 Mapa das simulações

O método estatístico adotado para tabular o conjunto de resultados das simulações, foi a média truncada. Este método descarta os extremos máximos e mínimos. Em qualquer medição pode acontecer erros humanos ou de equipamentos, o que podem gerar resultados muito fora das médias esperadas. Para minimizar a participação destes pontos extremos na média, descartam-se os valores máximos e mínimos, normalmente entre 5 e 20 % das amostras. Desta forma conseguimos uma média com menor desvio. Nas tabelas abaixo podem ser verificados os mapas dos resultados das simulações por classe de serviços:

Tabela 2 - Mapa dos resultados das simulações: *Conversational*

Conversational - CAC-RDF											
T(s)	Média(block)	Simula1	Simula2	Simula3	Simula4	Simula5	Simula6	Simula7	Simula8	Simula9	Simula10
20	13	12	13	14	11	14	12	14	13	12	15
40	30	31	30	27	31	33	31	30	27	30	30
60	32	33	34	29	31	33	32	30	31	34	33
80	35	34	35	33	34	37	35	34	37	35	36
100	37	38	36	35	38	39	35	36	39	38	36
120	39	40	41	38	39	40	37	38	39	40	38
140	41	41	45	39	40	43	41	39	40	42	40
160	44	43	46	44	42	45	45	43	42	45	45
180	46	45	48	45	46	47	47	44	46	47	45
200	47	48	49	45	46	48	47	45	46	49	47

Conversational - CAC-RDFF											
T(s)	Média(block)	Simula1	Simula2	Simula3	Simula4	Simula5	Simula6	Simula7	Simula8	Simula9	Simula10
20	13	12	14	15	13	11	14	12	13	12	14
40	27	26	28	27	26	25	30	28	27	26	27
60	29	31	29	30	27	28	32	28	29	27	29
80	30	32	29	33	28	29	32	30	30	28	29
100	34	35	36	35	33	33	35	34	33	32	34
120	36	35	37	38	35	36	37	37	34	35	36
140	38	38	39	40	37	36	38	37	36	40	39
160	39	39	40	41	39	37	39	38	37	41	39
180	41	39	42	43	40	40	42	43	39	41	41
200	43	42	43	46	43	43	44	43	40	44	42

Fonte: Dados da pesquisa

Tabela 3 - Mapa dos resultados das simulações: Streaming

Streaming - CAC-RDF											
T(s)	Média(block)	Simula1	Simula2	Simula3	Simula4	Simula5	Simula6	Simula7	Simula8	Simula9	Simula10
20	16	16	14	17	16	16	17	16	15	16	17
40	43	42	44	43	41	45	43	41	44	45	42
60	47	46	47	46	45	48	47	47	48	49	47
80	50	48	49	48	50	53	51	50	52	50	49
100	53	52	54	53	51	55	53	53	54	53	52
120	56	57	57	56	54	58	55	58	55	55	55
140	59	61	57	57	59	60	59	62	59	57	59
160	63	63	61	62	64	65	62	65	63	62	63
180	66	64	66	68	66	68	65	68	65	64	66
200	69	68	71	70	69	70	68	70	68	67	69

Streaming - CAC-RDF											
T(s)	Média(block)	Simula1	Simula2	Simula3	Simula4	Simula5	Simula6	Simula7	Simula8	Simula9	Simula10
20	18	19	17	17	19	18	17	19	18	20	16
40	42	41	44	42	41	43	42	41	43	41	42
60	44	43	45	43	44	46	45	42	43	45	44
80	46	47	47	46	47	46	45	46	44	45	47
100	48	47	47	49	50	49	47	49	46	47	49
120	49	48	50	51	51	49	48	49	47	48	49
140	52	53	52	54	51	52	50	52	52	50	54
160	55	57	53	56	54	55	53	55	56	54	57
180	57	59	55	57	56	58	57	58	57	55	58
200	58	59	57	57	56	58	58	61	59	57	58

Fonte: Dados da pesquisa

Tabela 4 - Mapa dos resultados das simulações: *Interactive*

Interactive - CAC-RDF											
T(s)	Média(block)	Simula1	Simula2	Simula3	Simula4	Simula5	Simula6	Simula7	Simula8	Simula9	Simula10
20	20	19	20	20	18	22	20	19	21	23	18
40	45	42	46	47	45	44	45	46	44	45	46
60	47	46	49	48	46	48	47	46	44	48	48
80	51	53	51	49	50	52	52	52	49	49	53
100	55	56	54	55	57	54	55	55	52	55	57
120	61	62	61	63	62	61	60	59	61	60	61
140	63	64	65	63	62	64	61	62	63	62	64
160	69	70	69	68	72	68	69	69	71	68	66
180	72	73	71	73	75	70	70	71	72	73	72
200	75	76	75	73	77	74	76	77	72	74	76

Interactive - CAC-RDF											
T(s)	Média(block)	Simula1	Simula2	Simula3	Simula4	Simula5	Simula6	Simula7	Simula8	Simula9	Simula10
20	20	21	19	20	23	17	20	20	19	20	21
40	42	40	41	44	42	43	42	41	43	42	42
60	44	44	45	44	43	46	43	41	44	46	44
80	46	47	48	47	44	46	45	44	47	47	45
100	48	47	49	48	46	47	48	46	50	51	48
120	52	51	52	50	49	53	52	52	55	54	52
140	54	55	55	53	54	53	54	52	57	54	53
160	58	59	58	57	55	57	60	57	59	60	58
180	61	61	60	62	60	61	64	59	60	62	61
200	64	65	64	62	63	64	66	61	66	65	64

Fonte: Dados da pesquisa

Tabela 5 - Mapa dos resultados das simulações: *Background*

Background - CAC-RDF											
T(s)	Média(block)	Simula1	Simula2	Simula3	Simula4	Simula5	Simula6	Simula7	Simula8	Simula9	Simula10
20	5	6	6	5	4	5	6	3	5	4	6
40	7	7	6	9	4	7	7	7	8	9	6
60	9	11	9	10	7	9	8	9	10	9	8
80	11	12	11	10	8	12	12	13	10	11	11
100	14	12	14	13	12	17	16	14	13	14	15
120	16	17	15	15	16	19	16	15	14	16	17
140	18	18	20	16	17	19	20	17	16	19	18
160	20	21	22	19	20	19	20	19	19	20	21
180	24	23	25	26	24	24	23	23	24	22	26
200	27	26	26	27	29	28	27	25	27	28	27

Background - CAC-RDFF											
T(s)	Média(block)	Simula1	Simula2	Simula3	Simula4	Simula5	Simula6	Simula7	Simula8	Simula9	Simula10
20	3	3	4	3	2	3	1	5	4	1	4
40	7	6	8	8	7	6	6	8	8	5	8
60	7	6	8	8	7	6	6	8	8	5	8
80	9	6	11	10	11	9	7	9	9	10	8
100	10	10	11	10	11	9	8	10	11	10	10
120	12	12	12	13	12	9	13	11	13	11	14
140	14	13	15	14	14	12	15	13	15	13	16
160	15	15	15	16	15	14	15	15	15	14	16
180	17	18	17	17	16	16	18	17	17	16	18
200	18	19	18	18	17	16	19	18	18	18	19

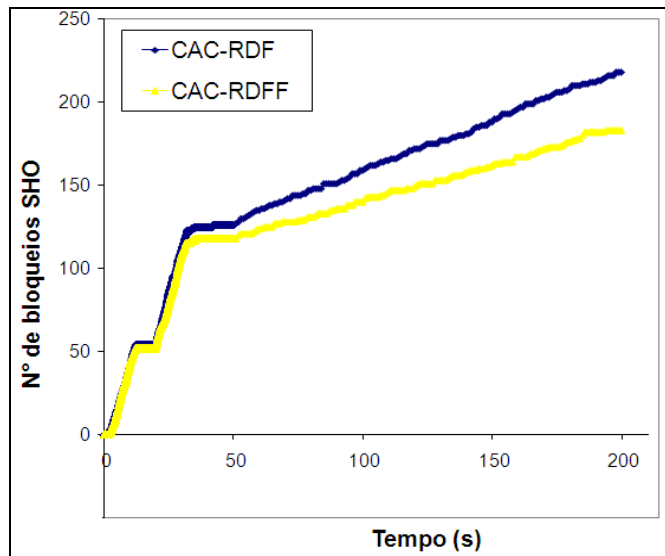
Fonte: Dados da pesquisa

5.3 Resultados de bloqueios de SHO

O Gráfico 1 apresenta os resultados de bloqueios de *handover* para o CAC-RDF e para o CAC-RDFF. Neste gráfico, também estão incluídas as quatro classes de serviços: *Conversational*, *Streaming*, *Interactive* e *Background*.

Os resultados mostram que o CAC-RDFF é uma evolução do CAC-RDF, diminuindo os bloqueios de chamados do tipo *handover*.

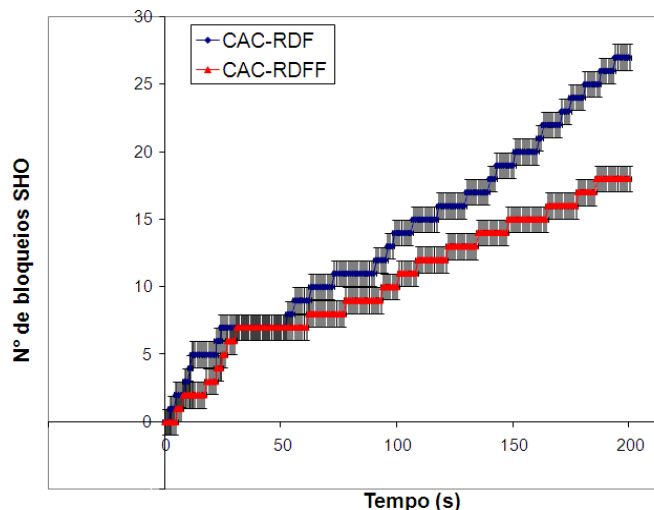
Gráfico 2 - Análise de bloqueios SHO entre CAC-RDF x CAC-RDFF



Fonte: Elaborado pelo autor

Pelo Gráfico 2 fica evidente que o CAC-RDFF bloqueia menos *handovers* que o CAC-RDF. Seu desempenho será sempre igual ou melhor que o CAC-RDF em termos de aceitação SHO. O ganho obtido é resultado do segundo mecanismo de LN, uma vez que o CAC-RDFF somente aciona este segundo mecanismo de LN quando a requisição de chamada for SHO bloqueada no primeiro mecanismo de LN.

Gráfico 3 - Análise serviço *Background*



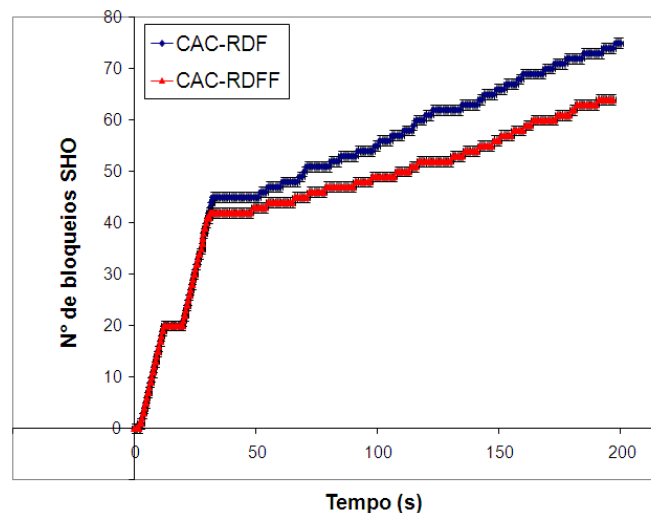
Fonte: Elaborado pelo autor

Este exemplo (Gráfico 3), se enquadra numa distribuição normal, onde é traçada a curva do valor medido (eixo X) pelo número de ocorrências (eixo y). Com a área sob a curva, pode-se obter a probabilidade de ocorrência. A área pode ser

obtida através da integração da função que deu origem à curva ($f(x)$ = Gráfico 2). Como a função é muito complexa e só pode ser integrada através de software específico, normalmente usamos os valores tabelados para determinar o intervalo de confiança. Para uma vez o desvio padrão, o intervalo de confiança segundo a tabela padronizada é de aproximadamente 68,27%, para duas vezes o desvio padrão, o intervalo de confiança é de aproximadamente 95,45%. Resumindo-se, foi utilizado o intervalo de confiança 2 sigmas (2 vezes o desvio padrão) a probabilidade de uma amostra aleatória estar contida no intervalo Média \pm 2 desvios, é de 95,45%.

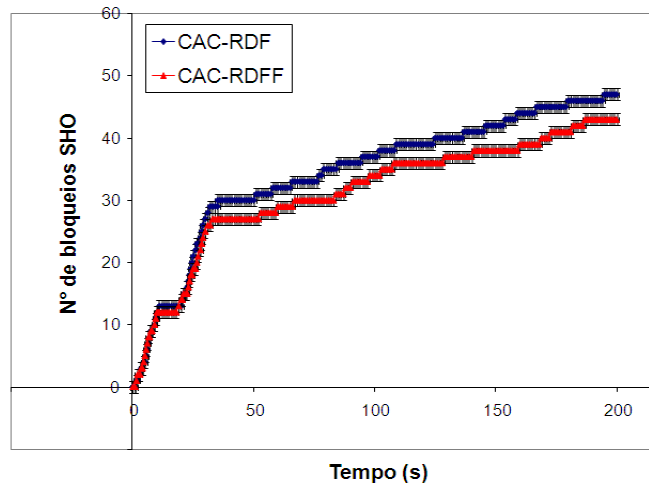
Conforme apresentado (Gráfico 3), o CAC-RDFF foi mais eficiente em relação aos números de bloqueios durante todo o período da simulação no serviço *Background*, ainda que o *Background* seja a classe de serviço menos prioritária, o CAC-RDFF foi mais eficiente nas simulações.

Gráfico 4 - Análise serviço *Interactive*



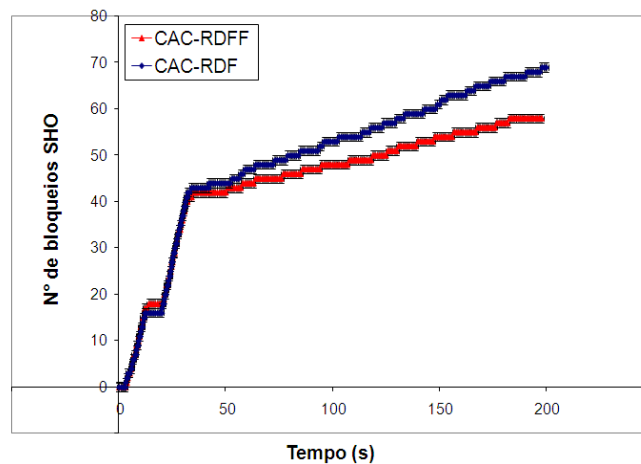
Fonte: Elaborado pelo autor

Conforme mostra (Gráfico 4), para a classe de serviço *Interactive* o CAC-RDFF demonstrou vantagem em relação ao CAC-RDF. O limite inferior do CAC-RDF, calculado pelo intervalo de confiança, não coincide com o limite superior do CAC-RDFF a partir do tempo 60 segundos. Com isto, podemos afirmar que o CAC-RDFF é mais eficiente que o CAC-RDF no serviço *Interactive*, após determinado intervalo de tempo.

Gráfico 5 - Análise serviço *Conversational*

Fonte: Elaborado pelo autor

Para a classe de serviço mais prioritária, que é a *Conversational*, o CAC-RDFF demonstrou visivelmente sua eficiência em relação ao CAC-RDF. O limite inferior do CAC-RDF calculado pelo Intervalo de Confiança não coincide com o limite superior do CAC-RDFF após 40 s de simulação.

Gráfico 6 - Análise serviço *Streaming*

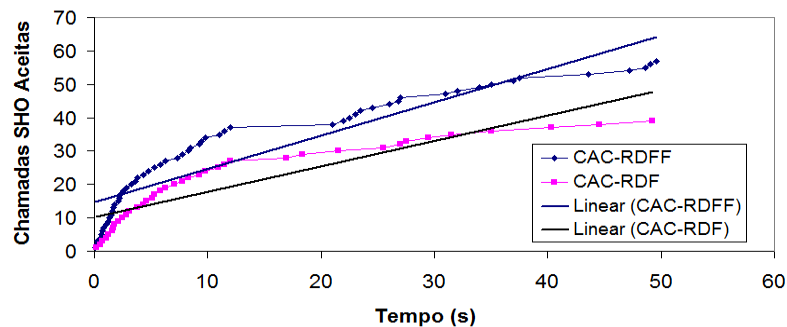
Fonte: Elaborado pelo autor

Por fim, para a classe de serviço *Streaming*, o CAC-RDFF também apresentou um desempenho superior ao CAC-RDF e fechou a simulação com vantagem em termos de bloqueios.

5.4 Resultados de aceites de SHO

Outra análise realizada foi quanto ao número de aceitação de chamada *handover* obtidas pelo CAC-RDF e CAC-RDFF. As simulações para estas análises de aceitação, foram realizadas por 50 segundos. O cenário foi composto por 210 usuários com mobilidade entre 3 BSs. A opção de simular em um curto período de tempo (50 s) na geração dos gráficos de aceitação, se justifica quando observamos as simulações e gráficos de bloqueio já apresentados. Ao analisar estes gráficos, percebemos que, quanto maior o período de tempo, maior é a diferença positiva em favor da solução proposta neste trabalho (CAC-RDFF), por isso, é importante a análise em um curto período de tempo. Os gráficos abaixo representam os momentos de aceitação das classes em questão.

Gráfico 7 - Análise de aceites SHO entre CAC-RDF x CAC-RDFF – A



Fonte: Elaborado pelo autor

O gráfico (Gráfico 7) mostra o desempenho de aceitação de *handover* para o CAC-RDF e o CAC-RDFF nas quatro classes de serviço (*Background*, *Interactive*, *Conversational* e *Streaming*). Percebe-se que o CAC-RDFF apresentou um resultado melhor que o CAC-RDF durante todo o período da simulação. Para determinação da equação da reta média (CAC-RDF e CAC-RDFF), foi utilizado o método de regressão linear. Assim, é possível estimar o número médio de aceitação de chamadas *handover* para qualquer instante de tempo. A estimativa abaixo se refere ao número de aceitação de chamadas *handover* no milésimo segundo (10 horas). Conforme mostra a equação abaixo, para o instante de tempo $T=1000s$, o CAC-RDF aceitará apenas 770 chamadas *handover* contra 1012 pelo CAC-RDFF. Esta vantagem do CAC-RDFF aumenta ao longo do tempo.

CAC – RDF

$$Y = 0,7601 T + 10,292$$

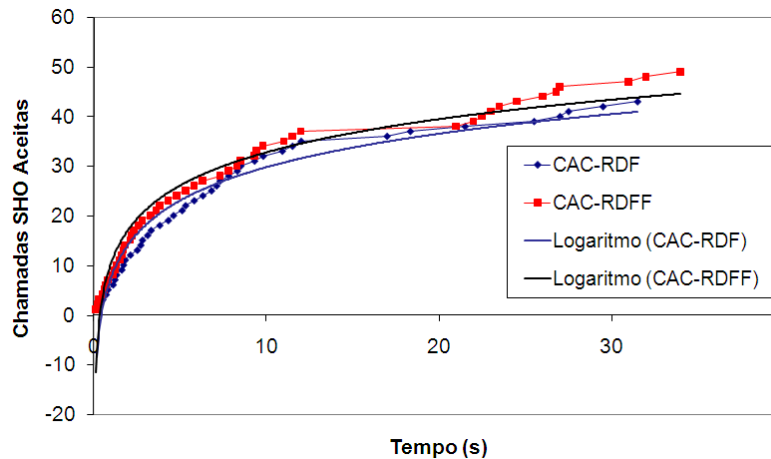
$$Y = 770$$

CAC-RDFF

$$Y = 0,9978 T + 14,673$$

$$Y = 1012$$

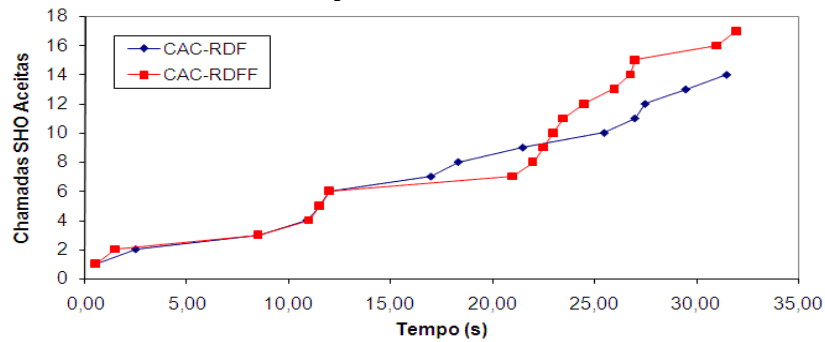
Gráfico 8 - Análise de aceites SHO entre CAC-RDF x CAC-RDFF – B



Fonte: Elaborado pelo autor

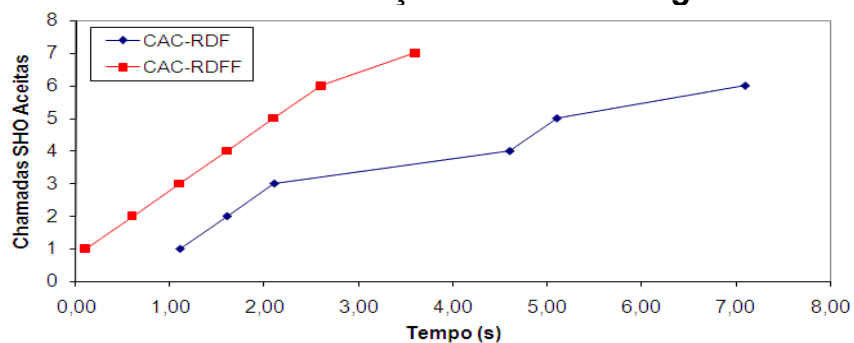
O índice de correlação R^2 é o grau de relação de x com y . Conforme mostra (Gráfico 8), também foi aplicado o modelo de regressão logarítmica, para os dados apresentados na análise do desempenho de aceitação de *handover* para o CAC-RDF e para o CAC-RDFF nas quatro classes de serviço (*Background*, *Interactive*, *Conversational* e *Streaming*).

Através das simulações e comprovações matemáticas, foi comprovada a vantagem do CAC-RDFF sobre o CAC-RDF, em relação ao melhor tratamento de chamadas do tipo *handover*. Logo, o CAC-RDFF resolve o problema de bloqueio de chamadas do tipo *handover* do CAC-RDF. Isso se deve à inclusão do novo mecanismo de LN como arquitetura em cascata, que permite a reavaliação de bloqueios de aplicações prioritárias na rede.

Gráfico 9 - Aceitação SHO Conversational

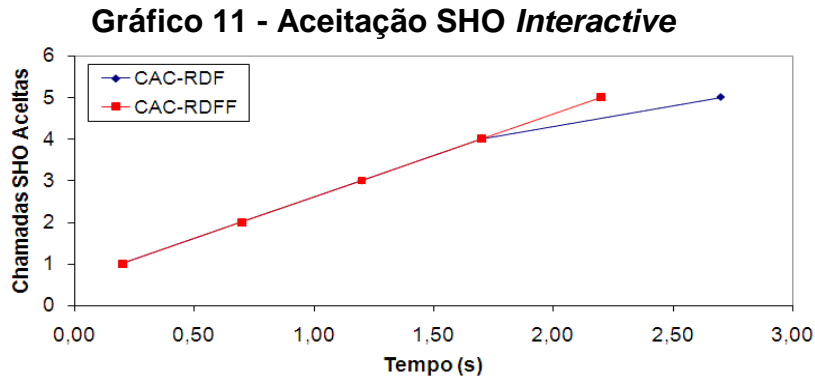
Fonte: Elaborado pelo autor

O CAC-RDFF demonstra uma desvantagem em relação ao CAC-RDF durante um curto intervalo de tempo (Gráfico 9). Em tese, o CAC-RDFF não poderia ter um comportamento inferior em relação ao CAC-RDF em nenhum momento, pois o segundo mecanismo de LN só é acionado se houver bloqueio do primeiro mecanismo de LN e se a chamada for do tipo *handover*. Mas, levando em consideração o cenário da simulação, este comportamento é justificado pela divergência natural ao simular a mesma solução (por exemplo, CAC-RDF) em dois momentos distintos. Logo, o fato da superioridade do CAC-RDFF ter se estabelecido em crescimento constante a partir do 23º segundo, evidencia a eficiência da nova solução para a classe de serviços *Conversational*.

Gráfico 10 - Aceitação SHO Streaming

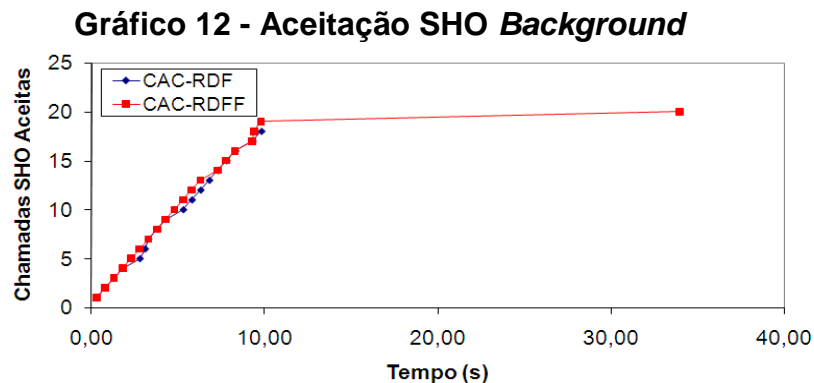
Fonte: Elaborado pelo autor

O Gráfico 10 mostra os resultados de aceitação de *handover* da classe *Streaming*. Os resultados comprovam a vantagem do CAC-RDFF em relação ao CAC-RDF para aceitação desse tipo de chamadas.



Fonte: Elaborado pelo autor

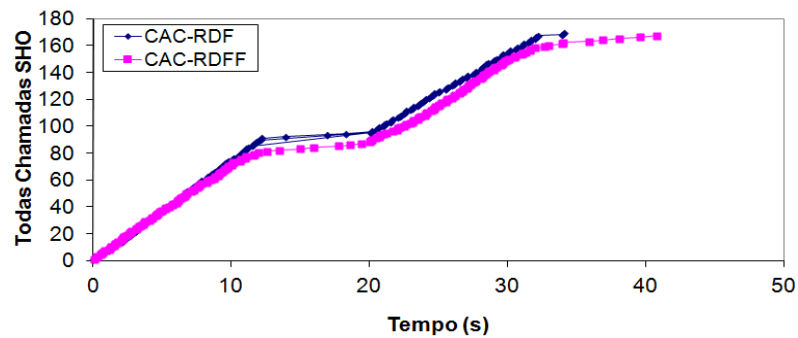
Os resultados de aceitação para *handover* da classe *Interactive* são apresentados pelo gráfico (Gráfico 11). Nesse quesito, o desempenho do CAC-RDFF equivale ao comportamento do CAC-RDF. Isto é justificado pelo fato da classe *Interactive* ser menos prioritária. Com isso, os recursos foram atribuídos principalmente para *Conversational* e *Streaming*.



Fonte: Elaborado pelo autor

O mesmo comportamento de *handover Interactive* é obtido para a aceitação de *handover Background*, que é a classe menos prioritária. Esse resultado é apresentado no Gráfico 12. É apresentada uma ligeira vantagem do CAC-RDFF em relação ao CAC-RDF.

Gráfico 13 - Total de chamadas SHO (Bloqueadas e Aceitas)



Fonte: Elaborado pelo autor

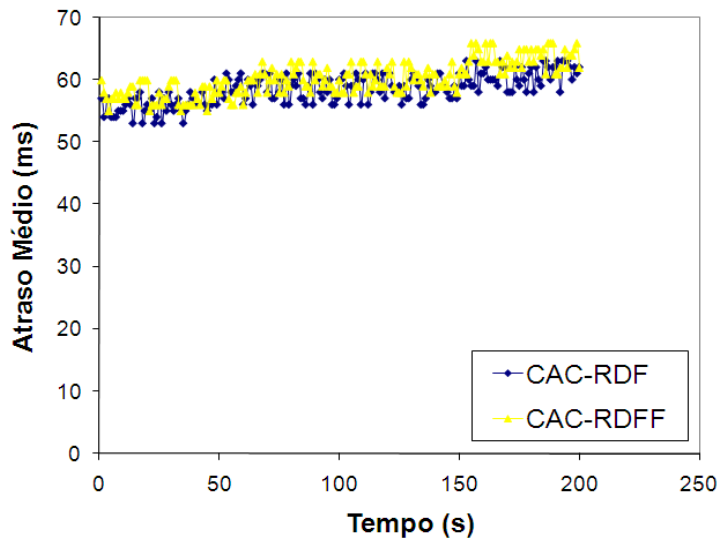
Conforme demonstrado acima (Gráfico 13) compara o CAC-RDF com o CAC-RDFF, em relação à média do total de chamadas SHO. Por simulação, o CAC-RDF bloqueou em média 126 chamadas SHO contra 118 do CAC-RDFF, em termos de aceitação foram 43 chamadas SHO para o CAC-RDF contra 49 chamadas para o CAC-RDFF.

De acordo com os dados apresentados, chegamos a uma média total de 169 chamadas SHO do CAC-RDF contra 167 chamadas SHO do CAC-RDFF.

Em simulações, informações dos pacotes trafegados na rede são armazenadas em arquivos chamados *trace*. Esses arquivos podem ser utilizados para avaliar a QoS da rede e das BSs. Este trabalho utilizou um analisador de *trace* por BS e por classe. Seu objetivo foi agrupar os serviços com mesma sensibilidade de QoS por região, possibilitando uma melhor análise da rede. As informações dos parâmetros de QoS foram armazenadas em tabelas de dados MySQL, possibilitando uma melhor manipulação dos dados através de Linguagem de Consulta Estruturada (SQL).

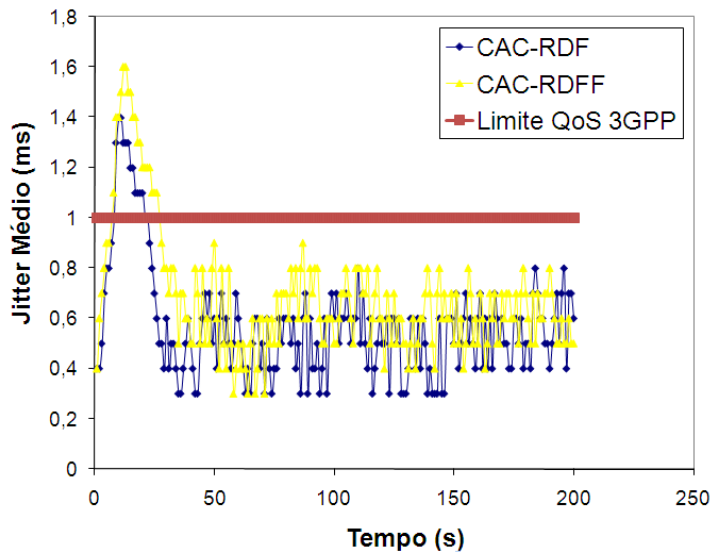
5.5 Resultados de QoS

Para avaliar o desempenho da solução proposta, foram analisados os seguintes parâmetros de QoS: vazão, jitter e atraso. Todas as classes de serviços nas simulações dos CAC-RDFF tiveram vazão acima do limite mínimo especificado pelo 3GPP.

Gráfico 14 - QoS – Atraso Conversational

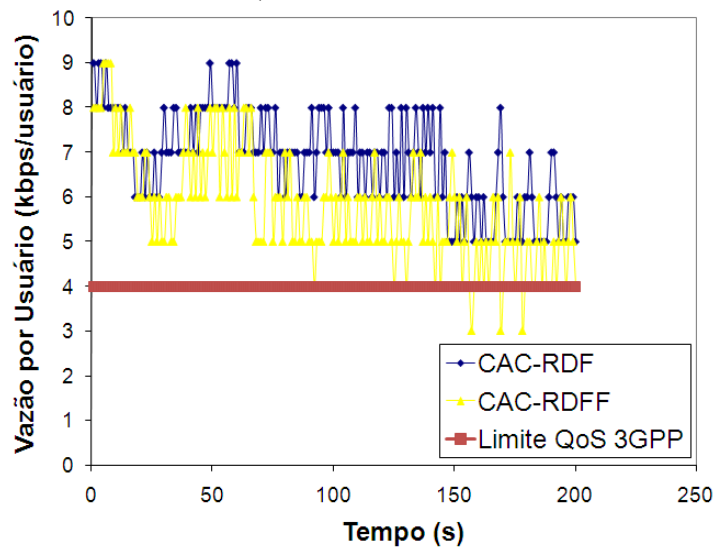
Fonte: Elaborado pelo autor

O Gráfico 14 apresenta o parâmetro de QoS atraso médio pelo tempo de simulação, observa-se que o atraso ficou dentro do limite estabelecido pelo 3GPP (<400s).

Gráfico 15 - QoS – Jitter Conversational

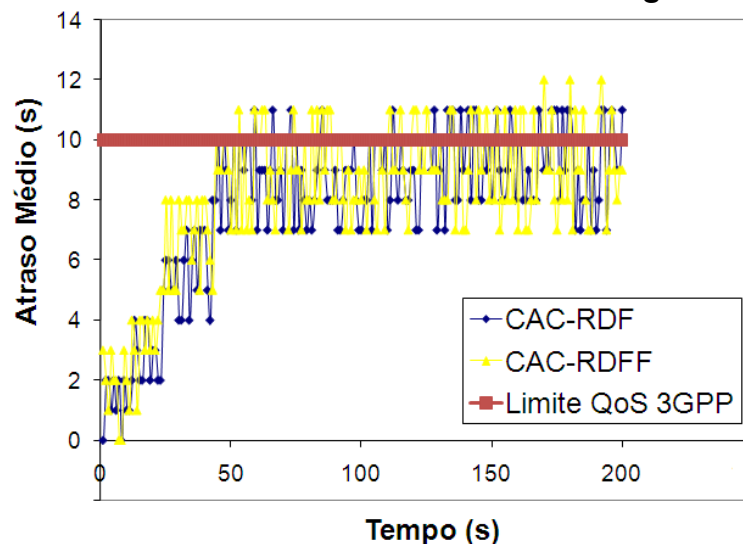
Fonte: Elaborado pelo autor

Conforme Gráfico 15, a partir de 35 s, os dois CACs mantiveram jitter aceitável para a classe de serviço *Conversational*.

Gráfico 16 - QoS – Vazão Conversational

Fonte: Elaborado pelo autor

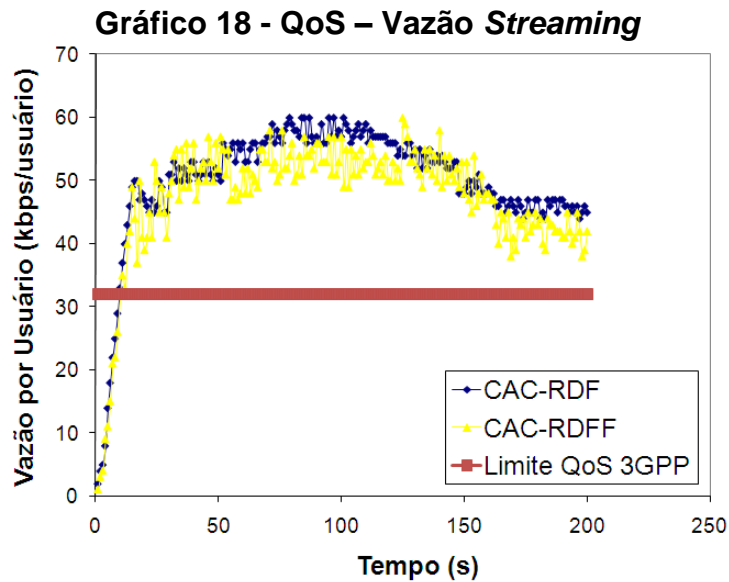
Com relação à vazão por usuário, o limite foi respeitado em ambas as simulações, ocorrendo somente três momentos em que o CAC-RDFF comprometeu este limite. Isto se justifica pela saída Aceitação Fraca (AFR) do segundo mecanismo de LN do CAC-RDFF, onde não é garantida a QoS da chamada.

Gráfico 17 - QoS – Atraso Streaming

Fonte: Elaborado pelo autor

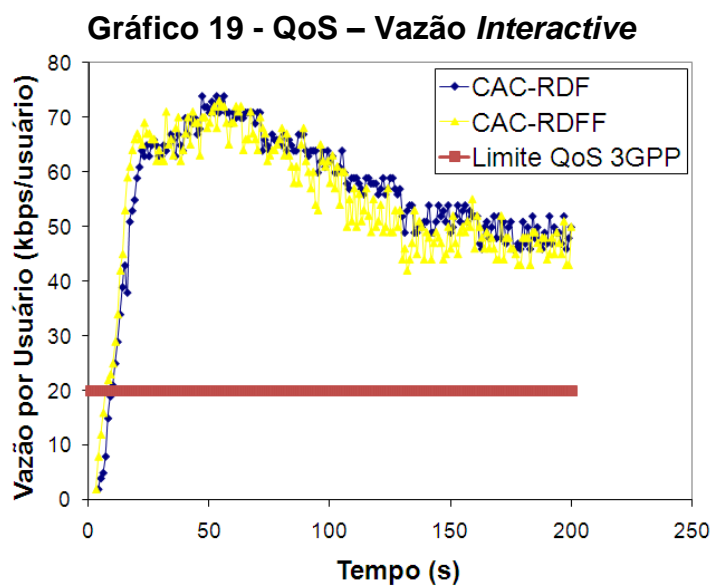
Para o parâmetro atraso médio da classe de serviço Streaming, este indicador aumentou até atingir e ultrapassar por vezes o limite de 10 s. O CAC-RDF garantiu QoS para as chamadas por mais tempo, mas os dois CACs ficaram perto do limite a partir dos 60 s de simulação. Após atingir este limite de 10 s, a tendência do atraso,

caso nenhuma chamada fosse terminada, seria de aumentar devido ao escalonamento de pacotes (isto é, o pacote fica cada vez mais tempo na fila).



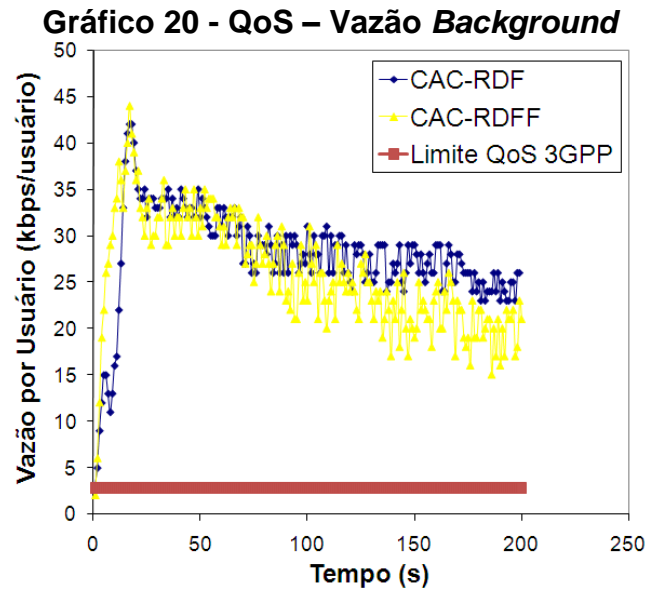
Fonte: Elaborado pelo autor

Com relação à vazão por usuário da classe de serviço *Streaming*, o limite foi respeitado em ambas as simulações. O pior rendimento do CAC-RDFF (quando comparado ao CAC-RDF) durante quase toda a simulação é devido a saída Aceitação Fraca (AFR) do segundo mecanismo de LN do CAC-RDFF, onde não é garantido a QoS da chamada.



Fonte: Elaborado pelo autor

Assim como ocorreu com vazão por usuário da classe de serviço *Streaming*, o limite foi respeitado em ambas as simulações, mas, houve um pior rendimento do CAC-RDFF durante quase toda a simulação para a classe de serviço *Interactive*.



.Fonte: Elaborado pelo autor

O gráfico acima mostra a vazão por usuário para *Background*, por tempo de simulação, o CAC-RDF apresentou maior vazão após 100 s de simulação, mas ambos respeitaram o limite mínimo (2,8 kbps) definido pelo 3GPP

Enfim, através destas simulações e comprovações matemáticas, o CAC-RDFF resolve o problema proposto para um melhor tratamento nos processos de *handover* em redes UMTS, utilizando Lógica Nebulosa em cascata, sobre o CAC-RDF, reavaliando bloqueios, caso sejam *handovers*. Mesmo aceitando mais, o CAC-RDFF não compromete a QoS das chamadas já em andamento, conforme foi mostrado.

6 CONCLUSÕES E TRABALHOS FUTUROS

Diversos controles de admissão de chamadas são propostos na literatura para garantir QoS em redes de celulares. Um exemplo é o CAC-RDF, que apresenta uma estratégia de bloqueios dinâmicos de chamadas menos prioritárias com Lógica Nebulosa. Essa técnica transforma o CAC-RD, seu antecessor, quanto ao seu caráter estático de bloqueios de chamadas, que pode causar desperdício de recursos. Contudo, o CAC-RDF deixa em aberto o alto índice de bloqueios para chamadas do tipo *handover*, que foi o foco deste trabalho.

Este trabalho propôs uma estrutura de Lógica Nebulosa adicional ao CAC-RDF. A primeira estrutura LN especifica dinamicamente limites de aceitação de chamadas prioritárias sem considerar *handover*. Quando a chamada é bloqueada nesse primeiro nível, a requisição é reavaliada em uma segunda estrutura LN. O termo “cascata” é utilizado devido ao reaproveitamento da variável de saída da primeira estrutura LN. Vale lembrar que, se a modelagem da Lógica Nebulosa não for bem feita, o mecanismo LN pode não funcionar conforme o desejado, produzindo um desempenho até pior que o esperado. Por isso, a implementação deste trabalho seguiu uma metodologia de 7 passos para a representação desta segunda camada de Lógica Nebulosa.

Para analisar o CAC-RD *Fuzzy* em Cascata (CAC-RDFF), simulações foram realizadas com os controles CAC-RDF e CAC-RDFF. Os resultados mostraram que o CAC-RDFF apresentou melhores resultados em relação ao CAC-RDF. Em todos os cenários simulados, houve um número maior de aceitação de chamadas do tipo *handover*. Este resultado se justifica pelo fato do CAC-RDFF acionar a segunda estrutura de Lógica Nebulosa somente quando a requisição de chamada for *handover* e bloqueada na primeira estrutura de Lógica Nebulosa. Esta arquitetura permite inferir que o CAC-RDFF é mais eficiente em relação ao CAC-RDF em uma proporção diretamente relacionada ao número de requisições de *handovers* não aceitos na primeira estrutura de Lógica Nebulosa. O melhor desempenho do CAC-RDFF se deve à reavaliação do bloqueio de *handover* de acordo com o estado da rede. Por outro lado, caso todas as solicitações *handover* sejam aceitas na primeira estrutura de LN, o desempenho do CAC-RDFF é equivalente ao desempenho do CAC-RDF. Resumidamente, o CAC-RDFF aumenta a disponibilidade da rede para

serviços previamente estabelecidos e em processo de *handover*, sem prejudicar o desempenho das aplicações em curso.

Como trabalhos futuros, pretende-se comparar o CAC-RDFF com outros CACs de outros grupos de pesquisa, simular cenários maiores com o CAC-RDFF, adaptar o CAC-RDFF para redes heterogêneas integradas, implementar o CAC-RDFF em simulações de outras redes, tais como LTE (*Long Term Evolution*) ou WiMax.

REFERÊNCIAS

- ANTONIOU, J. **A system level simulator for enhanced UMTS coverage and capacity planning**. 2004. Thesis (Doctorate) - University of Cyprus, Nicosia, Department of Computer Science.
- ENGELBRECHT, A. **Computational intelligence: an introduction**. New York: Halsted Press, 2002.
- FIGUEIREDO, Fátima de Lima Procópio Duarte. **DIFFMOBIL: uma arquitetura de qualidade de serviço fim-a-fim em redes GPRS**. 2004. 111 f. Tese (Doutorado) - Universidade Federal de Minas Gerais.
- FIGUEIREDO, L.C. et al. **Ambiente integrado para análise e desenvolvimento de controladores difusos**. Rio Claro / SP: UNESP, 1993. p. 291-299.
- GENERATION partnership project end-to-end Quality of Service (QoS) concept and architecture. TS-23.207. 3rd, 2011a. Disponível em: <http://www.3gpp.org/ftp/tsg_sa/TSG_SA/TSGS_24/Docs/PDF/SP-040317.pdf>. Acesso em: 20 out. 2011.
- GENERATION partnership project: services and service capabilities. TS-22.105. 3rd, 2011b. Disponível em: <[http://www.arib.or.jp/english/html/overview/doc/ STD-T63v9_20/5_Appendix/Rel8/22/22105-840.pdf](http://www.arib.or.jp/english/html/overview/doc/STD-T63v9_20/5_Appendix/Rel8/22/22105-840.pdf)>. Acesso em: 14 ago.2011.
- GHADERI, Majid; BOUTABA, Raouf. Call admission control in mobile cellular networks: a comprehensive survey. **Wireless Communications and Mobile Computing**, Canada, v. 6, n. 1, Feb., p.69-93, 2006. Disponível em: <<http://dx.doi.org/10.1002/wcm.246>> Acesso em: 11 jan. 2012.
- HALONNE, T.; ROMERO, J.; MELERO, J. **GSM, GPRS, and edge performance: evolution towards 3G/UMTS**. 2nd ed. West Sussex: John Wiley & Sons, 2003.
- JÚNIOR, I. N.; NOVAKOWSKI, S. **Lógica fuzzy sistemas de apoio a decisão**. Rio Grande do Sul, Unisinos, 2005. Disponível em: <http://www.inf.unisinos.br/~cazella/dss/SAD_Logica_Fuzzy.pdf>. Acesso em: jun. 2011.
- KUROSE, J.; ROSS, K. W. **Redes de computadores e a Internet: uma abordagem top-down**. São Paulo: Addison-Wesley, 2005.
- LINDEMANN C., L. M.; THUMMLER, A. A uni_ed approach for improving QoS and provider revenue in 3g mobile networks. **ACM Journal on Special Topics in Mobile Networks and Applications**, Texas, 2003.
- MANNER, J. **Provision of quality of service in IP-based mobile access networks**. 2003. Dissertação (Mestrado)- University of Helsinki, Department of Computer Science. Disponível em: <<http://ethesis.helsinki.fi/julkaisut/mat/tieto/vk/manner>> Acesso em: 15 jan. 2002.
- MIKUSZEWSKI, MARCIN. **Simplified UMTS network architecture**, 2009. Disponível em: <<http://mobile-telecom-networks.blogspot.com/2009/11/simplified->

umts-network-architecture.html>acesso 24/01/2012

MWANSA, D.; HANRAHAN, H. **Conceptual QoS framework for managing end-user services in the next generation network**. Johannesburg: Centre for Telecommunications Access and Services, 2004.

PEDRYCZ, Witold; VASILAKOS, Athanasios. (Ed.) **Computational Intelligence in Telecommunications Networks**. Texas, EUA: Taylor & Francis Group, 2000.

POSSELT, E. L.; FROZZA, R.; MOLZ, R. F.. **Software Infuzzy 2011**: programa de mestrado em sistemas e processos industriais PPGSPI, UNISC, 2011. Disponível em: <<http://www.unisc.br/ppgsapi>> Acesso em: 15 jan. 2012.

RAO, V. B.; RAO, H. **C++, neural networks and fuzzy logic**. 2nd ed. New York: MIS:Press, 1995.

REIS, Suéllen de Oliveira. **Avaliação de qualidade de serviço em redes WiMAX e UMTS integradas para múltiplos usuários**. 2010. 57f. Dissertação (Mestrado) - Pontifícia Universidade Católica de Minas Gerais, Programa de Pós-Graduação em Informática.

RIBEIRO, Anna Izabel João Tostes; FIGUEIREDO, Fátima de Lima Procópio Duarte; ZÁRATE, Luis E.. Controle de admissão Fuzzy baseado em limites dinâmicos de congestionamento para redes de celulares. In: SIMPÓSIO BRASILEIRO DE REDES DE COMPUTADORES E SISTEMAS DISTRIBUÍDOS, 24, 2011, Belo Horizonte, **Anais...**Belo Horizonte: 2011. Disponível em: < http://sbrc2011.facom.ufms.br/files/main/ST04_3.pdf> Acesso em: 14 mar. 2012.

RIBEIRO, Anna Izabel João Tostes. **Representações neural e fuzzy de controle de admissão de chamadas para redes UMTS**. 2010. 111 f. Dissertação (Mestrado) - Pontifícia Universidade Católica de Minas Gerais, Programa de Pós-Graduação em Informática

RIBEIRO, Anna Izabel João Tostes; STORCK, C. R.; FIGUEIREDO, Fátima de Lima Procópio Duarte. CAC-RD: an UMTS Call Admission Control . **Journal of Telecommunication Systems**, Springer, 2010.

SILVA, Rennan J. M.; COSTA, João C. W. A. **Controle de admissão de chamadas conjunto em redes heterogêneas usando aprendizado por reforço**. 2008. Disponível em:<http://www.ufpa.br/ppgcc/ppgcc/files/File/Seminario_Andamento2008/Redes/seminario_Rennan.pdf> Acesso em: 14 jan. 2012.

SIMÕES, Marcelo Godoy; SHAW, Ian S. **Controle e modelagem Fuzzy**. São Paulo: Edgard Blucher Ltda, 2007.

STASZCZAK, Lizelaine Moletta; Francisco, Antonio Carlos de; Marçal, Rui Francisco. **Lógica Fuzzy**: uma proposta na tomada de decisão. 2004. Disponível em: <http://www.pg.cefetpr.br/setor/incubadora/wpcontent/themes/1o_epege/FUZZY.pdf> Acesso em: 01/02/2012

STEINMETZ, R.; WOLF, L. Quality of Service: Where are We ? In: CAMPBELL, Andrew, NAHRSTEDT, Klara. (Ed.). **Building QoS into distributed systems**: IFIP TC6 WG6.1 Fifth International Workshop on Quality of Service (IWQOS '97), New York, USA: Chapman & Hall, 1997. p. 211- 221.

STORCK, Carlos Renato. **CAC-RD**: controle de admissão de chamadas para redes UMTS. 2007. xiii, 70 f. Dissertação (Mestrado) - Pontifícia Universidade Católica de Minas Gerais, Programa de Pós-Graduação em Informática

STORCK, Carlos Renato; RIBEIRO, Anna Izabel João Tostes; FIGUEIREDO, Fátima de Lima Procópio Duarte. **CAC-RD**: Controle de admissão de chamadas para redes UMTS. In: SIMPÓSIO BRASILEIRO DE REDES DE COMPUTADORES, 26, 2008, Rio de Janeiro, **Anais...** Rio de Janeiro: SBRC, 2008.

WEBER, L.; KLEIN, P. A. T. **Aplicação da lógica fuzzy em software e hardware**. Canoas: Ulbra, 2003. 110p.