Uso do Modelo E para Monitoramento de Qualidade de Voz em Tempo Real

André A. D. P. Souza, Paulo H. de A. Rodrigues

PPGI – DCC/NCE – UFRJ

andre@voip.ufrj.br, aguiar@nce.ufrj.br

Abstract. *The E-Model was designed as a quality tool for use during the planning phase of a telephony architecture. However, it has been used as a quality measuring tool for VoIP, despite the fact it has never been tested for such usage. In order to certify the model accuracy in real time scenarios, the E-Model and its extension, the Extended E-Model, were tested for compliance to recommendation ITU-T P.564, which compares the performance of objective quality models with PESQ. Compliance tests were performed using the G.711 μ-law and GSM, different ptime values and both with and without the use of VAD and PLC. The results proved the superiority of the Extended E-Model to the original model, showed that the models are more accurate when used with GSM and indicated that the E-Model was calibrated using a ptime of 20ms.*

Resumo. *O Modelo E foi concebido como uma ferramenta de qualidade para a fase de planejamento de uma arquitetura de telefonia. Contudo, tem sido utilizado como um modelo de aferição de qualidade para VoIP, apesar de nunca ter sido testado e sua precisão avaliada para tal fim. A fim de atestar a precisão do modelo quando usado em cenário de tempo real, o Modelo E e sua extensão, o Modelo E Estendido, foram submetidos aos testes de conformidade da recomendação ITU-T P.564, que compara a performance de modelos objetivos de qualidade com o PESQ. Os testes de conformidade foram realizados usando os codecs G.711 μ-law e GSM, diferentes valores de ptime e com e sem o uso de PLC e VAD. Os resultados comprovaram a superioridade do Modelo E Estendido sobre o original, mostraram que os modelos são precisos quando usado GSM e indicaram que o Modelo E foi calibrado usando ptime de 20ms.*

1. Introdução

Com VoIP cada vez mais presente nos serviços de telecomunicação, o usuário final deixa de considerar esta tecnologia apenas como uma opção barata em relação à telefonia tradicional e passa a exigir uma qualidade pelo menos equivalente. E com a Internet sendo usada como um meio multimídia convergente, é essencial o monitoramento em tempo real da qualidade dos serviços VoIP.

O Setor de Normatização das Telecomunicações (ITU-T) da União Internacional de Telecomunicações (ITU) definiu em sua recomendação P.862 [ITU-T 2001] um algoritmo altamente sofisticado para a medição da qualidade de uma chamada de voz, o PESQ. Esse algoritmo, todavia, torna-se inviável para uso em tempo real e para aferição de conversas privadas, pois precisa comparar o áudio original com o áudio degradado pelo sistema telefônico. Por outro lado, o Modelo E do ITU-T [ITU-T 2009],

originalmente concebido para a fase de planejamento de redes de telecomunicação, tem sido amplamente utilizado na aferição de qualidade de chamadas em infraestrutura VoIP [Lustosa et al. 2005]. A adoção do Modelo E se deve a sua formulação baseada exclusivamente em dados estatísticos do fluxo de mídia, sem necessidade de acesso à mídia.

Recentemente, foi publicada a recomendação P.564 [ITU-T 2007a], especificando critérios mínimos para modelos objetivos de medição de qualidade de voz para aplicações de telefonia que usam a pilha IP/UDP/RTP. Além de definir características básicas do modelo aderente, a P.564 apresenta uma metodologia de testes de conformidade baseada na comparação das medições do modelo sob teste com o PESQ em cenários com degradação variável. Considerando o amplo uso do Modelo E, é fundamental identificar, com o uso da P.564, os cenários para os quais ele pode ser uma opção válida e precisa para a monitoração da telefonia IP.

A ideia de passar o Modelo E pelo crivo da P.564 não é nova. De fato, alguns anos atrás, a comunidade da ITU-T usou os testes de conformidade da P.564 para avaliar o Modelo E [ITU-T 2007b] e mostrou os resultados em publicação interna de acesso restrito. Os testes foram feito numa configuração limitada de cenários de codificação e os resultados não foram positivos para o modelo.

Primeiramente, testes com a versão original do modelo usando os *codecs* G.711 e G.729 e sempre com o *ptime* (tempo em milissegundos da voz transportada no pacote IP) de 20ms foram realizados, sem sucesso. Em seguida, os autores do trabalho ajustaram parâmetros do modelo de uma maneira não especificada a fim de diminuir o erro quadrático médio dos testes. Ainda assim, o Modelo E não foi aprovado nos testes da P.564. Baseado nessas experiências, os autores concluem o trabalho afirmando que calibrar o Modelo E não é suficiente para que ele seja aprovado pela P.564.

O trabalho interno da ITU-T, porém, não apresenta a metodologia dos testes de forma transparente: o esquema de realização de chamadas não é apresentado e os dados mostrados não apresentam confiança estatística alguma. Esses fatos diminuem a credibilidade dos testes e fazem com que o trabalho não seja uma resposta razoável à questão da precisão do Modelo E para medir qualidade de chamadas de voz em tempo real.

Observando tais limitações, a aderência do modelo E à P.564 foi reavaliada usando cenários abrangentes, com descrição detalhada da metodologia de testes empregada. Para atingir as metas estabelecidas de confiança estatística, foram realizadas centenas de milhares de chamadas com diferentes *codecs* e *ptimes*, com uso ou não de PLC (*packet loss concealment* - recuperação de perda) e VAD (*voice activity detection* - detecção de atividade para supressão de silêncio). Os experimentos indicaram que o modelo E é apenas aderente em alguns casos e que extensões para seu aperfeiçoamento são viáveis.

O artigo está organizado da forma que se segue. A segunda seção descreve o PESQ, o Modelo E e sua extensão e biblioteca que implementa o modelo. A terceira seção apresenta a P.564, detalhando os requisitos para um modelo aderente e metodologia para testes de conformidade. A seção seguinte discute a aplicação da P.564 ao Modelo E, enquanto a quinta seção analisa os resultados dos testes de conformidade. Por fim, a sexta e última seção apresenta as conclusões e trabalhos futuros.

2. Modelos Objetivos de Medição de Qualidade

Tradicionalmente, a qualidade é medida em pesquisas subjetivas diretamente com os usuários, com o resultado apresentado como a média das opiniões dos usuários, em inglês *Mean Opinion Score* (MOS). A recomendação P.800 [ITU-T 1996] padroniza a forma subjetiva de se medir o MOS com avaliadores independentes atribuindo uma pontuação de 5 (excelente) a 1 (inaceitável) à qualidade da fala. Modelos objetivos são baseados em formulas matemáticas e procuram estimar um MOS o mais próximo possível das avaliações subjetivas. O PESQ e o Modelo E são atualmente os modelos objetivos de maior interesse e são descritos a seguir.

2.1. PESQ

O PESQ (*Perceptual Evaluation of Speech Quality*), definido na recomendação P.862 [ITU-T 2001], introduz um método objetivo que compara o áudio original com a gravação do mesmo áudio após passar pelo sistema de comunicação em teste. O algoritmo do PESQ faz inicialmente um alinhamento temporal das duas gravações, para depois, então, comparar a degradação dos sinais de áudio. Os sinais são separados em trechos chamados elocuções e a comparação de tais elocuções fornece atrasos que a rotina de alinhamento temporal repassa ao modelo perceptual. O resultado após o alinhamento temporal é o valor e o grau de confiança do atraso de cada elocução, levando em conta os períodos de silêncio. O impacto das variações de atraso dos tempos de elocução e de silêncio é computado na degradação de qualidade do áudio.

O modelo perceptual do PESQ é usado para calcular a diferença entre o sinal original e o degradado. Os sinais da voz são mapeados para o domínio de tempo-frequência com a Transformada Rápida de Fourier usando janelas de 32ms, 50% sobrepostas às janelas adjacentes. Filtros e compensações são aplicados aos sinais a fim de simular a percepção dos sinais sonoros pelo sistema auditivo humano. O modelo leva em consideração, por exemplo, que o aparelho auditivo humano possui uma resolução maior para frequências sonoras baixas que para altas. A densidade sonora do sinal de voz degradado através do tempo é calculada desconsiderando os períodos de silêncio e comparada à densidade do sinal original. A diferença pode ser tanto positiva quanto negativa. Quando é positiva, significa que algum componente foi adicionado ao sinal, como ruído, por exemplo. Caso a diferença seja negativa, partes do sinal original estão faltando.

A diferença entre as densidades sonoras gera a densidade da perturbação sonora. O PESQ leva em conta também uma possível assimetria causada pela distorção do sinal no processo de codificação e decodificação, calculando, além da densidade de perturbação sonora normal, a densidade de perturbação assimétrica. Quadros consecutivos com perturbação acima de um limite têm seus atrasos recalculados a fim de se tentar chegar a uma melhor sincronização dos sinais no tempo. O modelo perceptual do PESQ utiliza a noção de efeito de memória recente, onde distorções no fim do sinal sonoro degradam mais a qualidade que no início.

A avaliação final do PESQ é a combinação linear do valores médios da perturbação e da perturbação assimétrica. Esse valor final pertence ao intervalo [-0,5–4,5] e pode ser convertido para a escala do MOS de [1–5]. Apesar do PESQ retornar um MOS sem necessidade de avaliadores, há limitações. O PESQ mede a qualidade apenas num sentido, desconsiderando fatores de degradação como o atraso total, que afeta a interatividade, e

o eco. Além disso, o acesso direto à gravação original e à gravação degradada é intrusivo e completamente inadequado para aferir a qualidade de chamadas privadas. Em serviços VoIP é desejável um método objetivo, computacionalmente eficiente, de implantação simples e que não precise analisar diretamente a mídia, como o Modelo E.

2.2. O Modelo E

Um método de avaliação de qualidade de chamadas amplamente utilizado é o Modelo E, definido na recomendação G.107 [ITU-T 2009], que calcula uma métrica única de qualidade de voz, chamada fator R, que pode ser convertida para a escala do MOS. Ele foi desenvolvido para a fase de planejamento de sistemas telefônicos, mas passou a ser usado em medições de qualidade de serviços VoIP. O Modelo E adapta-se tanto à telefonia tradicional, onde a degradação é devida à atenuação do sinal e a ruído, quanto às redes multimídia modernas, cujos fatores de degradação principais são perda, eco, atraso ida e volta e o impacto da codificação da voz

O princípio básico do modelo é baseado num conceito estabelecido há mais de 30 anos atrás por J. Allnatt [Allnatt 1975]: "fatores psicológicos na escala psicológica são aditivos." Este princípio faz com que o Modelo E seja facilmente extensível. A saída do modelo, ou seja, a qualidade ao final da chamada, é o fator R, formulado abaixo:

$$R = R_o - I_s - I_d - I_e + A \tag{1}$$

R_o representa a relação sinal-ruído básica, enquanto os fatores I_s , I_d , I_e e A mapeiam percepções do usuário. I_s mapeia a degradação que afeta diretamente o sinal de voz, como volume alto demais, interferência sobre o *headset* e distorção da digitalização da voz. Já o fator I_d representa a degradação devido ao atraso, como existência de eco e perda de interatividade, quando o atraso total for muito longo. I_d deve considerar os atrasos de codificação e decodificação, da rede e do *buffer* de compensação de *jitter*. O fator I_e mapeia a degradação do equipamento, como *codecs* de baixa taxa. Como a rede também é considerada um equipamento, este fator também considera a perda de pacotes. Finalmente, A é chamado de fator de expectativa ou de vantagem e reflete o fato da qualidade percebida ser influenciada pela flexibilidade operacional em se estabelecer uma conexão. Por exemplo, em redes celulares usa-se A=5, compensando o efeito subjetivo da qualidade inferior da transmissão.

O fator R pode ser convertido para MOS segundo [ITU-T 2009] assumindo $\text{MOS}(R) = 1 + 0.035 \cdot R + 7 \cdot 10^{-6}(R-60)(100-R)$, se $0 \leq R \leq 100$. Caso contrário, $\text{MOS}(R) = 1$, se $R < 0$ e $\text{MOS}(R) = 4,5$, se $R > 100$.

2.3. Modelo E Estendido

Uma importante contribuição ao Modelo E foi feita por Allan Clark [Clark 2001]. Ela foi adotada pelo grupo TIPHON na especificação técnica ETSI TS 102 024-5 V4.1.1 [ETSI 2003], no que se chamou de Modelo E Estendido. O Modelo E Estendido altera o cálculo de I_d e I_e para considerar novos fatores como qualidade instantânea, comportamento de perda de pacotes de forma alternante entre períodos de perdas isoladas e em rajadas e o efeito de memória recente. Com as extensões, a qualidade medida pelo modelo se aproxima mais daquela percebida pelos usuários.

Oposto ao modelo original, o Modelo E Estendido considera as reações humanas tardias em reconhecer uma alteração de qualidade, para melhor ou para pior. O Modelo E Estendido usa curvas exponenciais com constantes de tempo de 5s para a transição de bom para ruim e de 15s de ruim para bom.

O comportamento alternante usa uma Cadeia de Markov (CM) com quatro estados para modelar perda em rajada ou de forma isolada. Uma rajada se inicia quando o número de pacotes recebidos entre duas perdas é menor que oito, caso contrário a perda é isolada. À medida que perdas ocorrem, as probabilidades de transição da CM são dinamicamente recalculadas, permitindo determinar a rajada média e a taxa de perda.

O efeito da memória recente reflete a maneira como um usuário se lembra da chamada após o seu fim. Um intervalo de qualidade ruim tende a ter um impacto negativo maior quando acontece próximo do fim da chamada. Num exemplo dado por Clark [Clark 2001], um período de 15s de ruído foi deslocado do início para o final de uma chamada de 60s de duração, provocando uma queda de 3,82 para 3,18 no MOS.

2.4. VQuality: Uma implementação do Modelo E

A biblioteca VQuality [Lustosa et al. 2005] implementa tanto o Modelo E quanto o Modelo E Estendido e pode ser incorporada a um cliente final para obtenção do MOS em tempo real. A VQuality foi desenvolvida em C++ usando orientação a objetos, oferecendo flexibilidade e facilidade de integração a diferentes pilhas VoIP.

A integração se dá através de uma interface orientada a eventos e métodos específicos da biblioteca devem ser chamados sempre que um quadro de voz é tocado ou quando um quadro, que deveria ser tocado, é perdido por não estar disponível. Também são informados quando o tamanho do *buffer* de compensação de *jitter* ou a estimativa do atraso da rede (RTT) mudam. A VQuality já foi integrada ao GnomeMeeting (atual Ekiga), no Linux, e ao OpenPhone, no Windows [Lustosa et al. 2005]. A biblioteca também foi integrada à ferramenta ACME [Lustosa et al. 2008], uma solução distribuída para a realização de testes automatizados de qualidade em ambiente VoIP, e ao Asterisk [Souza 2008], viabilizando a aferição de qualidade neste ambiente livre favorito de telefonia IP.

3. A Recomendação P.564

A recomendação P.564 [ITU-T 2007a] foi criada com o intuito de servir de guia de implementação e de testes de validação de modelos objetivos de monitoramento de qualidade de voz em tempo real. Ela define as características básicas que um modelo deve implementar, como quais parâmetros podem ser usados como entrada, além dos testes aos quais o modelo deve ser submetido e os resultados mínimos necessários para conformidade.

A recomendação aplica-se a modelos tanto de banda estreita (300 a 3400 Hz) quanto de banda larga (30 a 7000 Hz) e que usam as pilhas IP/UDP/RTP para o transporte da mídia. Um modelo objetivo pode ser usado em diferentes cenários como monitoramento da qualidade de transmissão em operação ou manutenção, monitoração de SLA (*Service Level Agreement*) entre provedores de telefonia IP ou uso em laboratório para testes de aferição de qualidade de equipamentos.

Os testes de conformidade medem o desempenho do modelo sob teste em comparação ao PESQ em cenários que contemplam uma gama variada de combinações de dois fatores de degradação simultâneos, denominados de pontos de interesse.

3.1. Definição do Modelo Aderente

Segundo a P.564, um modelo aderente deve fazer sua avaliação baseando-se apenas em parâmetros do fluxo pela rede e da manipulação da mídia no cliente, mas sem qualquer acesso à mídia em si. Dessa forma, um modelo aderente pode trabalhar nos seguintes modos de operação: dinâmico, estático ou embutido. No modo estático, o modelo só tem acesso ao fluxo RTP e deve conhecer *a priori* ou assumir determinadas características de operação do cliente final. Na operação dinâmica, o modelo usa informações do fluxo RTP e relatórios RTCP em sua medição de qualidade. Finalmente, no modo embutido, o modelo está co-localizado com o cliente final da chamada e pode obter dados da operação do *software*, como perdas e atraso no *buffer* de compensação de *jitter*.

A partir da análise dos dados ao seu alcance, o modelo de avaliação deve produzir, ao final da chamada, o MOS relativo à experiência do usuário final sob as condições observadas. Fatores como volume da fala, ruído de fundo, atraso e eco não devem ser refletidos no cômputo do MOS do modelo, estando fora do escopo básico da P.564.

Todavia, no caso do atraso, como o PESQ ainda é capaz de medir em pequena escala o impacto do atraso e de sua variação sobre a qualidade, a P.564 permite que o atraso do *buffer* de compensação de *jitter* seja considerado. Mas mesmo deixado de fora da avaliação principal de aderência à P.564, um fator tão importante como o atraso da rede deve ter seu impacto levado em conta em diagnósticos adicionais.

3.2. Testes de Conformidade

Os testes de conformidade consistem em comparar o MOS gerado pelo modelo sob teste ao MOS gerado pelo PESQ em uma série de variados cenários, envolvendo sempre o impacto simultâneo de dois fatores de degradação sobre o fluxo de mídia da chamada de voz. A recomendação define quatro fatores de degradação: perda aleatória de pacotes, *jitter* e dois tipos diferentes de ocupação de banda. Foi desconsiderada a ocupação da banda, pois seu efeito indireto é sentido na perda e no *jitter* que ocorrem em gargalos.

A P.564 estipula testes onde perdas acontecem apenas de forma aleatória, o que é uma limitação. Na internet, as perdas em roteadores congestionados costumam ocorrer em rajadas associadas a um alto *jitter*, o que motivou estender os testes da P.564 para cenários de perda em rajada.

Deve-se salientar que, conforme a própria recomendação, os testes devem ser repetidos para cada configuração de *codec*, *ptime* e uso de PLC ou VAD, opções configuráveis pelos usuários em chamadas VoIP. É possível que um modelo não passe nos testes para uma determinada combinação de *codec* e *ptime*, o que não significa que ele não possa ser usado com outras configurações aprovadas.

Os dois fatores de degradação de um ponto de interesse deve ser definido por um parâmetro único. Por exemplo, no caso da perda aleatória, o parâmetro é a porcentagem de pacotes perdidos na chamada. Isso permite representar os cenários dos testes num plano cartesiano cujos eixos são os valores dos parâmetros dos fatores de degradação. A Figura 1 mostra o plano formado pelo *jitter* e pela porcentagem de perda aleatória em testes com o *codec* G.711 μ-law com *ptime* 20ms e sem uso de VAD ou PLC. Essa figura será usada mais a frente para exemplificar como os pontos de interesse são encontrados.

Para cada fator de degradação deve ser encontrado I_{max}, que é o valor do parâme-

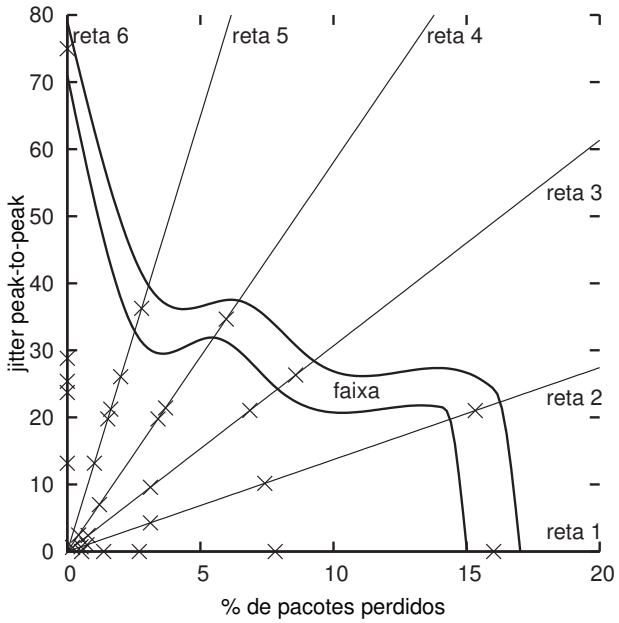

Figura 1. Pontos de interesse calculados para o G.711 μ-law com *ptime* 20ms sem VAD nem PLC

tro corespondente que, sozinho, faz com que o MOS de uma chamada fique no intervalo [1,3–1,5]. Uma vez calculado o I_{max} para os dois fatores de degradação, as escalas máximas dos eixos ficam definidas como mostrado na Figura 1. Na figura, o I_{max} do *jitter* é próximo de 80ms e o I_{max} da perda aleatória é próximo de 20%. Com as escalas ajustadas, pode-se definir a distribuição dos pontos de interesse.

Primeiramente, são definidas seis semirretas regularmente distribuídas no quadrante do plano, separadas por 18 graus. Na Figura 1, a primeira e a última reta correspondem aos eixos. Em seguida, devem ser calculados cinco pontos de interesse em cada semirreta, totalizando 30 pontos de interesse por cenário. Os pontos de interesse são combinações de parâmetros de fatores de degradação que fazem com que o MOS_{PESQ} fique em determinados intervalos. Os intervalos são calculados em função da qualidade máxima que um *codec* pode oferecer. Por exemplo, no caso do G.711, cujo MOS_{PESQ} máximo é 4,5, usando *ptime* de 20ms sem PLC ou VAD, os cinco pontos de interesse de cada reta devem produzir MOS nos seguintes intervalos: [1,8–2,1], [2,7–3,0], [3,45–3,75], [3,9–4,2] e [4,2–4,5]. A faixa destacada na Figura 1 possui os seis pontos da faixa de pior qualidade, [1,8–2,1].

É importante ressaltar que dificilmente um aumento linear de fator de degradação fará com que o MOS também caia linearmente, dificultando a localização de pontos no plano que representam cenários com MOS em intervalo desejado. Geralmente, a influência de um fator sobre a qualidade é complexa e a disposição dos pontos de interesse e seus intervalos no plano acaba não sendo simétrica, como no exemplo da Figura 1.

Todos os experimentos devem ser realizados sobre quatro arquivos de áudio fornecidos pela recomendação. Cada arquivo possui duas frases sem conexão lógica entre elas, espaçadas por alguns segundos, num total de menos de oito segundos de áudio cada. Os áudios são de uma mulher falando inglês britânico, uma mulher falando francês e dois são

Tabela 1. Faixas de classificação dos desvios das avaliações e resultados necessários para aderência de um modelo à P.564

Medida	Limite inferior	Limite superior	Critério	
			Classe C1	Classe C2
Correlação	-	-	$>$ 0,90	$>$ 0,85
Falsos negativos (%)	-	-	$<$ 5,0	$<$ 5,0
Falsos positivos (%)	-	-	$<$ 3,0	$<$ 3,0
Grupo 1 ($MOS_{PESQ} \geq$ 2,8)				
% de desvios na faixa 1	-0,25	+0,25	$\geq$ 95,0	$\geq$ 75,0
% de desvios na faixa 2	-0,35	+0,5	$\geq$ 97,9	
% de desvios na faixa 3	-0,5	+0,5		$\geq$ 95,0
% de desvios na faixa 4	-0,6	+1,0	$\geq$ 99,0	
% de desvios na faixa 5	-1,0	+1,0		$\geq$ 97,9
% de desvios na faixa 6	-1,5	+1,5		$\geq$ 99,0
Grupo 2 ($MOS_{PESQ} <$ 2,8)				
% de desvios na faixa 7	-0,4	+0,4	$\geq$ 90,0	
% de desvios na faixa 8	-0,5	+0,5		$\geq$ 90,0
% de desvios na faixa 9	-0,7	+0,7	$\geq$ 95,0	
% de desvios na faixa 10	-1,0	+1,0		$\geq$ 95,0
% de desvios na faixa 11	-1,2	+1,2	$\geq$ 99,0	
% de desvios na faixa 12	-1,5	+1,5		$\geq$ 99,0

de um mesmo homem falando espanhol. O tempo de fala e de silêncio é aproximadamente o mesmo durante os áudios.

A recomendação deixa a critério do aplicador realizar os experimentos através de chamadas reais ou emuladas. Em ambos os casos, os mesmos quatro arquivos de áudio devem ser usados e o resultado de cada rodada é a qualidade média medida. A recomendação permite que concatenações dos quatro arquivos sejam usadas. Mais de uma rodada pode ser feita para cada cenário e, neste caso, a avaliação é a média das avaliações das rodadas. Todavia, nenhum rigor estatístico é especificado na recomendação.

Depois de encontrados todos os 30 pontos de interesse, deve-se fazer a avaliação de todos os pontos novamente, agora utilizando o modelo sob teste. Com os resultados das avaliações, pode-se calcular para cada ponto de interesse o desvio da previsão dado por $desvio = MOS_{modelo} - MOS_{PESQ}$, como definido na P.564.

A primeira medida de comparação exigida pela P.564 é o coeficiente de correlação entre as duas séries de avaliações, que serve para medir a dependência linear entre elas. Sabendo que as duas séries de previsões encontram-se na mesma escala (MOS) e são previsões de um mesmo fenômeno, espera-se uma correlação alta. Em seguida, deve-se classificar os pontos de interesse em dois grupos. No Grupo 1, ficam os pontos nos quais $MOS_{PESQ} \geq$ 2,8. No Grupo 2, ficam aqueles com $MOS_{PESQ} <$ 2,8, ou seja, a qualidade foi ruim ou péssima, inviabilizando o serviço.

Para cada grupo, os desvios de previsão do modelo sob teste devem ser classificados em diferentes faixas, que servem para observar a sua distribuição. Uma faixa engloba todos os desvios cujos valores estão entre os valores das colunas **limite inferior** e **limite superior** mostrados na Tabela 1. Por exemplo, caso o desvio da previsão de um ponto de interesse do Grupo 1 seja -0.3, aquele ponto não se enquadra na faixa 1, porém se encaixa nas faixas de 2 a 6. Também deve ser calculada a porcentagem de pontos de interesse den-

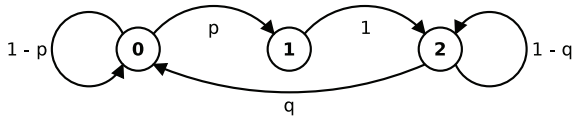

Figura 2. Cadeia de Markov de modelo de perdas em rajada de tamanho > 1

tro de um grupo cujo desvios de previsão se enquadra em cada faixa. Pode-se observar que toda vez que um ponto se enquadra numa faixa mais restritiva, ele automaticamente se enquadra a todas as outras faixas mais amplas.

Por último, devem ser calculadas as porcentagens de medições classificadas como falsos negativos e falsos positivos. Por definição, os falso negativos são aqueles casos onde $MOS_{PESQ} < 2{,}0$ e $MOS_{modelo} \geq 3{,}0$. Analogamente, os casos de falso positivos são aqueles onde $MOS_{PESQ} \geq 3,0$ e $MOS_{modelo} < 2{,}0$.

Finalmente, as medidas de comparação são checadas contra os limites na Tabela 1 para classificação em uma de duas classes de conformidade. A classe C1 demonstra uma maior qualidade, mas é mais difícil de ser alcançada. Já a classe C2 é mais flexível e mais fácil de ser alcançada, porém indica uma qualidade de previsão inferior.

4. Aplicação da P.564 ao Modelo E

Em relação a sua arquitetura, o Modelo E implementado pela VQuality se enquadra no modo de operação embutido da P.564, obtendo informações passadas pelo software do cliente final. O cliente repassa à VQuality dados sobre os diferentes tipos de atraso que a mídia sofre (atraso da rede, do *buffer* de compensação de *jitter* e de codificação) e eventos de quando a mídia é tocada ou perdida.

Tanto o Modelo E original da ITU-T quanto o Modelo E Estendido por Clark foram submetidos ao teste de conformidade da P.564. Como o uso de ambos é idêntico, as particularidades aqui descritas sobre a aplicação da P.564 servem para os dois modelos.

Ainda que a P.564 não permita que sejam usados parâmetros de atraso no modelo sob teste, ela permite que o impacto do atraso do *buffer* seja usado no I_d. Os outros termos do fator R são calculados da maneira usual.

Para realizar de forma automatizada os testes de conformidade, foi criado um emulador de chamadas de voz. A codificação/decodificação de voz é feita como numa chamada real, mas a mídia não trafega pela rede. Ao invés disso, as degradações que a rede causaria são emuladas no software, que também é responsável por chamar as bibliotecas que avaliam a qualidade da chamada.

Para implementar a perda aleatória, uma porcentagem dos pacotes igual ao parâmetro do fator de degradação é descartada uniformemente na chamada. Já o parâmetro do *jitter* é a variação máxima de atraso que um pacote pode sofrer, medida conhecida como *peak-to-peak jitter*. Esse fator é emulado sorteando um atraso aleatório entre zero e o parâmetro do fator de degradação, o que causa que pacotes tenham atraso variável e pode causar pacotes fora de ordem no receptor.

A emulação de perdas em rajada é feita usando uma CM inspirada no Modelo de Gilbert [Gilbert 1960] mostrada na Figura 2. No estado **0**, gera-se perdas com probabilidade p, transicionando-se para o estado **1**. Uma perda consecutiva é gerada e transiciona-

se para o estado **2**. No estado **2**, com probabilidade $(1 - q)$ perdas subsequentes ocorrem e com probabilidade q um pacote é recebido com sucesso, transicionando para o estado **0**. A CM garante rajadas de perda de pelo menos dois pacotes.

Analisando o estado estacionário da CM, chega-se à conclusão que o tamanho esperado das rajadas é dado por $(1+1/q)$. Entre todos os pontos de interesse dos cenários de perdas em rajada, foram criadas rajadas de tamanho médio de até 3,3 pacotes.

Dois parâmetros, as probabilidades p e q, são necessários para definir um cenário de perda em rajada usando a CM. Assim, os experimentos com perdas em rajada usam apenas a perda como fator de degradação, operando sem *jitter* e com apenas um pequeno atraso equivalente ao tamanho mínimo do *buffer* de compensação de *jitter*. A obtenção e a análise dos pontos de interesse foram mantidos como originalmente proposto pela P.564.

4.1. Cálculo dos Pontos de Interesse

Apesar de definir matematicamente como os pontos de interesse devem ser formados, a recomendação P.564 não define qualquer metodologia para encontrar os pontos. Assim, foi escolhido usar busca binária sobre as retas com os parâmetros dos fatores de degradação para encontrar valores que implicam num MOS conforme desejado.

A fim de agregar confiabilidade, foi decido usar certo rigor estatístico nos cálculos. Dessa forma, o cálculo dos pontos de interesse e das médias do MOS do Modelo E e do PESQ nos pontos de interesse foram todos realizados e repetidos até que fosse alcançado um intervalo de confiança de 95% menor que 10% do valor do centro do intervalo. Para encontrar um ponto de interesse em uma reta, é feita uma busca sobre a reta variando a distância à origem. Cada ponto buscado é testado para que se descubra se é um ponto de interesse válido ou não, isto é, se todo o intervalo de confiança do MOS_{PESQ} correspondente está dentro do intervalo que define o ponto de interesse. O primeiro ponto de interesse dentro da fixa satisfazendo à restrição acima é considerado como um ponto de parada para a busca na faixa naquele eixo. Uma busca é feita para cada um dos 30 pontos de interesse de cada teste de conformidade.

4.2. Emulador

Os experimentos necessários à aplicação da P.564 foram realizados com um software emulador de chamadas criado especificamente para esta tarefa. Ele foi escrito em C++ a fim de utilizar a facilidade da orientação a objetos e, ao mesmo tempo, usar a biblioteca PJSIP [Prijono 2010], escrita em C, para o tratamento da mídia. A PJSIP é usada para codificar/decodificar a mídia e para emular o *buffer* de compensação de *jitter*, cujo tamanho é modificado dinamicamente conforme a variação do atraso.

Para cada chamada emulada, quatro permutações dos quatro arquivos de áudio disponibilizados pela P.564 (totalizando 32s de chamada) são lidas e codificadas por inteiro, a degradação é emulada sobre a mídia, os pacotes passam pelo *buffer* de compensação de *jitter* da PJSIP, a mídia é decodificada, o $MOS_{ModeloE}$ é medido VQuality, o áudio é decodificado num arquivo e, finalmente, é usado o PESQ para medir a qualidade, comparando o arquivo de entrada com o de saída. Os passos descritos são executados serialmente de forma que toda a mídia passa uma só vez por cada um dos passos.

O emulador também é capaz de realizar as buscas necessárias e calcular por conta

Tabela 2. Resultado dos testes de conformidade para o G.711 sem VAD nem PLC

Medida	Modelo E					Modelo E Estendido				
	ptime (ms)					*ptime* (ms)				
	10	**20**	**30**	**40**	**50**	**10**	**20**	**30**	**40**	**50**
Correlação	0,99	0,98	0,97	0,98	0,97	0,98	0,98	0,98	0,98	0,98
Falsos negativos (%)	0	0	0	0	0	0	0	0	0	0
Falsos positivos (%)	0	0	0	0	0	0	0	0	0	0
Grupo 1 ($MOS_{PESQ} \geq$ 2,8)										
% de desvios na faixa 1	78,2	94,7	82,6	65,2	87,5	73,9	94,7	82,6	82,6	95,8
% de desvios na faixa 2	82,6	100	95,6	100	100	78,2	100	100	95,6	100
% de desvios na faixa 3	91,3	100	100	100	100	78,2	100	100	100	100
% de desvios na faixa 4	91,3	100	100	100	100	78,2	100	100	100	100
% de desvios na faixa 5	100	100	100	100	100	100	100	100	100	100
% de desvios na faixa 6	100	100	100	100	100	100	100	100	100	100
Grupo 2 ($MOS_{PESQ} <$ 2,8)										
% de desvios na faixa 7	14,2	90,9	71,4	71,4	83,3	28,5	72,7	100	85,7	83,3
% de desvios na faixa 8	71,4	100	85,7	100	83,3	71,4	81,8	100	100	100
% de desvios na faixa 9	85,7	100	85,7	100	100	100	100	100	100	100
% de desvios na faixa 10	100	100	100	100	100	100	100	100	100	100
% de desvios na faixa 11	100	100	100	100	100	100	100	100	100	100
% de desvios na faixa 12	100	100	100	100	100	100	100	100	100	100
Resultado (C2)	×	✓	×	×	×	×	×	✓	✓	✓

própria os pontos de interesse para cada cenário, além de fornecer, ao final, um relatório de conformidade e aderência dos Modelos E e E Estendido à P.564.

5. Resultados dos Testes de Conformidade com o Modelo E

O suporte do Modelo E a diferentes *codecs* se dá com o uso de algumas constantes em suas fórmulas. Os *codecs* suportados oficialmente são o G.711, G.723, G.728, G.729 e o GSM [ITU-T 2009]. *Codecs* mais modernos como Speex e iLBC não são ainda suportados.

Os *codecs* suportados pela versão *open source* da biblioteca PJSIP são G.711, G.722, GSM, iLBC e Speex. Como outros *codecs* proprietários exigem alguma espécie de licença, sua codificação e decodificação não são feitas pela biblioteca. Por outro lado, caso seja instalado uma segunda biblioteca com suporte a tais *codecs*, o PJSIP pode usá-la para oferecer os *codecs* proprietários através da sua interface.

Tendo em vista as limitações do Modelo E e da biblioteca em relação à oferta de *codecs*, os testes se limitaram à avaliação da conformidade do Modelo E à recomendação P.564 apenas com os *codecs* G.711 μ-law e GSM. Os testes foram feitos para diferentes valores de *ptime*, variando de 10 a 100ms, com e sem PLC e VAD.

Depois de serem realizadas todas as chamadas para os pontos de interesse de todos os casos desejados, sempre com rigor estatístico, foram montadas tabelas como exemplificado pela Tabela 2, que representa o relatório final de teste de conformidade para os Modelos E original e E Estendido para G.711 μ-law sem PLC e sem VAD.

Pode-se observar a distribuição dos desvios para cinco *ptimes* diferentes na Tabela 2. As células hachuradas indicam os valores medidos que fizeram com que a configuração fosse reprovada nos critérios da classe C2. Nota-se que alguns casos conseguiram resultados bons suficientes para serem aprovados pelos critérios de C2, enquanto nenhum

Tabela 3. Resultados dos testes de conformidade da P.564

Cenário de degradação	Modelo de medição	*ptime* (ms)	G.711 μ-law				GSM			
			—		VAD		—		VAD	
			—	PLC	—	PLC	—	PLC	—	PLC
Perdas aleatórias e jitter	Modelo E	10	-	-	-	-				
		20	C2	-	-	-	C2	C1	C1	C1
		30	-	-	-	-				
		40	-	-	-	-	-	C2	C1	C1
	Modelo E Estendido	10	-	-	-	-				
		20	-	-	-	-	C1	C1	C1	C1
		30	C2	-	C2	-				
		40	C2	-	C2	-	-	C1	-	C1
Perdas em rajada	Modelo E	10	-	-	-	-				
		20	-	-	-	-	C2	C1	C1	C1
		30	-	-	-	-				
		40	-	-	-	-	-	C2	C1	C1
	Modelo E Estendido	10	-	-	-	-				
		20	C1	-	C2	-	C1	C1	C1	C1
		30	C1	-	C1	-				
		40	C1	-	C1	-	-	C1	-	C1

caso foi aprovado para a classe C1.

Na Tabela 2, há uma aparente recuperação do desempenho dos modelos para os casos com *ptime* de 50ms em relações aos casos de 40ms. Esse fenômeno, porém, parece ser causado pela imprecisão do PESQ quando quadros maiores são perdidos e substituídos por silêncio. O modelo de avaliação do PESQ parece ter problemas em medir a qualidade em casos com grandes períodos de silêncio.

Pennock [Pennock et al. 2002] já havia relatado resultados ruins do PESQ em casos de perdas em rajada, porém sem discriminar o *ptime* utilizado. O mesmo trabalho indica que a qualidade das avaliações do PESQ diminuem conforme aumenta o *ptime* e que a baixa precisão com perdas em rajada ocorre também com *ptimes* grandes. Pode-se observar que quando *ptime* pequenos (10ms, 20ms ou 30ms) são usados, uma perda em rajada pode ser vista como equivalente a uma perda isolada de um *ptime* grande.

Uma possível justificativa para tal comportamento é a dinâmica entre o tamanho do *ptime* e da elocução empregada pelo PESQ, de 32ms. A perda de um pacote com *ptime* maior que o 32ms significa pelo menos uma elocução inteira perdida. Com o aumento do *ptime*, mais elocuções consecutivas são inteiramente perdidas, causando uma perturbação no sinal que força o algoritmo de alinhamento temporal do PESQ a tentar realinhar a fala com maior frequência, piorando a precisão da avaliação de qualidade. Mesmo que a P.564 não leve isso em consideração, a consequência de tais problemas com o PESQ é que os resultados de conformidade para *ptimes* maiores que 40ms são progressivamente menos confiáveis. Considerando tais fatos, os testes de conformidade foram limitados a *ptimes* de, no máximo, 40ms.

A Tabela 3 resume os resultados de conformidade de todos os cenários propostos usando os *codecs* G.711 e GSM. Enquanto a PJSIP consegue codificar qualquer *ptime* múltiplo de 10ms para G.711, para GSM, só é possível usar *ptimes* múltiplos de 20ms.

Em relação ao desempenho do G.711, os resultados sugerem que o Modelo E foi

projetado e testado usando o *ptime* de 20ms, sem PLC e sem VAD, já que este cenário foi o único no qual o modelo passou nos testes. Parece razoável assumir que a variação do *ptime* não foi considerada ao projetar o Modelo E, já que os textos do Modelo E e do Modelo E Estendido não tratam do tema.

Como esperado, o desempenho do Modelo E Estendido foi melhor que o do Modelo E original maior. Isso se deu por duas vantagens do Modelo E Estendido. A primeira é que, ainda que de forma limitada, o *ptime* é considerado no cálculo do MOS. Afinal, o Modelo E Estendido usa a informação do tamanho do *ptime* para calcular a duração dos períodos de perda em rajada e de perdas isoladas. Além disso, o *buffer* de compensação de *jitter* costuma provocar a perda de diversos quadros de voz cada vez que é sincronizado, provocando perdas em rajada, situação para a qual o Modelo E Estendido foi preparado. A Tabela 2 também exemplifica como a distribuição de desvios do Modelo E Estendido tende a ser melhor que a do Modelo E original.

Quando usado G.711, o Modelo E Estendido passou nos testes de conformidade em mais casos que o Modelo E tanto quando a chamada foi degradada com perdas aleatórias e *jitter* quanto nos casos de perdas em rajada, situação na qual o Modelo E não passou em nenhum caso. Já quando foi usado GSM, o resultado dos dois modelos foi muito parecido. Com este *codec* os resultados dos testes de perdas aleatórias e *jitter* foi exatamente o mesmo dos testes de perdas em rajada.

Em relação ao desempenho dos modelos com uso de GSM, somente casos que não usavam PLC foram reprovados. Nenhum caso foi reprovado quando foi usado *ptime* de 20ms. Quatro casos foram reprovados quando não foi usado VAD, enquanto apenas dois foram reprovados quando tal recurso foi usado. Esses fatos levam a crer que o Modelo E, no caso do GSM, foi calibrado considerando um *ptime* de 20ms e o uso de VAD e PLC.

6. Conclusões e Trabalhos Futuros

O Modelo E e sua extensão, o Modelo E Estendido, foram apresentados neste trabalho como opções de modelos de avaliação da qualidade de chamadas de voz em tempo real cuja eficiência precisava ser medida. Assim, foi apresentada a recomendação P.564 cuja finalidade é exatamente servir de teste de conformidade para tais modelos.

Após adequadamente apresentada a recomendação, foi discutido como aplicar sua metodologia ao caso do Modelo E. Foram apresentadas como contribuições as extensões aos testes de conformidade da P.564 o uso de confiança estatística e o uso de cenários de perdas em rajada. Diversas chamadas foram emuladas seguindo a P.564 a fim de comparar as avaliações do Modelo E às do PESQ, algoritmo de referência para avaliar a qualidade de chamadas de voz. Os testes de conformidade foram feitos usando os *codecs* G.711 μ-law e GSM, diferentes valores de *ptime* e tanto com quanto sem o uso de PLC e VAD.

Os resultados dos testes mostraram que a concepção dos modelos não considera o uso de diferentes *ptimes* ou de VAD ou PLC. Os textos que definem os dois modelos não discutem a variação de seus parâmetros conforme o *ptime* utilizado, assim especula-se que os modelos foram calibrados apenas com o *ptime* de 20ms.

Os resultados dos modelos foram melhores quando o *codec* GSM foi usado, o que indica que o Modelo E foi melhor calibrado para uso com este *codec*. No caso do G.711, estipula-se que o modelo foi calibrado sem o uso de PLC ou VAD. Já no caso do GSM,

estipula-se que o Modelo E foi calibrado usando tanto PLC quanto VAD.

Essas conclusões abrem portas à pesquisa de melhorias do Modelo E tal que ele passe a ser sensível à variação do *ptime* e ao uso de PLC e VAD, podendo enfim passar nos testes de conformidade da P.564 para uma gama maior de configurações de codificação.

Referências

Allnatt, J. (1975). Subjective rating and apparent magnitude. *International Journal of Man-Machine Studies*, 7(6):801 – 816.

Clark, A. D. (2001). Modeling the Effects of Burst Packet Loss and Recency on Subjective. In *Proceedings of the Internet Telephony Workshop (IPTEL '01)*.

ETSI (2003). ETSI TS 102 024-5 V4.1.1. Telecommunications and Internet Protocol Harmonization Over Networks (TIPHON) Release 4, End-to-end Quality of Service in TIPHON systems; Part 5: Quality of Service (QoS) measurement methodologies. ETSI, França.

Gilbert, E. N. (1960). Capacity of a burst-noise channel. *Bell Systems Technical Journal*, 39:1253–1265.

ITU-T (1996). Recomendação P.800 — Methods for subjective determination of transmission quality. ITU, Geneva.

ITU-T (2001). Recomendação P.862 — Perceptual evaluation of speech quality (PESQ): An objective method for end-to-end speech quality assessment of narrow-band telephone networks and speech codecs. ITU, Geneva.

ITU-T (2007a). Recomendação P.564 — Conformance testing for voice over IP transmission quality assessment models. ITU, Geneva.

ITU-T (2007b). SG 12 - C 100 — E-model P.564 Compliance Testing and E-model evolution proposals. ITU, Geneva.

ITU-T (2009). Recomendação G.107 — The E-model: a computational model for use in transmission planning. ITU, Geneva.

Lustosa, L., Rodrigues, P., David, F., and Quinellato, D. (2005). Arquitetura de Monitoração de Qualidade de Chamadas Telefônicas IP. In *Proceedings of SBRC '05*, pages 1073–1086.

Lustosa, L., Souza, A., de A. Rodrigues, P., and Quinellato, D. (2008). ACME: An Automated Tool for Generating and Evaluating the Quality of VoIP Calls. In *Management of Converged Multimedia Networks and Services*, volume 5274 of *Lecture Notes in Computer Science*, pages 91–103. Springer Berlin / Heidelberg.

Pennock, S., Holub, J., and Smid, R. (2002). Accuracy of the perceptual evaluation of speech quality (PESQ) algorithm. In *Proceedings of Online Workshop on Measurement of Speech and Audio Quality in Networks*, pages 49–68.

Prijono, B. (2010). *Open source SIP stack and media stack for presence, im/instant messaging, and multimedia communication*. Online: http://www.pjsip.org (acesso em 20/12/2010).

Souza, A. A. D. P. (2008). Integração de Qualidade de Voz a Gateway Asterisk. Projeto Final de Curso. DCC/UFRJ.