UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
CURSO DE ENGENHARIA DE COMPUTAÇÃO

GUSTAVO FRANCISCO

Aceleração em *hardware* de algoritmo de redução de ruído com preservação de cenário acústico para aparelhos auditivos binaurais

Porto Alegre
2021

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
CURSO DE ENGENHARIA DE COMPUTAÇÃO

GUSTAVO FRANCISCO

Aceleração em *hardware* de algoritmo de redução de ruído com preservação de cenário acústico para aparelhos auditivos binaurais

Monografia apresentada como requisito parcial para a obtenção do grau de Bacharel em Engenharia da Computação

Orientador: Prof. Gabriel Luca Nazar
Co-orientador: Prof. Fábio Pires Itturriet

Porto Alegre
2021

Dedico este trabalho aos meus irmãos Renato e Karen, meu pai Vanderlei e minha mãe Marli, por todo carinho e apoio.

AGRADECIMENTOS

Se eu fosse agradecer todos aqueles que passaram e me ajudaram até aqui, este seria provavelmente o maior capítulo deste trabalho.

Agradeço primeiramente à Deus, que me permitiu seguir este caminho, à minha família, que mesmo de longe, seja no Rio ou Salvador, sempre enviaram boas energias e pensamentos positivos para que tudo isso fosse possível. As chamadas de vídeo e ligações, além das visitas, ajudavam a injetar combustível rumo à graduação.

Agradecer a Bruna minha namorada, que sempre me apoiou e ajudou em todos momentos que precisei, com carinho e atenção mesmo quando eu explicava os detalhes técnicos dos problemas e ela não entendia nada e dizia que eu ia conseguir, que era só colocar um "printf".

Gostaria de agradecer também aos amigos que me acompanham a mais tempo e que me ajudaram a seguir firme nessa caminhada, como no grupo VMNV composto por Gaspa, Deco, Buy, S, Rot, Ma, Guimo e por último mas não menos importante meu grande amigo Facini, que anda comigo desde os 6 ou 7 anos nas categorias fraldinha do CEPE e ajudou muito em todos nessa trajetória.

Agradecer também aos amigos que o futebol me proporcionou, pelo PNC do China ou Futsal da UFRGS, que fomentaram e ajudaram no caminho até aqui, seja com passes pra gol ou resenhas com caxeta, que sem isso tudo seria muito mais difícil.

Aos meus amigos e colegas de computação, que sempre foram solícitos e auxiliaram para que tudo fosse possível, com grupos de estudos e ajudas até a madrugada, e que sem eles eu não estaria hoje a 3 anos campeão de futsal do IntegraENG, além do título do Dacompeonato.

E também ao meu orientador Gabriel Nazar e coorientador Fábio Iturriet, que sempre se mostraram extremamente solícitos, e ajudaram a deixar esses meses de conversas e reuniões mais leves.

“You will never find a rainbow if you are looking down.”

— Charlie Chaplin

RESUMO

O uso de aparelhos auditivos para quem possui deficiências auditivas é algo extremamente importante. Porém, ainda há muito a se avançar em melhorias para os dispositivos auditivos. Dentre eles, a redução de ruído é uma de muito interesse pelos usuários. Técnica de redução de ruido, além de funções aplicadas juntamente para manutenção das posições espaciais dos sons, aumentam a experiencia de quem utiliza o aparelho. Para utilização em tempo real, existem fatores importantes a serem analisados para que a aplicação dos algoritmos seja sólida e sem maiores problemas, sem que haja atrasos com o tempo de processamento, visando não causar desconforto ao usuário. Estes pontos serão explorados neste trabalho com a elaboração de uma implementação de algoritmo de redução de ruído, juntamente com a utilização de técnica de preservação de cenário acústico, em *hardware*, usando como ponto de partida o FPGA para utilização como dispositivo de borda. Serão analisados os resultados da aplicação do algoritmo de redução nos cenários acústicos simulados, demonstrando os benefícios e as penalidades de adotar paralelamente técnicas de preservação.

Palavras-chave: Audição. FPGA. Lateralização. Redução de ruído.

Noise reduction algorithm implemented in hardware with acoustic scenario preservation for binaural hearing aids

ABSTRACT

The use of hearing aids for people with hearing impairments is extremely important. However, there is still a long way to go ahead improving hearing devices. Among them, noise reduction is one of great interest to users. Noise reduction technique, in addition to functions applied together to maintain the spatial positions of sounds, increase the experience of those who use the device. For real-time use, there are important factors to be analyzed so that the application of algorithms is solid and without major problems, with no delays in processing time, so as not to cause discomfort to the user. These points will be explored in this work with the elaboration of an implementation of a noise reduction algorithm, together with the use of the acoustic scenario preservation technique, in hardware, using the FPGA as a starting point for use as an edge device. The results of the application of the reduction algorithm in simulated acoustic scenarios will be analyzed, demonstrating the benefits and harms of adopting preservation techniques in parallel.

Keywords: Acoustic. Binaural. FPGA. Hearing. Speech..

LISTA DE ABREVIATURAS E SIGLAS

DSP	Digital Signal Processor
DTFT	Discrete-Time Fourier Transform
FPGA	Field-Programmable Gate Array
HRIR	Head Related Transfer Functions
HLS	High Level Syntesis
ILD	Interaural Level Delay
IPD	Interaural Phase Difference
ITD	Interaural Time Delay
ITF	Interaural Transfer Function
ICRA	International Colle-gium of Rehabilitative Audiology
MOS	Mean Opinion Score
MWF	Multichannel Wiener Filter
PESQ	Perceptual Evaluation of Speech Quality
RTL	Register-Transfer Level
STFT	Short-Term Fourier Transform
STOI	Short Term Objective Intelligibility
SNR	Signal-Noise Rate
VAD	Voice Activity Detection

LISTA DE FIGURAS

LISTA DE TABELAS

SUMÁRIO

1 INTRODUÇÃO

A audição é um dos cinco sentidos usados pelos seres humanos para captação de informações que os conecta ao ambiente que estão inseridos. Apresenta papel fundamental na vida dos seres humanos, principalmente na parte de comunicação, impactando diretamente na vida social e qualidade de vida. Além disso, o ato de ouvir pelos seres humanos é uma função que nunca é desativada, e por conta disso, possui importante função de alerta. Uma série de fatores como envelhecimento, exposição prolongadas a ruídos, efeitos colaterais de algumas medicações, entre outros, podem comprometer a capacidade auditiva em diferentes níveis. Elemento importante na locomoção e equilíbrio do ser humano, a perda de audição pode ocasionar problemas domésticos, devido à falta de consciência espacial em relação a outras pessoas, animais ou objetos, além de perigos em travessia de ruas e calçadas. Estima-se que um quarto da população, ou 2,5 bilhões de pessoas no mundo, terá algum grau de perda auditiva em 2050 (WORLD HEALTH ORGANIZATION, 2021).

As perdas auditivas podem ser diagnosticadas através de um exame chamado audiometria, aplicado por médicos ou fonoaudiólogos. Quando constatadas, dispositivos conhecidos como aparelhos auditivos são usualmente indicados por esses profissionais para tentar restaurar a capacidade auditiva perdida. Perdas auditivas em apenas um dos ouvidos, chamada unilateral, demanda o uso do aparelho auditivo apenas no lado comprometido. Por outro lado, perdas em ambos os ouvidos necessitam o uso de aparelhos bilaterais (independentes) ou aparelhos binaurais (com comunicação entre si). Estudos atuais mostraram que 2 em cada 3 pessoas buscam por aparelhos auditivos binaurais (HEARING. . . , 2020 (Acessado Out 20, 2021)). A principal função dos aparelhos é amplificar as frequências que apresentam perdas de acordo com o resultado do exame audiométrico. Esses aparelhos são constituídos por três módulos principais: (i) um ou mais microfones, responsáveis pela captação dos sinais sonoros do ambiente em torno do usuário, (ii) uma unidade de processamento digital de sinais para amplificação e execução dos algoritmos de processamento de fala e (iii) um alto-falante para reproduzir os áudios processados no ouvido do usuário.

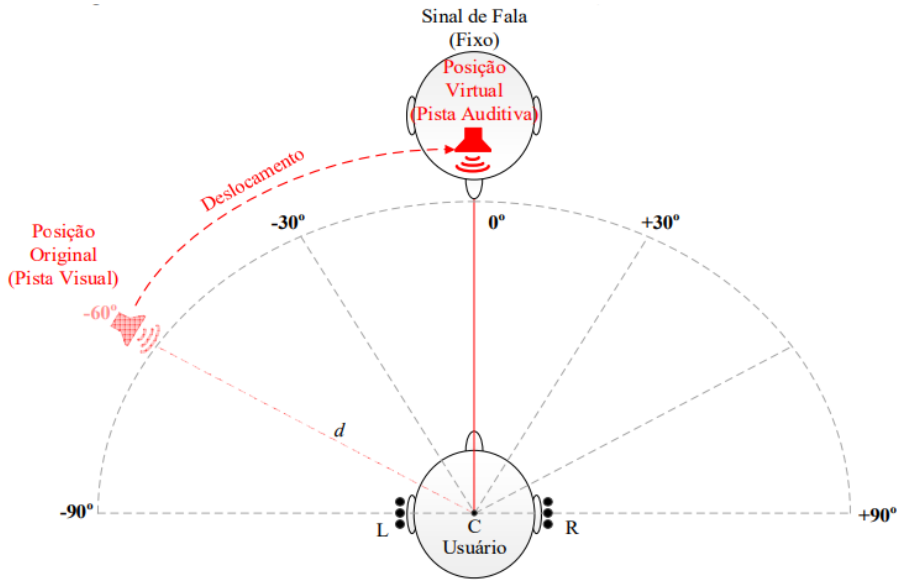
Entretanto, o áudio captado pelos microfones, amplificado e enviado ao ouvido do usuário é composto por todas as fontes sonoras presentes em cenários acústicos reais. Isso inclui as fontes de interesse (geralmente fala) e as fontes de ruídos acústicos existentes na vida cotidiana como aparelhos de ar-condicionado, ventiladores, automóveis, etc.

A amplificação do som emitido por essas fontes se torna extremamente desconfortável ao usuário, e compromete a inteligibilidade dos sinais de interesse do usuário. Por esse motivo, técnicas de redução de ruído são embarcadas nos aparelhos auditivos visando evitar esse problema. Em geral, ao reduzir o ruído acústico presente, as técnicas existentes também alteram as duas principais pistas binaurais do ruído remanescente definidas na literatura como a diferença de tempo interaural (ITD - *Interaural Time Difference*) e a diferença de nível interaural (ILD - *Interaural Level Difference*) (BLAUERT, 1997). O cérebro utiliza essas pistas binaurais para identificar a posição de fontes sonoras pontuais nos cenários acústicos. Por isso, alterar as pistas binaurais originais de uma fonte sonora, acarreta na mudança da percepção subjetiva da posição dessas fontes, gerando uma imagem virtual distorcida da fonte sonora, diferente da posição real. Esse descasamento entre as pistas visuais e sonoras (binaurais) gera confusão e desconforto aos usuários de aparelhos auditivos binaurais. Para que isso não ocorra, as técnicas de redução de ruído devem possuir estratégias de preservação das pistas binaurais originais captadas através dos microfones no áudio processado e enviado aos alto-falantes dos aparelhos.

Um exemplo deste problema pode ser observado no cenário apresentado na Figura 1.1, onde um usuário de aparelhos auditivos binaurais com uma técnica de redução de ruído está conversando com outra pessoa dentro de um cenário acústico acrescido de uma fonte de ruído. A posição de fontes sonoras no plano é chamada de azimute e tem valores compreendidos entre -90º e +90º. O azimute de 0º, onde encontra-se a fonte de fala, corresponde a posição em frente ao usuário. No mesmo cenário, o azimute real da fonte de ruído é de -60º. Porém, como relatado anteriormente, as técnicas de redução de ruído tendem a modificar as pistas binaurais e, por conseguinte, deslocar a imagem real da fonte de ruído, gerando uma segunda imagem virtual no mesmo azimute da fonte de fala. Esse efeito indesejado compromete a consciência espacial do usuário dos aparelhos e pode acarretar até mesmo em acidentes em situações cotidianas.

A organização EHIMA (*European Hearing Instrument Manufacturers Association*), que é composta pelos principais fabricantes de aparelhos auditivos no mercado atual e representa mais de 90% da produção global de aparelhos, efetua a cada três anos um estudo chamado *EuroTrak* voltado para as melhorias dos seus produtos. O objetivo desse estudo é entender as satisfações e frustrações do ponto de vista de seus clientes (usuários dos aparelhos). No estudo (HEARING..., 2020 (Acessado Out 20, 2021)), é visível o aumento global na procura por aparelhos auditivos, inclusive em pessoas de diferentes faixas etárias conforme apresentado na Figura 1.2. Esses indicadores seguem impulsio-

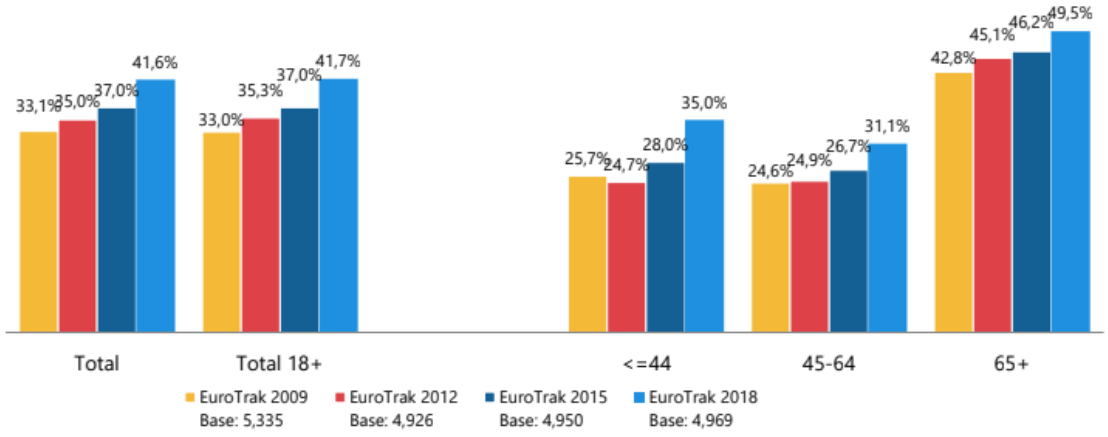
Figura 1.1: Cenário acústico com ruído.



Fonte: Tese de Fábio Itturriet (ITTURRIET, 2019)

nando as pesquisas que visam os aprimoramentos dos aparelhos em busca de qualidade de vida e satisfação de usuários.

Figura 1.2: Crescimento no uso de aparelhos auditivos.

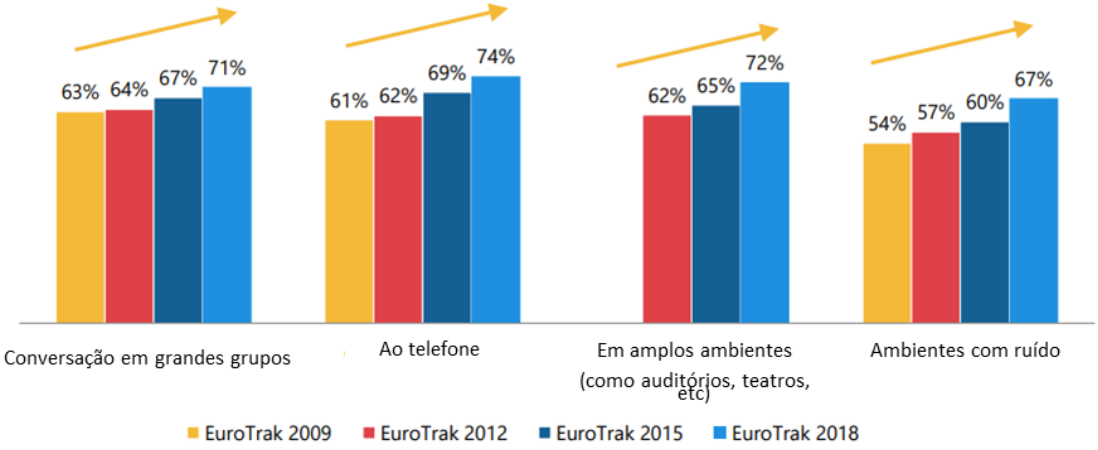


Fonte: Estudo EHIMA 2020

Ainda no mesmo estudo, outro ponto importante foi o levantamento sobre a satisfação dos usuários em cenários acústicos clássicos, conforme mostrado na Figura 1.3. Embora sejam relatadas melhoras ao longo dos estudos em todos os cenários, conversas em ambientes com ruído ainda permanecem como o cenário acústico com pior desempenho. Esse fato destaca a necessidade de técnicas de redução de ruídos cada vez mais efetivas dentro dos aparelhos auditivos.

Com relação ao *hardware* utilizado pelas fabricantes de aparelhos auditivos e implantes cocleares, não há uma divulgação aberta dessas plataformas. Acredita-se que sejam utilizados circuitos dedicados ao invés de processadores digitais de sinais de pro-

Figura 1.3: Satisfação dos usuários em diferentes ambientes acústicos.



Fonte: Estudo EHIMA 2020

pósito geral devido às duras restrições impostas nesse tipo de dispositivo. Os principais desafios de projeto de *hardware* nessa área são:

- **Área reduzida:** Os aparelhos auditivos estão cada vez menores e por isso demandam que *hardware* embarcado dentro do aparelhos tenha dimensões reduzidas.
- **Baixo consumo de potência:** Aparelhos auditivos são dispositivos que funcionam alimentados por baterias. Quanto menor o consumo, maior a duração das baterias.
- **Alto desempenho:** Aparelhos auditivos são dispositivos de tempo real. Necessitam executar uma ampla gama de algoritmos de processamento de fala dentro do espaço de tempo disponível para a frequência de amostragem. Muitos usuários de aparelhos auditivos fazem leitura labial simultaneamente com o áudio entregue pelos aparelhos. Atrasos de processamento acometem a inteligibilidade na comunicação.

Outra possibilidade é a de utilizar o *hardware* elaborado como dispositivo de borda (edge devices). Esse conceito trata de equipamentos eletrônicos que possuem capacidade de coletar informações de seu ambiente por meio de sensores, ou por interface de rede, e de transmitir essas informações para outros dispositivos. Como mostrado em (SHI et al., 2016), essa tecnologia pode ser utilizada em diversos segmentos, como *Cloud Offloading*, onde os dispositivos transferem processamentos computacionais para uma plataforma externa, sendo possível realizar tarefas como análise de vídeos, executar funções referentes a casas e cidades inteligentes, entre outros.

Eles permitem então que, ao receberem informações do dispositivo do usuário, seja efetuado o processamento dos dados conforme necessidade e seja devolvido ao usuário somente o resultado final, barateando o tratamento pesado do lado do usuário. Porém,

existe uma penalidade na latência de resposta ao processamento desses dados, que podem ser diminuídas conforme a velocidade de transmissão de dados pela rede utilizada e também com a distância, em segmentos de rede, do dispositivo de borda para o dispositivo do usuário.

Com o avanço das tecnologias, é cada vez mais habitual encontrar sistemas computacionais embutidos em equipamentos do dia-a-dia, como fones de ouvido, telefones celulares, assistentes eletrônicos, e etc. Com tais avanços, algumas ferramentas surgiram para auxiliar na construção destes sistemas, tendo como objetivo auxiliar na tradução de algoritmos de software para uma determinada especificação em *hardware*.

A síntese de alto nível, ou *High-Level Synthesis* (HLS), é o processo que interpreta um sistema descrito funcionalmente em uma linguagem de alto nível (normalmente C, SystemC, C++ ou Matlab), e que produz uma arquitetura RTL (*Register Transfer Level*) correspondente para implementação em um dispositivo alvo. Tal arquitetura RTL é especificada pela síntese de alto nível por meio de uma linguagem de descrição de *hardware* como VHDL ou Verilog.

Existem no mercado dispositivos que executam as ações descritas pela arquitetura RTL. Dentre elas, destaca-se o uso do FPGA (*Field Programmable Gate Array*), que trata-se de um circuito integrado projetado para ser configurado por um consumidor ou projetista após a fabricação, podendo ser programado inúmeras vezes pelo usuário, além de possuir diversos recursos embutidos.

Portanto, o presente trabalho propõe a implementação do algoritmo, em tempo real, de redução de ruído e da técnica que busca manter as pistas binaurais originais dos sons captadas pelos microfones, tanto de ruído quanto de fala, em circuitos integrados (CI). Será utilizada a ferramenta de síntese de alto nível, tendo como dispositivo alvo o FPGA, analisando os recursos utilizados, tempo de execução, além de métricas objetivas que comprovem a redução de ruído e preservação do espaço acústico. Será explorado a comparação do uso do filtro de redução de ruído com e sem o uso da técnica de preservação do espaço acústico, bem como os impactos do aumento da utilização desta técnica. O objetivo da proposta é fornecer uma implementação em *hardware*, que possibilite a execução destes algoritmos em tempo real por meio de computação de borda, sendo este validado e comprovado a sua funcionalidade.

2 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo, serão apresentados os conceitos teóricos necessários para a compreensão do trabalho proposto. Além disso, serão mostrados os algoritmos utilizados, ferramentas de implementação de *hardware*, métricas de avaliação da solução proposta e trabalhos relacionados.

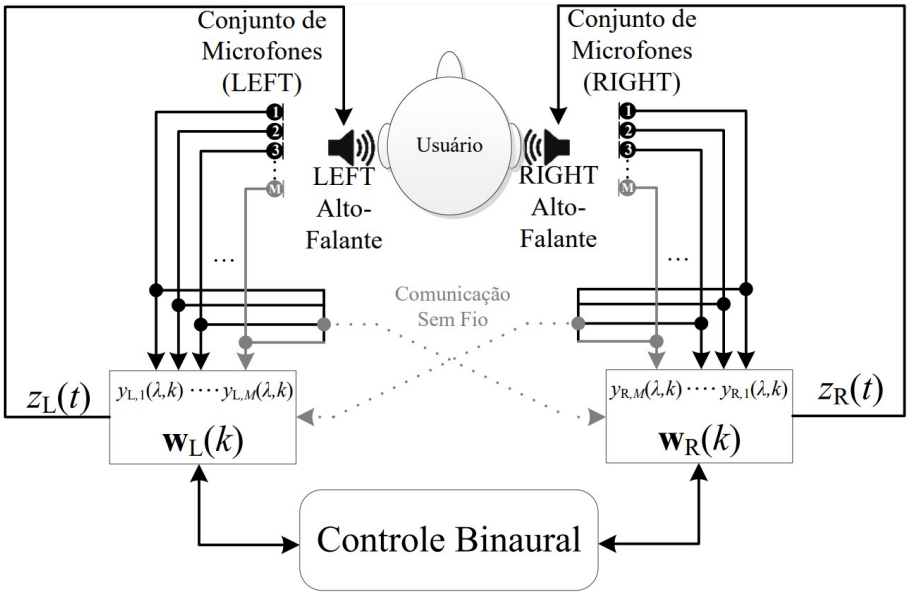
2.1 Filtro de Wiener Multicanal (MWF)

Dentre os diversos tipos de aparelhos auditivos presentes no mercado hoje em dia, o presente trabalho irá utilizar como base os aparelhos binaurais. Esses aparelhos são utilizados em indivíduos que apresentam perdas auditivas em ambas as orelhas. Cada aparelho é composto por um *array* de microfones em diferentes posições visando ampliar o acesso à informação espacial dos cenários acústicos. Sua principal caraterística é o compartilhamento de informações sonoras entre os aparelhos, permitindo assim um processamento cooperativo capaz de preservar as características acústicas originais captadas pelos microfones.

Uma das técnicas de redução de ruído mais utilizadas na literatura para aparelhos auditivos binaurais é o filtro de Wiener Multicanal (MWF - *Multichannel Wiener Filter*), que é extremante eficaz ao extrair o ruído do sinal de interesse (geralmente fala) através das características estatísticas desses sinais (VAN DEN BOGAERT et al., 2009),(CORNELIS; MOONEN; WOUTERS, 2010),(ITTURRIET; COSTA, 2019) e (WERNER; COSTA, 2020). Por se tratar de uma técnica multicanal, ela proporciona redução de ruído superior quando comparada com as técnicas monocanal (DOCLO; MOONEN, 2005). Em contrapartida, essa técnica altera as pistas acústicas das fontes sonoras, e por isso compromete a consciência espacial dos usuários de aparelhos auditivos.

A figura 2.1 apresenta um diagrama em blocos com os principais componentes de um sistema binaural. É composto por 2 aparelhos, um colocado na orelha esquerda contendo M_L microfones, e o outro aparelho colocado na orelha direita contendo M_R microfones. O número total de microfones é definido como $M = M_L + M_R$. Os áudios captados pelos M microfones são agrupados em *frames* ainda no domínio do tempo. Em seguida, os *frames* são transformados para domínio da frequência através da Transformada de Fourier de Tempo Curto (STFT - *Short Time Fourier Transform*) e representados

Figura 2.1: Funcionamento de um aparelho binaural full-duplex.



Fonte: Tese de Fábio Itturriet (ITTURRIET, 2019)

pela letra y. Para cada frame λ, e frequência k, temos a equação 2.1:

$$\mathbf{y}(\lambda, k) = [\mathbf{y}_{\mathrm{M}}(\lambda, k)]^T \tag{2.1}$$

O vetor **y**(λ,k), que representa o vetor de entrada dos sons a cada *frame* por frequência, tem dimensão de Mx1 e pode ser acessado por ambos os aparelhos. O operador $[\cdot]^T$ representa o transposto do vetor em questão. Cada **y**(λ,k) pode ser descrito por **y**(λ,k)=**x**(λ,k)+**v**(λ,k), onde **x**(λ,k) é o sinal de fala, e **v**(λ,k) o ruído presente no sinal de entrada.

Assumindo-se **x**(λ,k) e **v**(λ,k) como vetores aleatórios estacionários de média zero, define-se as matrizes de coerência da fala, ruído e entrada (fala+ruído), respectivamente, pela equação 2.2.

$$\begin{aligned}
\mathbf{\Phi}_x(k) &= E\left\{\mathbf{x}(\lambda, k)\mathbf{x}^H(\lambda, k)\right\}, \\
\mathbf{\Phi}_v(k) &= E\left\{\mathbf{v}(\lambda, k)\mathbf{v}^H(\lambda, k)\right\}, \\
\mathbf{\Phi}_y(k) &= E\left\{\mathbf{y}(\lambda, k)\mathbf{y}^H(\lambda, k)\right\}
\end{aligned} \tag{2.2}$$

De modo que $E\{\cdot\}$ é o valor esperado estatístico em função de l, $(\cdot)^H$ é o hermitiano transposto, $\mathbf{\Phi}_x$ é a matriz de coerência do sinal de fala, $\mathbf{\Phi}_v$ é a matriz de coerência do ruído, $\mathbf{\Phi}_y$ é a matriz de coerência do sinal de entrada. Assumindo-se independência estatística entre os sinais de fala e ruído, tem-se que $\mathbf{\Phi}_y = \mathbf{\Phi}_x + \mathbf{\Phi}_v$.

De forma geral, o microfone frontal de cada aparelho auditivo é designado como microfone de referência. O sinal recebido por esse microfone é dado pela equação 2.3.

$$\begin{aligned} y_{l,m}(\lambda,k) &= x_{l,m}(\lambda,k) + v_{l,m}(\lambda,k), \\ &= \mathbf{q}_l^T \mathbf{y}(\lambda,k) \end{aligned} \tag{2.3}$$

Aonde *m* é o índice correspondente ao microfone de referência, tendo o lado definido por $l = \{L, R\}$, e $\mathbf{q}_l$ é um vetor com valor 1 na posição m e 0 nas demais posições.

O sinal processado no domínio da frequência é dado pela equação 2.4.

$$\mathbf{z}_l(\lambda,k) = \mathbf{w}_l^H(k)\mathbf{y}(\lambda,k) \tag{2.4}$$

No qual $\mathbf{w}_l$ representa o vetor de coeficientes de redução de ruído, $\mathbf{w}_R$ e $\mathbf{w}_L$, cada qual correspondente a um dos aparelhos auditivos. Este sinal resultante então é transformado para o domínio do tempo utilizando-se a transformada inversa de Fourier e um método de reconstrução do tipo *overlap-and-add* (CROCHIERE, 1980), sendo, por fim, devolvido aos alto-falantes do dispositivo.

O conjunto de coeficientes de ambos os aparelhos auditivos pode ser descrito pela equação 2.5.

$$\mathbf{w}(k) = [\mathbf{w}_L^T(k)\mathbf{w}_R^T(k)]^T \tag{2.5}$$

Chegamos então a função custo utilizada pelo filtro MWF, no qual determina um estimador do sinal de fala recebido nos microfones de referência, $x_{L,mL}$ e $x_{R,mR}$, através da minimização da seguinte função apresentada na equação 2.6.

$$J_{MWF}(k, \mathbf{w}_L(k), \mathbf{w}_R(k)) = E\left\{ \left|\left| \begin{array}{l} x_{L,mL}(\lambda,k) - \mathbf{w}_L^H(\mathrm{k})\ \mathbf{y}(\lambda,k) \\ x_{R,mR}(\lambda,k) - \mathbf{w}_R^H(\mathrm{k})\ \mathbf{y}(\lambda,k) \end{array} \right|\right|^2 \right\} \tag{2.6}$$

Os sinais resultantes da filtragem de Wiener multicanal apresentam as pistas acústicas do sinal de fala perfeitamente preservadas, além da redução de ruído. No entanto, as pistas binaurais do sinal de ruído muitas vezes são alteradas, geralmente sendo deslocadas para a posição da fonte de fala (costumeiramente em frente ao usuário), gerando confusões, desconforto e até mesmo situações de perigo (KLASEN, T. J. et al., 2006). Por esse motivo, existem na literatura soluções que buscam operar juntamente com o MWF

visando restaurar a localização original das fontes sonoras.

2.2 Restauração da localização da fonte de ruído

A redução de ruído binaural para aplicações em aparelhos auditivos apresenta duas funções importantes: reduzir ruído e preservar o cenário acústico na forma mais próxima possível do original. Como destacado na seção anterior, o MWF modifica as pistas binaurais do ruído residual, o que acarreta na mudança subjetiva da posição da fonte de ruído.

Para compreender esse fenômeno, é necessário entender como o cérebro localiza as fontes sonoras direcionais. A localização de fontes ocorre por meio das pistas acústicas extraídas das diferenças do campo sonoro em cada orelha. Tais diferenças se dão por modificações que sofrem os sinais que chegam até as orelhas em relação a fonte do som. Essas modificações são compostas por atrasos e atenuações gerados pelo tamanho da cabeça humana que reflete e refrata ondas sonoras, efeito conhecido como *head shadow*. Quando um som está localizado perfeitamente a frente da cabeça, como podemos ver na Figura 2.2, ambas as orelhas receberão o sinal no mesmo instante de tempo e com a mesma intensidade. À medida que a fonte se desloca ao redor da cabeça, o som chegará até a orelha mais distante com um atraso em relação a orelha mais próxima, introduzindo uma diferença de tempo interaural (ITD), além da diferença de nível interaural (ILD).

Figura 2.2: Som no azimute 0º, perfeitamente à frente do indivíduo.

Fonte: Artigo de Wayne Staab - *Localization: More Important Than Word Recognition?*

Dentro das pesquisas da área sobre preservação das pistas, a teoria duplex (R.S., 1907) sustenta que a ILD é a principal pista binaural usada pelos humanos para sons de alta frequência (acima de 1,5kHz), e que abaixo dessa frequência, tais diferenças não são

relevantes, levando a utilização da ITD para preservação nessa faixa de frequência. Porém, existem pesquisas que avaliam a influência da ILD no processo de lateralização dos sons em frequências abaixo de 1,5kHz (HARTMANN; RAKERD et al., 2016) (HARTMANN; MACAULAY, 2014), no qual conclui que ela é capaz de inverter o hemisfério de sinais acústicos com frequências pertencentes à banda da ITD, demonstrando que a teoria duplex não é uma regra que estabelece que apenas uma pista comanda individualmente cada banda.

Percebe-se então que na literatura, ainda não há uma regra plenamente estabelecida para este fator. Foi apresentada uma técnica de preservação das pistas binaurais da (única) fonte interferente de ruído através da preservação da função de transferência interaural ITF (*Interaural Transfer Function*)(DOCLO; SPRIET et al., 2005). A ITF é composta intrinsecamente pela diferença de tempo interaural (ITD) e pela diferença de nível interaural (ILD) tornando-a uma solução robusta e aplicável em todas as faixas de frequência. A função custo de preservação da localização da fonte de ruído, que é baseada no erro quadrático médio entre a ITF do ruído captado pelos microfones (ITF_{in}) e a ITF do ruído processado (ITF_{out}), é apresentada na equação 2.7.

$$\begin{aligned} J_{ITF}(k) &= E\left\{||ITF_{\text{Out}}(\lambda,k) - ITF_{\text{In}}(\lambda,k)||^2\right\} \\ &= E\left\{\left|\left|\ \frac{\mathbf{w}_L^H(k)\mathbf{v}(\lambda,k)}{\mathbf{w}_R^H(k)\mathbf{v}(\lambda,k)} - \frac{\mathbf{q}_L^T\mathbf{v}(\lambda,k)}{\mathbf{q}_R^T\mathbf{v}(\lambda,k)}\ \right|\right|^2\right\} \end{aligned} \tag{2.7}$$

Como apresentado no mesmo trabalho, a função custo total proposta (J_T) mostrada na equação 2.8 é composta pela soma da função custo do MWF mostrada na equação 2.6 com a função custo da ITF mostrada na equação 2.7. A variável γ é um fator de ponderação da função do custo da ITF dentro do cenário global, permitindo uma relação direta entre redução de ruído e preservação da localização da fonte de ruído.

$$J_T(k) = J_{MWF}(k) + \gamma J_{ITF}(k) \tag{2.8}$$

Todos os trabalhos supracitados nesse capítulo realizam o processamento prévio dos coeficientes dos filtros através de um algum método de otimização, para numa segunda etapa, o processo de filtragem ser realizado. O principal objetivo desses trabalhos foi propôr diferentes métodos (funções custo) de preservação de cenários acústicos. Por isso, essas implementações não operam em tempo-real, o que acaba limitando sua aplicação em cenários reais.

2.3 Solução adaptativa proposta

Em (REYS et al., 2019), é apresentado uma implementação recursiva da solução apresentada na equação 2.8. Como o objetivo primordial de J_T é a redução de ruído do sinal, espera-se que a constante γ na equação 2.8 deva possuir valor relativamente pequenos, com o intuito de que a convexidade de J_{MWF} seja assumida como preponderante sobre J_{ITF}.

Sendo assim, é possível aplicar o método do gradiente descendente sobre a superfície da função custo da equação 2.8, permitindo o cálculo iterativo dos coeficientes conforme descrito na equação 2.9.

$$\mathbf{w}_{n+1} = \mathbf{w}_n - \frac{\mu}{2}\nabla_{\mathbf{w}}.J_T \tag{2.9}$$

A variável μ é uma constante de adaptação que controla a velocidade de convergência e a estabilidade do algoritmo. No mesmo trabalho, após algumas derivações matemáticas, foi demonstrada a equação adaptativa na qual baseia-se a implementação proposta neste trabalho, apresentada na equação 2.10.

$$\begin{aligned}\mathbf{w}_{n+1} =&(\mathbf{I} - \beta\mathbf{\Phi}_{yy})\mathbf{w}_n + \beta\boldsymbol{\phi}_x \\ &+ \gamma\left(\frac{(\mathbf{w}^H\mathbf{\Phi}_{vr}\mathbf{w})\mathbf{\Phi}_{vt}\mathbf{w} - (\mathbf{w}^H\mathbf{\Phi}_{vt}\mathbf{w})\mathbf{\Phi}_{vr}\mathbf{w}}{(\mathbf{w}^H\mathbf{\Phi}_{vr}\mathbf{w})^2}\right)\end{aligned} \tag{2.10}$$

Sendo que β é o fator utilizado pela função custo do MWF, γ o fator de ponderação de utilização da técnica ITF e $\mathbf{I}$ denota a matriz identidade de dimensão 2Mx2M.

Para ser possível chegar ao valor proposto pela equação 2.10, algumas definições

precisam ser mencionadas, como as seguintes:

$$\begin{aligned}
\phi_x &= \begin{bmatrix} \mathbf{\Phi}_x & \mathbf{q}_L \\ \mathbf{\Phi}_x & \mathbf{q}_R \end{bmatrix}, \\
\mathbf{\Phi}_{yy} &= \begin{bmatrix} \mathbf{\Phi}_y & 0 \\ 0 & \mathbf{\Phi}_y \end{bmatrix}, \\
\mathbf{\Phi}_{vt} &= \begin{bmatrix} \mathbf{\Phi}_v & -ITF_{in}^{*}\mathbf{\Phi}_v \\ -ITF_{in}\mathbf{\Phi}_v & |ITF_{in}|^2\,\mathbf{\Phi}_v \end{bmatrix}, \\
\mathbf{\Phi}_{vr} &= \begin{bmatrix} 0 & 0 \\ 0 & \mathbf{\Phi}_v \end{bmatrix},
\end{aligned} \tag{2.11}$$

sendo 0 uma matriz de zeros de dimensão 2*M*x2*M*, ϕ_x a potência dos microfones de referência esquerdo e direito, Φ_{yy} e Φ_{vr} expansões necessárias para manipulação das matrizes de entrada e ruído, e Φ_{vt} uma matriz que utiliza a ITF_{in} juntamente com a matriz de ruído.

As matrizes de coerência, também necessárias para o cálculo da equação adaptativa, são obtidas a partir das seguintes equações:

$$\begin{aligned}
&\hat{\mathbf{\Phi}}_y(\lambda,k) \\
&= \begin{cases} \eta_y\hat{\mathbf{\Phi}}_y(\lambda-1,k) + (1-\eta_y)\big(\mathbf{y}(\lambda,k)\mathbf{y}^H(\lambda,k)\big), & VAD=1, \\ \hat{\mathbf{\Phi}}_y(\lambda-1,k), & VAD=0 \end{cases}
\end{aligned} \tag{2.12}$$

$$\begin{aligned}
&\hat{\mathbf{\Phi}}_v(\lambda,k) \\
&= \begin{cases} \hat{\mathbf{\Phi}}_v(\lambda-1,k), & VAD=1, \\ \eta_v\hat{\mathbf{\Phi}}_v(\lambda-1,k) + (1-\eta_v)\big(\mathbf{y}(\lambda,k)\mathbf{y}^H(\lambda,k)\big), & VAD=0 \end{cases}
\end{aligned} \tag{2.13}$$

$$\hat{\mathbf{\Phi}}_x(\lambda,k) = \eta_x\hat{\mathbf{\Phi}}_x(\lambda-1,k) + (1-\eta_x)\big(\hat{\mathbf{\Phi}}_y(\lambda,k) - \hat{\mathbf{\Phi}}_v(\lambda,k)\big) \tag{2.14}$$

Sendo η_y, η_v e η_x os fatores de esquecimento para a estimação das matrizes. VAD (*Voice Activity Detection*) refere-se ao algoritmo que indica se existe a presença de voz ou apenas ruído no *frame* analisado (RAMÍREZ; GORRIZ; SEGURA, 2007). O valor de saída corresponde a 0 caso contenha apenas ruído, onde somente a matriz de coerência do

ruído é atualizada, ou 1 caso contenha voz, onde somente a matriz de coerência de entrada é atualizada.

Com isso, será utilizada neste trabalho a solução adaptativa que utiliza o algoritmo de redução de ruído MWF, juntamente com a técnica de preservação do cenário acústico ITF, sendo chamada de AMWF-ITF.

2.4 Revisão bibliográfica

Existem hoje na literatura alguns trabalhos que utilizam filtros de redução de ruído, sendo um dos mais utilizados o filtro Wiener Multicanal (MWF – *Multichannel Wiener Filter*) que, como mostra em (WIENER, 1964), por ser uma técnica multicanal, proporciona uma significativa redução de ruído ao mesmo tempo em que preserva as pistas acústicas da fonte de interesse (fala). No entanto, as informações binaurais das demais fontes acústicas (ruído) são alteradas. Em cenários com uma fonte sonora interferente pontual, a posição percebida é deslocada para a posição da fonte de fala (geralmente em frente ao usuário), o que pode acarretar desconforto, confusão ou mesmo situações de perigo.

Em (KLASEN, T. et al., 2006) é apresentada uma proposta de método de redução de ruído visando a preservação das pistas binaurais (ITD e ILD) da (única) fonte interferente de ruído através da preservação da função de transferência interaural (ITF - *Interaural Transfer Function*). Essa proposta consiste na minimização de uma função custo definida pela soma ponderada entre a função custo do MWF e uma função associada ao descasamento entre a ITF do ruído captado pelos microfones e a do ruído processado. Como grande desvantagem, essa técnica apresenta alto custo computacional em função de se apresentar como um problema de otimização não convexa.

Em (DOCLO; SPRIET et al., 2005), foi apresentada outra forma de abordagem que visa a preservar a função de transferência do caminho acústico entre a fonte de ruído e cada uma das orelhas, conhecida como função de transferência interaural (ITF – do inglês Interaural Transfer Function).

Em (REYS et al., 2019), trata da implementação recursiva do método de redução de ruído MWF, juntamente com a técnica de preservação do cenário acústico ITF, sendo possível a ponderação da técnica na função custo total.

Em (CARMO; COSTA, 2018), foi apresentado um método de aproximação online para a técnica de redução de ruído multicanal do filtro Wiener (MWF) com preservação

da diferença de nível interaural (ILD) de ruído para aparelhos auditivos binaurais.

Em (ITTURRIET, 2019) é abordada a utilização das técnicas de redução de ruido, com a preservação perceptualmente relevante da diferença de tempo interaural em aparelhos auditivos, avaliando a eficácia do método original de preservação da ITD através de experimentos psicoacústicos. Dentre os experimentos apresentados, a técnica de preservação do cenário acústico ITF apresenta os melhores resultados quanto a lateralização, uma vez que a ITF preserva ambas as pistas simultaneamente, tanto ILD quanto ITD.

2.5 Funcionamento da síntese de alto nível

Neste trabalho foi utilizado a ferramenta da Xilinx, Vitis High-Level Synthesis (HLS) versão 2021.1, que transforma uma especificação em linguagem alto-nivel em uma implementação de nível de transferência de registro (RTL) que pode ser sintetizada em um FPGA Xilinx. A especificação pode ser descrita em linguagens de programação C, C++ e System C. O FPGA tem uma arquitetura paralela e oferece benefícios em termos de desempenho, custo e consumo de energia em comparação com os processadores tradicionais. O uso do HLS para projetistas de *hardware* oferece benefícios, como um maior nível de abstração na criação de *hardware* de alto desempenho.

A validação e correção do projeto funcional são realizadas mais rapidamente do que com a linguagem de *hardware* tradicional, como VHDL ou Verilog. O uso de diretivas de otimização facilita criar implementações específicas de *hardware* de alto desempenho com possibilidade de utilizar no mesmo código C diferentes diretivas para uma implementação otimizada. O processo de aplicação de diretivas de otimização é realizado dependendo da descrição do código C, C++ e suas características. É necessário observar o que se destina ao projeto final antes de aplicá-las. Algumas métricas de análise importantes são: os ciclos de relógio, latência e recursos utilizados pelo circuito. Como descrito por (COUSSY et al., 2009), as fases de uma ferramenta HLS são as seguintes:

1. Compilação de especificação.
2. Alocação de recursos de *hardware*.
3. Mapeamento das instruções por ciclo de clock.
4. Conectar cada instrução à instrução funcional que a executará.
5. Gerar a arquitetura RTL.

Seguindo tais etapas é possível criar sistemas complexos em linguagem de má-

quina, podendo assim serem programados em aceleradores como FPGA.

2.6 Métricas objetivas de avaliação de áudio

As métricas utilizadas no projeto foram escolhidas com intuito de demonstrar numericamente a qualidade e inteligibilidade do áudio filtrado, além de quantificar a redução de ruído atingida, bem como a preservação do espaço acústico.

2.6.1 Redução de ruído

Em comunicações analógicas e digitais, uma relação sinal-ruído SNR (*Signal-Noise Ratio*) (KIESER; REYNISSON; MULLIGAN, 2005) é uma razão da potência do sinal desejado em relação ao ruído de fundo (sinal indesejado). A fórmula $SNR = S/N$, onde S é a potência do sinal de fala de entrada e N é a potência do sinal ruidoso, pode ser utilizada para comparar os dois níveis e retornar a proporção, que mostra se o nível de ruído está impactando o sinal desejado.

A razão é normalmente expressa como um único valor numérico em decibéis (dB). A proporção pode ser zero, um número positivo ou um número negativo. Uma relação sinal-ruído positiva indica que o nível do sinal de interesse é maior do que o nível do ruído. Valores abaixo de 0 dB indicam que a potência do ruído é maior do que a da fala.

Para realizar o cálculo da relação, extraindo o valor em decibéis, utilizamos a fórmula 2.15:

$$SNR_l = 10 * log10(S_l/N_l) \tag{2.15}$$

Onde *l* indica em qual lado está sendo calculada a relação, orelha direita ou orelha esquerda. Para demonstrar a variação obtida no sinal após a filtragem, foi calculado um ΔSNR comparando o áudio de entrada com o áudio filtrado pela equação 2.16:

$$\Delta SNR_l = SNR_{out,l} - SNR_{in,l} \tag{2.16}$$

Os resultados da equação 2.16 podem ter valores maiores ou menores que 0, no qual valores positivos indicam, em dB, o aumento da diferença de potência do sinal de fala para o sinal de ruído, o que gera áudios mais satisfatórios. Valores negativos sugerem

que a relação para o áudio de saída obteve uma piora, contendo um ruido com potência mais próxima ou maior que o sinal de fala.

2.6.2 Qualidade da fala processada

A qualidade é um dos diversos atributos relacionados com sinais de fala que permite avaliar a distorção gerada no processo de redução de ruído. Possui uma natureza subjetiva, pois depende da avaliação de pessoas previamente selecionadas (avaliadores) com diferentes conceitos de quão "boa" ou "ruim" é a qualidade da fala analisada, resultando em medidas com alta variabilidade entre os avaliadores. A qualidade é uma medida que acessa "como" um locutor produz uma determinada locução, abrangendo fatores como rouco, áspero, entre outros (LOIZOU, 2013). Um exemplo de métrica subjetiva de qualidade é o chamado *Mean Opinion Score* (MOS) (ITU, 2016), que resulta em um valor numérico numa escala entre 1 (ruim) e 5 (excelente) informado pelos avaliadores em relação aos experimentos avaliados. O processo de avaliação subjetiva de qualidade demanda tempo (grande número de avaliadores), infraestrutura e procedimentos experimentais previamente aprovados pelo comitê de ética da instituição envolvida. Por esse motivo, métricas objetivas de avaliação de qualidade se tornam uma interessante alternativa visando agilizar esse procedimento. Um padrão internacionalmente reconhecido intitulado PESQ (*Perceptual Evaluation of Speech Quality*) é usado para medir a qualidade de áudio levando em consideração parâmetros como atrasos variáveis, ruído na linha e corte de áudio. O PESQ fornece uma correlação significativamente maior com a métrica subjetiva MOS do que outros modelos conhecidos na literatura como P.861 (RIX et al., 2001), PSQM (BEERENDS; STEMERDINK, 1994) e MNB (VORAN, 1999).

Como resultado, o PESQ nos fornece um resultado de -0,5 a 4,5, onde quanto mais próximo de -0,5 for o resultado, pior é a qualidade do áudio, e quanto maior, melhor será a qualidade do áudio. O algoritmo utilizado para a computação da métrica está atualmente em código aberto, e pode ser baixado através do link <https://ecs.utdallas.edu/loizou/speech/software.htm>.

2.6.3 Inteligibilidade da fala processada

A métrica STOI (*Short-Time Objective Intelligibility*), apresenta alta correlação com predições de inteligibilidade em usuários que utilizam aparelhos auditivos. A inteligibilidade da fala pode ser definida como o grau com o qual a mensagem do falante pode ser decodificada pelo ouvinte (KENT et al., 1989). Em outras palavras, refere-se à facilidade com que o ouvinte é capaz de entender a fala de seu interlocutor.

O algoritmo se trata, como vemos em (TAAL et al., 2010), de uma medida de inteligibilidade do sinal, que está altamente relacionado com a inteligibilidade de sinais de fala degradados por algum fator, como por exemplo devido a ruído aditivo ao sinal de interesse.

A métrica está atualmente em código aberto, sendo possível realizar o download através do link <https://ceestaal.nl/code/>.

O resultado do algoritmo é um valor entre 0 e 1, onde 1 representa um áudio com inteligibilidade máxima, e quanto mais próximo de 0, menor a inteligibilidade do áudio analisado.

2.6.4 Lateralização

Para avaliar se o som de saída teve a lateralização da fonte sonora modificado, foram utilizados as variações de IPD (*Interaural Phase Delay*) e ILD em cada frame. Lateralização refere-se a posição virtual do som recebido dentro da cabeça para estímulos aplicados com fones de ouvido, diferentemente do termo de localização de uma fonte sonora, que é a habilidade do ser humano de determinar a posição de um fonte sonora em um campo aberto (sem fones de ouvido) (PLENGE, 1974).

A preservação da pista de ITD da fonte de ruído no domínio da frequência é alcançada através da preservação da IPD em cada bin de frequência, na qual baseia-se na diferenças de fase para cada frequência. Como está sendo utilizado a técnica de preservação de lateralização ITF, que utiliza na sua composição tanto a ITD quanto ILD, as variações de ambos foram calculadas em todas as bandas de frequência.

A variação de IPD ou ΔIPD é calculado a partir da equação 2.17:

$$\Delta IPD = \sum_{k=1}^{\mathrm{K}} \left| \frac{IPD_{out}(k) - IPD_{in}(k)}{\pi} \right| \tag{2.17}$$

Onde $IPD_{in}(k)$ e $IPD_{out}(k)$ representam, respectivamente, as IPDs dos sinais de entrada e saída para todos os bins k. Quanto mais próxima do valor 0 resultar a variação, menor será a diferença na pista de ITD para o sinal de saída.

A variação de ILD ou ΔILD é calculado a partir da diferença entre a ILD do sinal conhecido introduzido aos microfones de referência e a ILD do sinal filtrado, que chega aos alto-falantes dos aparelhos auditivos. A variação de ILD é calculado a partir da equação 2.18:

$$\Delta ILD = \sum_{k=1}^{\mathrm{K}} |10 \log_{10} ILD_{out}(k) - 10 \log_{10} ILD_{in}(k)| \tag{2.18}$$

Onde $ILD_{in}(k)$ e $ILD_{out}(k)$ representam, respectivamente, as ILDs dos sinais de entrada e saída para todos os bins k. A variação, o quanto mais próxima do valor 0 representa que o sinal de saída não obteve alterações na ILD resultante.

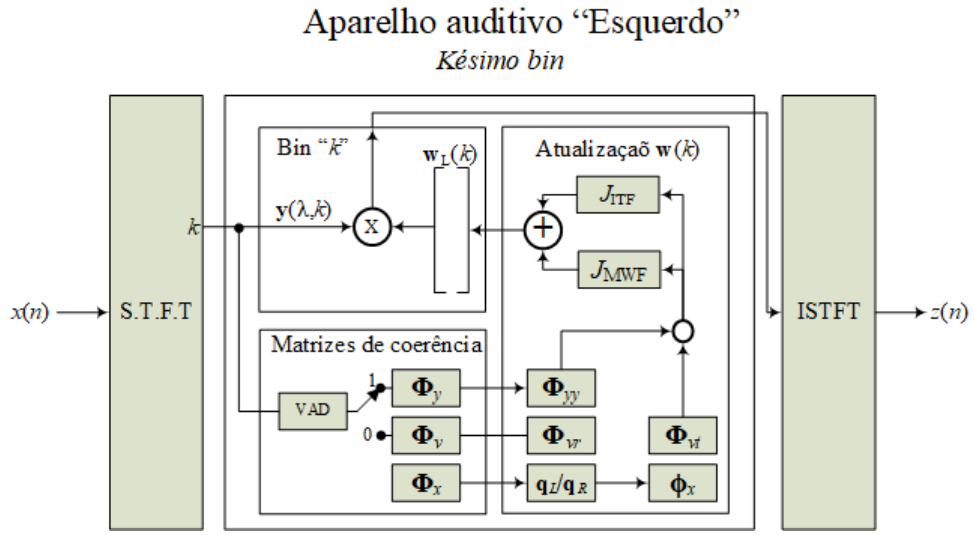
3 METODOLOGIA E IMPLEMENTAÇÃO

Neste capítulo serão discutidas quais metodologias foram utilizadas dentro da implementação do projeto, apresentando a organização das estruturas internas do dispositivo.

Para o presente trabalho, foi disponibilizado um código referência descritivo, em linguagem Matlab, pelo Professor Fábio Pires Itturriet, onde nele é realizada a redução de ruído dos sinais, e utilizada a técnica de preservação do cenário acústico ITF. Existia a possibilidade de utilizar a ferramente do Matlab HDL Coder, que realiza a tradução para RTL, porém não era possível a utilização devido a falta de licença da ferramenta. Portanto, a partir da implementação em Matlab, foi elaborado o código em linguagem C++, para que este fosse introduzido na ferramenta de síntese de alto nível, Vitis® HLS desenvolvida pela Xilinx com o objetivo de fornecer suporte para o desenvolvimento de circuitos em FPGAs Xilinx. A ferramenta permite que a especificação funcional de um sistema em alto nível (C/C++) seja usada para a produção de um circuito em nível Register Transfer Level (RTL), sem a necessidade de fazê-lo manualmente.

O código gerado foi produzido contendo a separação em blocos projetada para o circuito, por meio de funções que são chamadas pela entidade topo. Na figura 3.1 podemos observar de modo geral a separação em blocos do circuito, sendo esse um padrão neste tipo de aparelho, que será explicada a seguir.

Figura 3.1: Implementação exemplificada em blocos.



Fonte: Arquivo pessoal

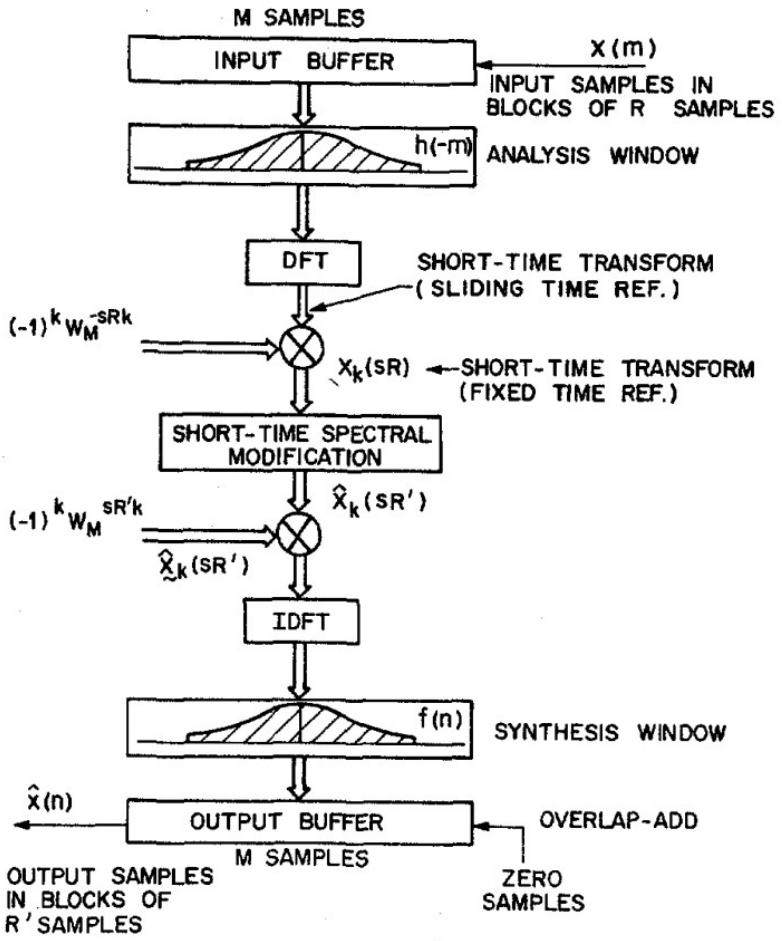
Para o presente trabalho será utilizado arbitrariamente como base o uso de 6 microfones, sendo 3 do lado direito e 3 do lado esquerdo. Os sons captados pelos microfones são armazenados por amostra em memórias do tipo BRAM, com uma frequência de

amostragem de 16kHz.

Devido ao comportamento não estacionário dos sinais de fala, a DFT (Transformada discreta de Fourier) não é recomendada neste caso por demandar longas janelas de análise, que acabam por omitir importantes transições no conteúdo espectral dos sinais. Por este fato, será utilizada a STFT, pois aplica a DFT em curtos períodos de tempo, nos quais os sinais de fala podem ser considerados localmente estacionários.

Muito utilizado em filtragem de sinais de fala, a técnica de *Weighted overlap-and-add* (BRILLINGER, 2001) (CROCHIERE, 1980) é amplamente utilizada para este fim, onde basicamente trata-se da análise "bloco a bloco" do sinal. No sinal de entrada é realizado o "janelamento" no tempo e, com isso, feita a sobreposição de segmentos recebidos, de duração finita. Em cada segmento é então aplicada a transformada de Fourier para obter um espectro de curto prazo. A figura 3.2 demonstra o funcionamento da técnica.

Figura 3.2: Implementação da técnica de *Weighted overlap-and-add*.



Fonte: (CROCHIERE, 1980)

Para o presente trabalho, foram utilizadas 256 amostras por análise, com blocos de 64 amostras, equivalente a 4μ segundos, que aqui chamaremos de *frame*. Nele é aplicada

uma janela de análise de 50%, sendo assim com tamanho de 128. A aplicação da técnica no código implementado pode ser observada na figura 3.3.

Figura 3.3: Uso da STFT na implementação.

```
81      for (int j = 0; j < 256; j++)
82          for (int i = 0; i < channel; i++)
83              frameMultOverlapAdd[i][j] = weightingOverlapAdd[j] * frame_in_tmp_in[j][i];
84          //Multiplica o frame de entrada pela janela de analise
85
86      for(int ch = 0; ch < channel; ch++)
87      {
88          for (int i = 0; i < 256; i++)
89              fft_input[i] = std::complex<float>((float)frameMultOverlapAdd[ch][i], 0);
90          //Resultado da multiplicação, colocado em numero complexo
91
92          bool ovflo;
93
94          fft_top(1, fft_input, fft_output, &ovflo);
95          //FFT
96
97          for (int i=0; i<256; i++)
98              frame_in_frq_in[i][ch] = fft_output[i] * phaseMod[i];
99          //Modificação de fase proposto pelo método weighted overlap-and-add
100     }
```

Fonte: Arquivo Pessoal

Na figura 3.3, é primeiramente realizada a multiplicação do frame de entrada pela janela de análise, como vemos na linha 83. Para multiplicações de matrizes em C++, é necessária a utilização de laços que percorram todas as células, que nesse caso é análoga a utilização de vetores de vetores. Após, é realizada a FFT (*Fast Fourier Transform*) das 256 amostras, algoritmo eficiente para calcular a DFT, que nesse caso será realizado uma adaptação para que seja performada a STFT. Após esta etapa é aplicada a modificação de fase linear, já no domínio da frequência, vista na linha 98. Este processo ocorre para cada microfone.

A Xilinx disponibiliza nativamente na sua plataforma um IP (*Intellectual Property*) que realiza tanto a operação de transformada rápida de Fourier, quanto a inversa da transformada, que será utilizada neste trabalho (XILINX, s.d.). Ele possibilita a instanciação do componente efetuando chamadas diretamente no C++. O IP possui diversas opções de configuração, como tamanho da FFT, tamanho dos dados de entrada e saída, tipo de dados, como ponto flutuante ou ponto fixo, entre outros. Neste trabalho foi utilizado tamanho da FFT com valor de 256, e a opção de ponto flutuante, sendo 16 bits para parte real e 16 bits para parte imaginária, para uma melhor precisão dos resultados. A escolha pelo ponto flutuante se deu por conta de encontrar dificuldades na configuração do IP da Xilinx com ponto fixo. Não foram exploradas outros tipos de dados de entrada e saída, que provavelmente irão refletir nos resultados encontrados de síntese e precisão. O IP permite ainda definir algumas configurações em tempo de execução, com as quais é possível passar por parâmetro algumas informações que modificam o modo de uso da FFT, como optar por realizar a transformada inversa.

Com o *frame* de entrada no domínio da frequência, é realizado a atualização das matrizes de coerência, demonstrado nas equações 2.12, 2.13 e 2.14. Este processo ocorre somente nos *frames* que tiverem energia maior que zero. Para efetuar a atualização, o processo necessita receber o sinal de VAD para o *frame*, que indica qual matriz deverá ser atualizada, sendo 1 para matriz de coerência da entrada e 0 para matriz de coerência do ruído, como vemos na figura 3.1, na caixa de "Matrizes de coerência". No presente trabalho este algoritmo não foi implementado em *hardware*, e portanto, o *frame* de entrada é analisado previamente, sendo o resultado carregado a cada iteração. O algoritmo VAD utilizado para realizar a análise prévia pode ser encontrado neste link <http://www.ee.ic.ac.uk/hp/staff/dmb/voicebox/voicebox.html>.

Para melhor convergência do filtro, é adotado um limite inicial de *frames* no qual apenas as matrizes de coerência são atualizadas, contendo apenas ruído. Dentro desta faixa de tempo o sinal de saída é igual ao de entrada. A partir de alguns testes realizados foi escolhido o valor de 1 segundo, equivalente a 250 *frames*, para esta faixa inicial.

Figura 3.4: Atualização das matrizes de coerência.

```
 99      if (ct_frame <= limiteFrames) //Verifica o numero de frames, se não atingiu o limite, atualiza somente matrizes de correlação
100      {
101          std::complex<float> sumFrqIn = 0;
102          for (int i=0; i<length_fft; i++)
103              for (int j=0; j<6; j++)
104                  sumFrqIn = sumFrqIn + frame_in_frq_in[i][j];
105              // Soma o valor total das frequências no frame
106
107          //Verifica se não é um frame sem energia, ou seja, sem nenhum áudio.
108          if (sumFrqIn.real() != zeroComplex.real() || sumFrqIn.imag() != zeroComplex.imag())
109              corr_matrix_estimation(0, length_fft, channel, frame_in_frq_in, rv, ry, rx, Vad, conty, contv, lambda);
110
111          for (int i=0; i<length_fft; i++)
112          {
113              std::complex<float> soma_linha_0(0,0);
114              std::complex<float> soma_linha_1(0,0);
115              for (int j=0; j<channel; j++)
116              {
117                  soma_linha_0 = soma_linha_0 + (frame_in_frq_in[i][j] * qL[j]);
118                  //Multiplicação para obter apenas a saída do microfone de referência esquerdo
119                  soma_linha_1 = soma_linha_1 + (frame_in_frq_in[i][j] * qR[j]);
120                  //Multiplicação para obter apenas a saída do microfone de referência direito
121              }
122              frame_out_frq_in[i][0] = soma_linha_0;
123              frame_out_frq_in[i][1] = soma_linha_1;
124          }
125      }
```

Fonte: Arquivo Pessoal

A figura 3.4 demonstra essa abordagem, onde na linha 99 é conferido se o *frame* analisado está dentro da faixa inicial, e, caso esteja, é realizado a atualização das matrizes de coerência, executado na linha 109, para os *frames* que possuam energia maior que 0, ou seja, possuam áudio. Essa atualização ocorre dentro da função criada em C++ chamada de "corr_matrix_estimation", que recebe por parâmetro, em ordem, a indicação se está dentro do limite inicial de *frames*, o tamanho da FFT, o numero de microfones, o *frame* de entrada no domínio da frequência, as matrizes de coerência do *frame* anterior, o sinal recebido do VAD, dois contadores para suavizar as matrizes após alcançar o limite inicial

de *frame* e o η de suavização. Entre as linhas 111 e 124 ocorre o mesmo processo descrito na equação 2.3, onde o microfone de referência de cada lado é guardado.

Quando o limite inicial é atingido, se dá início de fato a filtragem do sinal, onde a cada *frame* é realizada, além da atualização das matrizes de coerência, a renovação dos coeficientes do filtro, apresentada na equação 2.10, que ocorre dentro da função chamada de "update_amwf_itf". A função recebe por parâmetro, em ordem, γ, β, vetor de coeficientes do filtro adaptativo AMWF-ITF anterior, as matrizes de coerência atualizadas, o tamanho da FFT, e os vetores que indicam os microfones de referência, retornando pela variável "awmf_itf"o vetor de coeficientes atualizado.

Figura 3.5: Atualização dos coeficientes do filtro adaptativo (AMWF-ITF).

```
139     else
140     {
141         corr_matrix_estimation(1, length_fft, channel, frame_in_frq_in, rv, ry, rx, sumVad, conty, contv, lambda);
142         //Atualização das matrizes de coerência
143         update_amwf_itf(gama, beta, amwf_itf, rv, ry, rx, 256, qL, qR);
144         //Atualização dos coeficientes do filtro adaptativo AMWF-ITF
145
146         for (int i=0; i<length_fft; i++)
147         {
148             std::complex<float> soma_linha0(0,0);
149             std::complex<float> soma_linha1(0,0);
150             for (int j=0; j<channel; j++)
151             {
152                 soma_linha0 = soma_linha0 + (frame_in_frq_in[i][j] * std::conj(amwf_itf[j][i]));
153                 //Multiplicação do sinal de entrada de cada microfone pelos coeficientes do filtro adaptativo AMWF-ITF(1->6)
154                 soma_linha1 = soma_linha1 + (frame_in_frq_in[i][j] * std::conj(amwf_itf[j+channel][i]));
155                 //Multiplicação do sinal de entrada de cada microfone pelos coeficientes do filtro adaptativo AMWF-ITF(7->12)
156             }
157             frame_out_frq_in[i][0] = soma_linha0;
158             frame_out_frq_in[i][1] = soma_linha1;
159         }
160     }
```

Fonte: Arquivo Pessoal

Neste momento, as amostras de todos microfones são utilizadas para filtragem, na qual são multiplicadas pelo filtro atualizado, como demonstrado na equação 2.4. A figura 3.5 mostra as chamadas realizadas na função principal.

Muito frequente na realização deste trabalho, ao realizar a tradução de código da linguagem Matlab para C++, pequenos passos são feitos em um número muito maior de linhas, de modo que era necessário muita atenção neste procedimento. A figura 3.6 demonstra o código para atualizar a função custo de MWF em Matlab, enquanto a figura 3.7 exemplifica o código resultante em linguagem C++.

Com o sinal filtrado no domínio da frequência, agora é necessário realizar a transformada inversa de Fourier, para devolver ao domínio tempo e ser enviado aos alto-falantes do aparelho, finalizando assim o ciclo para um *frame* do som captado. Este processo pode ser visto na figura 3.8. Entre as linhas 201 e 203 é aplicado sobre o sinal filtrado a modificação de fase, proposto pela técnica *Weighted overlap-and-add*. Com isso, o sinal passa pela transformada inversa de Fourier, sendo necessário realizar a normalização do sinal resultante, devido ao fato que o IP utilizado da Xilinx realiza esse

Figura 3.6: Código Matlab para atualizar função custo MWF.

```
11  for bin = 1:M/2+1
12
13      Ry = RY(:,:,bin);
14      Rx = RX(:,:,bin);
15      Rv = RV(:,:,bin);
16
17      Rxx = [Rx, Zm; Zm, Rx];
18      Ryy = [Ry, Zm; Zm, Ry];
19
20      vl = af_paramt.Beta*Rxx*q;
21      Ml = Imm - (af_paramt.Beta*Ryy);
22
23      % MWF part and previous coefficients (recursion)
24      w_MWF = Ml*w(:,bin) + vl;
25      if( af_paramt.Gamma == 0)
26
27          w(:,bin) = w_MWF;
```

Fonte: Código fornecido pelo Prof. Fábio Itturriet

Figura 3.7: Código C++ para atualizar função custo MWF.

```
51  for (int bin=0; bin<(m/2+1); bin++)
52  {
53      for (int i=0; i<ch; i++)
54          for (int j=0; j<ch; j++)
55          {
56              ry_y[i][j]=ry[i][j][bin];
57              rx_x[i][j]=rx[i][j][bin];
58              rv_v[i][j]=rv[i][j][bin];
59          }
60
61      for (int i=0; i<ch*2; i++)
62          for (int j=0; j<ch*2; j++)
63              if(i < ch)
64                  if(j < ch)
65                  {
66                      rxx[i][j] = rx_x[i][j];
67                      ryy[i][j] = ry_y[i][j];
68                  }
69                  else
70                  {
71                      rxx[i][j] = 0;
72                      ryy[i][j] = 0;
73                  }
74              else
75                  if(j < ch)
76                  {
77                      rxx[i][j] = 0;
78                      ryy[i][j] = 0;
79                  }
80                  else
81                  {
82                      rxx[i][j] = rx_x[i-ch][j-ch];
83                      ryy[i][j] = ry_y[i-ch][j-ch];
84                  }
```

```
85      std::complex<float> vl[ch*2];
86      for (int i=0; i<ch*2; i++)
87      {
88          std::complex<float> soma_linha(0,0);
89          for (int j=0; j<ch*2; j++)
90              soma_linha = soma_linha + (beta * rxx[i][j] * q[j]);
91
92          vl[i] = soma_linha;
93      }
94
95      std::complex<float> ml[ch*2][ch*2];
96      for (int i=0; i<ch*2; i++)
97          for (int j=0; j<ch*2; j++)
98              ml[i][j] = iMM[i][j] - (beta*ryy[i][j]);
99
100     std::complex<float> j_MWF[ch*2];
101     for (int i=0; i<ch*2; i++)
102     {
103         std::complex<float> soma_linha(0,0);
104         for (int j=0; j<ch*2; j++)
105             soma_linha = soma_linha + (ml[i][j] * initial_coeffs[j][bin]);
106
107         soma_linha = soma_linha + vl[i];
108         j_MWF[i] = soma_linha;
109     }
110
111     if (gama == 0)
112     {
113         for (int i=0; i<ch*2; i++)
114             j_TOTAL[i][bin] = j_MWF[i];
115     }
```

Fonte: Arquivo Pessoal

procedimento internamente na saída. Após isso, já no domínio do tempo, o sinal é multiplicado novamente pela janela de análise, como vemos na linha 221.

Após esse processo, o bloco de 64 amostras está filtrado e pronto para ser transmitido nos alto-falantes do aparelho, e como se trata de uma implementação em tempo real, o processo se reinicia no momento que outras 64 amostras são recebidas.

Para ser possível realizar as avaliações dos resultados, o áudio filtrado foi salvo em arquivo texto (.txt) e carregado na plataforma Matlab, onde foi criado um arquivo que extrai os resultados para cada filtragem e gera os áudios de saída.

Figura 3.8: Transformada inversa de Fourier.

```
201     for (int i=0; i<256; i++)
202         for (int j=0; j<2; j++)
203             frame_out_frq_tmp[j][i] = conjPhaseMod[i] * frame_out_frq_in[i][j];
204     //Modificação de fase proposto pelo método weighted overlap-and-add
205
206     for (int j=0; j<2; j++)
207     {
208         bool ovflo;
209         for (int i=0; i<256; i++)
210             ifft_input[i] = std::complex<float>(frame_out_frq_tmp[j][i].real(),frame_out_frq_tmp[j][i].imag());
211
212         fft_top(0, ifft_input, ifft_output, &ovflo);
213         //IFFT
214
215         for (int i=0; i<256; i++)
216             ifft_output[i] = ifft_output[i] / normalization;
217         //Normalização necessária
218
219         for (int i=0; i<256; i++)
220         {
221             std::complex<float> frameMultOverlapAdd = ifft_output[i] * weightingOverlapAdd;
222             //Multiplica o frame de saida pela janela de análise, salvando-o e colocando na saída
223             frame_out[i][j] = frameMultOverlapAdd;
224         }
225     }
```

Fonte: Arquivo Pessoal

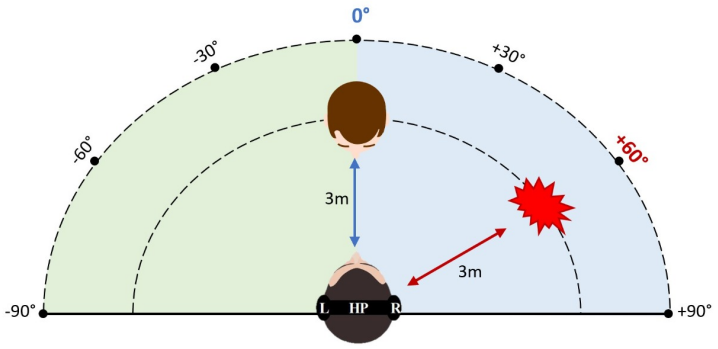
4 AVALIAÇÃO EXPERIMENTAL

Neste capitulo será descrito como foram elaborados os arquivos de entrada do circuito, bem como os resultados obtidos pelos testes realizados.

4.1 Cenários acústicos

Para avaliação do circuito elaborado, foi necessária a criação de cenários acústicos contendo um ponto de fala e um ruído conhecido. Para melhor entendimento foram gerados dois cenários acústicos, sendo o primeiro com uma fonte de fala feminina localizada exatamente à frente do indivíduo, sendo azimute de 0º e uma distância de 3m, e uma fonte de ruído com SNR de entrada de -5dB e posicionado à direita, sendo azimute em +60º e distância de 3m, como podemos observar na figura 4.1, podendo ser escutado um trecho clicando aqui Cenário 1. O segundo cenário possui as mesmas configurações do primeiro, com a alteração de o ruído estar posicionado à esquerda, tendo azimute de -60º, como podemos observar na figura 4.2, podendo ser escutado um trecho clicando aqui Cenário 2. Em ambos os cenários, os áudios captados foram simulados contendo 3 microfones do lado esquerdo, e 3 microfones do lado direito.

Figura 4.1: Cenário acústico 1.

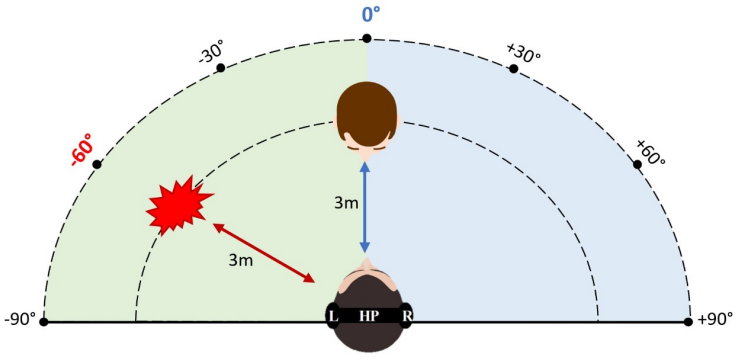


Fonte: Arquivo pessoal

O sinal resultante consiste em um período inicial de 1s contendo apenas ruído, seguido por 6s de fala contaminada. A frequência de amostragem dos sinais é de 16 kHz. O ruído é do tipo ICRA (*International Collegium of Rehabilitative Audiology*) (DRESCHLER et al., 2001), apresenta características espectro-temporais semelhantes às da fala humana e foi gerado simulando uma câmera anecoica, ou seja, sem reverberação.

Os cenários acústicos descritos foram gerados a partir da convolução dos sinais de

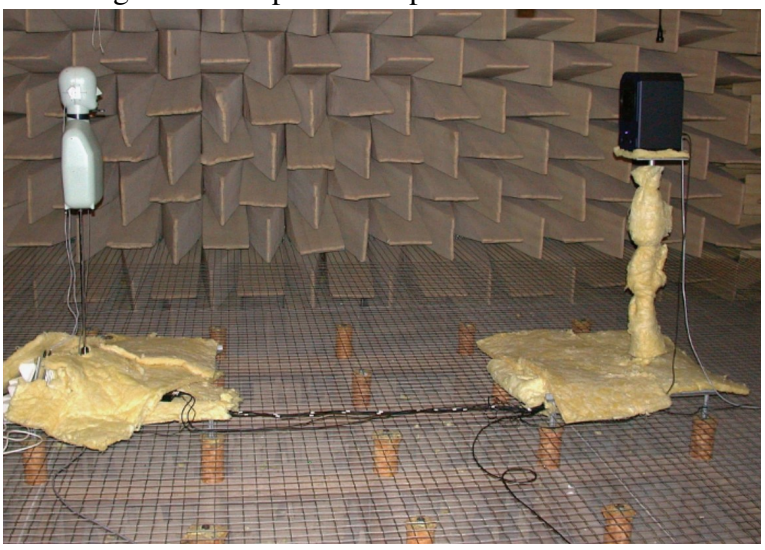
Figura 4.2: Cenário acústico 2.



Fonte: Arquivo pessoal

fala e de ruído com as funções de transferência HRIR (*Head Related Transfer Functions*) que permitem determinar o caminho acústico entre cada fonte e os microfones dos aparelhos auditivos. O banco de HRIRs utilizado foi obtido através de experimentos reais sob as seguintes condições (KAYSER et al., 2009): câmara anecoica, manequim Bruel Kjær tipo 4128-C e um par de aparelhos auditivos retro-auriculares. Cenários acústicos em câmaras anecoicas são usadas como aproximações de cenários conhecidos como *free-field* (campo livre) com baixíssima ou nenhuma reverberação, comuns em situações cotidianas. A configuração do experimento pode ser observada na figura 4.3.

Figura 4.3: Experimento para obter as HRIRs.



Fonte: Experimento de (KAYSER et al., 2009)

4.2 Resultados obtidos

Com os cenários acústicos gerados, foi possível extrair os resultados para as métricas objetivas propostas de $PESQ\text{-}MOS$, $STOI$, ΔSNR, ΔIPD e ΔILD. Para avaliação das métricas, devido ao áudio de entrada possuir 1 segundo inicial de apenas ruído, onde nesta etapa apenas as matrizes de coerência são atualizadas, os resultados e áudios gerados são apresentados a partir do momento que o filtro é acionado. Os áudios de saída, bem como os áudios de entrada, podem ser ouvidos clicando aqui RepositórioAudios. Para melhor entendimento dos resultados apresentados, aconselha-se escutar os sons com fones de ouvido.

4.2.1 Inteligibilidade e Qualidade

Primeiramente, optou-se por extrair os valores que indicam a qualidade e a inteligibilidade do áudio de saída, $PESQ\text{-}MOS$ e $STOI$ respectivamente, sendo o $PESQ\text{-}MOS$ calculado tanto para o canal esquerdo quanto para o direito separadamente.

Neste primeiro experimento, para ambos os cenários, foram aplicados quatro configurações de filtragem, variando o fator de γ com valores de 0 (apenas redução de ruído), 10^{-1}, 10^{-2} e 10^{-3}, escolhidos a partir de outros trabalhos (ITTURRIET; COSTA, 2019) (REYS et al., 2019) que já utilizaram valores dessas ordens, sendo assim possível comparar a utilização do filtro MWF com e sem a preservação da lateralização, incrementando o fator de multiplicação desta técnica. O valor de β foi definido em $1,5 \times 10^{-2}$, também em decorrência da utilização dessa ordem nos trabalhos citados. Para os fatores de esquecimento na estimação das matrizes η_y, η_v e η_x, foi utilizado o valor 0,999, também escolhido em decorrência da utilização desse valor em outros trabalhos consultados.

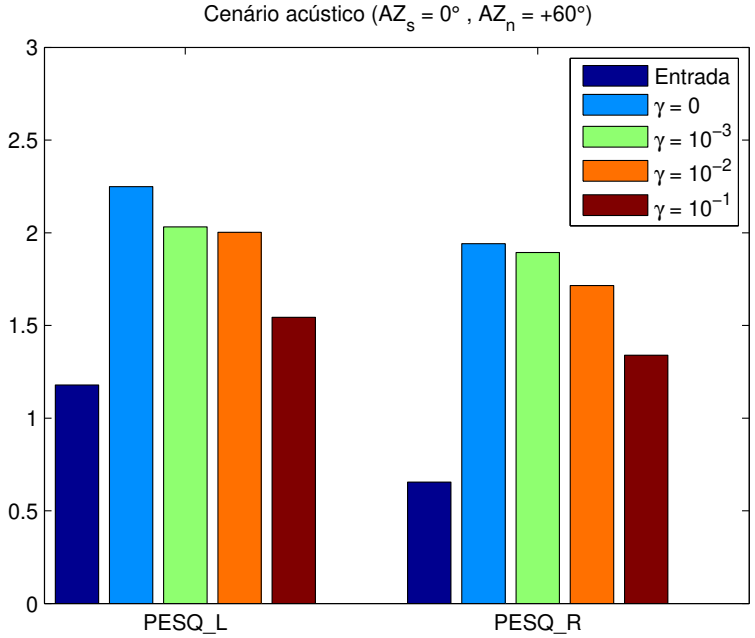
Para o primeiro cenário, no qual o ruído está posicionado à direita, os valores obtidos de referência de entrada, ou seja, sem filtragem, para tais métricas foram de $PESQ\text{-}MOS_L$ = 1.178, $PESQ\text{-}MOS_R$ = 0.655 e $STOI$ = 0.577.

Tabela 4.1: Dados obtidos para PESQ e STOI para o cenário acústico 1.

	$\gamma = 0$	$\gamma = 10^{-3}$	$\gamma = 10^{-2}$	$\gamma = 10^{-1}$
$PESQ\text{-}MOS_L$	2.259	2.057	2.014	1.544
$PESQ\text{-}MOS_R$	1.941	1.894	1.715	1.339
$STOI$	0.750	0.742	0.727	0.677

Como podemos observar na tabela 4.1, a coluna com $\gamma = 0$, onde não há utilização

Figura 4.4: Gráfico em barras para os dados obtidos de PESQ para o cenário acústico 1.



Fonte: Arquivo pessoal

da técnica ITF, é onde alcançamos os melhores resultados de qualidade e inteligibilidade, podendo perceber um ganho em relação ao áudio contaminado de entrada de $\Delta PESQ\text{-}MOS_R$ = 1.286, $\Delta PESQ\text{-}MOS_L$ = 1.070, $\Delta STOI$ = 0.173. À medida que é incrementado o valor de γ, podemos observar uma degradação na qualidade e inteligibilidade, que é esperada visto que a técnica ITF busca a manutenção da posição espacial do ruído, tendo uma penalidade quanto a redução de ruído. A degradação é atenuada com o incremento de γ, tendo a maior penalização para o γ de 10^{-1}, porém ainda assim com valores superiores em comparação ao áudio de entrada contaminado.

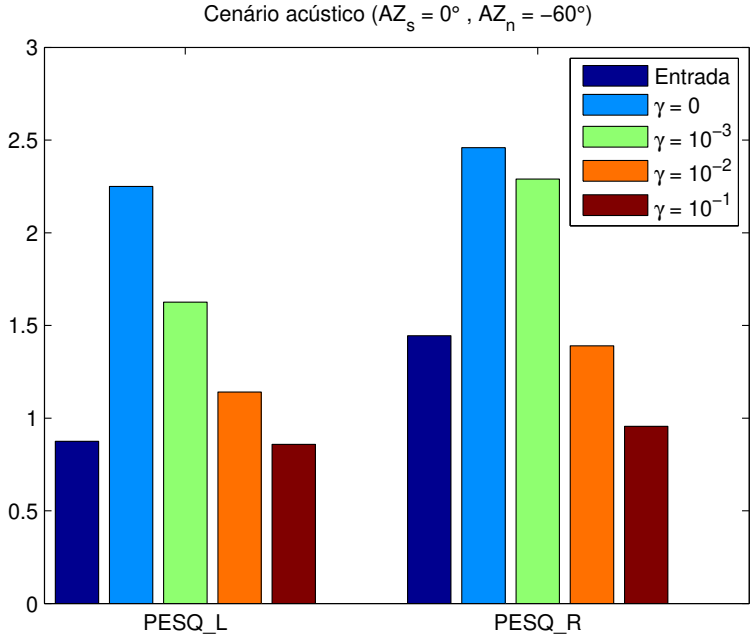
Para o segundo cenário, no qual o ruído está posicionado à esquerda, os valores obtidos de referência de entrada, ou seja, sem filtragem, para tais métricas foram de $PESQ\text{-}MOS_L$ = 0.875, $PESQ\text{-}MOS_R$ = 1.444 e $STOI$ = 0.604.

Tabela 4.2: Dados obtidos para PESQ e STOI para o cenário acústico 2.

	$\gamma = 0$	$\gamma = 10^{-3}$	$\gamma = 10^{-2}$	$\gamma = 10^{-1}$
$PESQ\text{-}MOS_L$	2.250	1.606	1.267	0.859
$PESQ\text{-}MOS_R$	2.459	2.290	1.390	0.956
$STOI$	0.765	0.708	0.628	0.502

Como mostra a tabela 4.2, os valores obtidos para γ = 0, onde não há utilização da técnica ITF, segue com os melhores resultados conforme esperado. Neste cenário acústico, o ganho atingido por esta técnica foi de $\Delta PESQ\text{-}MOS_R$ = 1.015, $\Delta PESQ\text{-}MOS_L$ = 1.375, $\Delta STOI$ = 0.161. Assim como no primeiro cenário acústico, obtemos

Figura 4.5: Gráfico em barras para os dados obtidos de PESQ para o cenário acústico 2.



Fonte: Arquivo pessoal

uma piora na qualidade e inteligibilidade do áudio de saída, à medida que aumentamos do valor de γ. Neste cenário, para $\gamma = 10^{-1}$ os valores de $\Delta PESQ\text{-}MOS_R$, $\Delta PESQ\text{-}MOS_L$ e $\Delta STOI$ foram negativos, o que indica um áudio resultante com qualidade e inteligibilidade inferior ao de entrada. Ainda nesta tabela, para $\gamma = 10^{-2}$, pode-se observar uma ligeira piora na qualidade do áudio do canal direito, por conta de uma leve inserção de ruido neste canal.

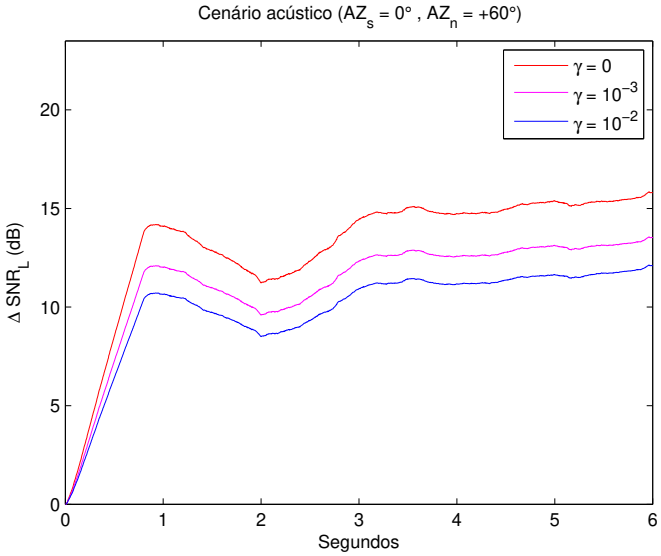
Devido a significativa degradação do áudio de saída ao utilizar $\gamma = 10^{-1}$, optou-se por eliminar a utilização deste fator nas demais métricas, mantendo os valores de γ para 0, 10^{-3} e 10^{-2}.

4.2.2 Redução de ruído

A seguir, serão demonstrados os resultados obtidos para as métricas de ΔSNR_L e ΔSNR_R, sendo os sufixos $_L$ para saída de áudio do aparelho esquerdo (*left*) e $_R$ para saída de áudio do aparelho direito (*right*). Para estes dados, quanto maior for o valor obtido, maior será a redução de ruído do áudio filtrado, em comparação ao áudio de entrada.

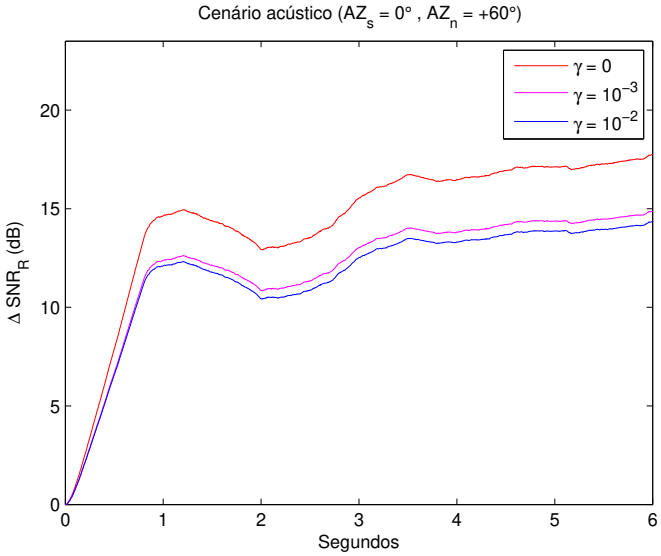
As figuras 4.6 e 4.7 demonstram os resultados obtidos de ΔSNR_L e ΔSNR_R para o cenário acústico 1, onde o ruído está posicionado à direita, tendo no eixo x o tempo em segundos do áudio analisado, podendo ser percebido o processo de adaptação

Figura 4.6: Variação de SNR para o aparelho esquerdo.



Fonte: Arquivo pessoal

Figura 4.7: Variação de SNR para o aparelho direito.

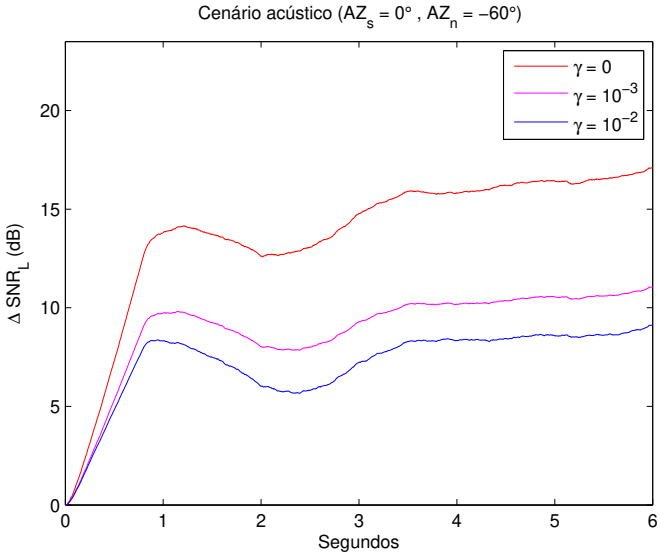


Fonte: Arquivo pessoal

dos coeficientes e impactos na métrica. Podemos observar, em ambos lados, uma melhora na relação sinal-ruído para todos os fatores analisados, sendo o ganho mais significativo com γ = 0, com valores médios de ΔSNR_L = +15.32dB e ΔSNR_R = +16.84dB, como esperado já que o filtro MWF, sem a presença do ITF, tem como objetivo principal a redução de ruído. Pode-se observar uma diferença entre os valores de γ = 0 para os obtidos por γ > 0, sendo mais perceptivo no aparelho do lado direito, onde o ruído está mais próximo. Isso ocorre devido a técnica de preservação das pistas binaurais, ITF, ao tentar manter a posição do ruído, prejudica a redução de ruído, ainda que obtenha valores satisfatórios. A degradação é maior à medida que o valor de γ aumenta, o que é

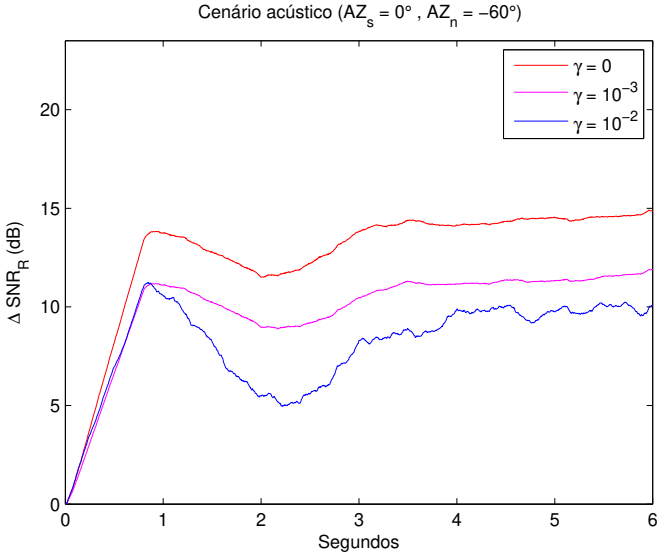
corroborado pelo fato de o maior valor de γ apresentar o menor ganho de SNR.

Figura 4.8: Variação de SNR para o aparelho esquerdo.



Fonte: Arquivo pessoal

Figura 4.9: Variação de SNR para o aparelho direito.



Fonte: Arquivo pessoal

As figuras 4.8 e 4.9 demonstram os resultados obtidos de ΔSNR_L e ΔSNR_R para o cenário acústico 2, onde o ruído está posicionado à esquerda. Podemos observar, como para o primeiro cenário, uma melhora na relação sinal-ruído para todos os fatores analisados, sendo o ganho mais significativo com $\gamma = 0$, com valores médios de ΔSNR_L = +16.43dB e ΔSNR_R = +14.79dB, como esperado já que o filtro MWF, sem a presença do ITF, tem como objetivo principal a redução de ruído. Pode-se observar novamente uma diferença entre os valores de $\gamma = 0$ para os obtidos por $\gamma > 0$, sendo mais perceptivo neste

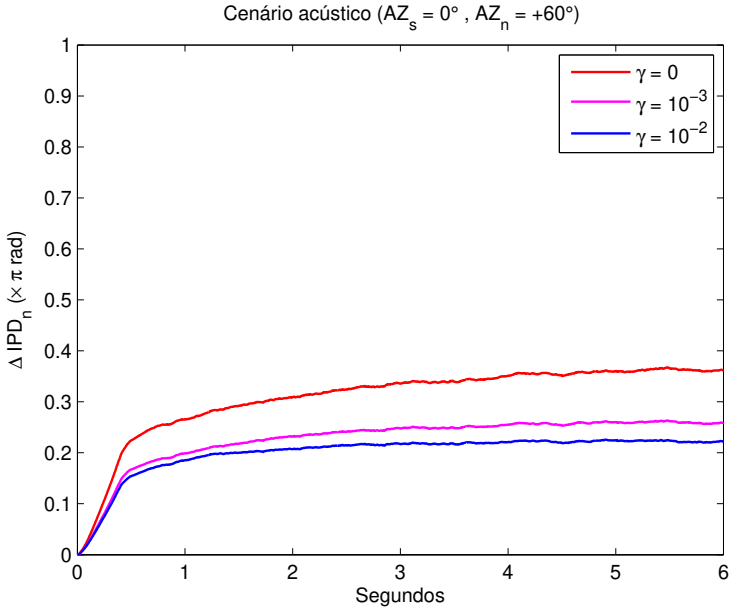
cenário no aparelho do lado esquerdo, onde o ruído está mais próximo. A degradação é maior à medida que o valor de γ aumenta, o que é corroborado pelo fato de o maior valor de γ apresentar o menor ganho de SNR.

Para ambos os cenários, pelo fato de ser um filtro adaptativo, obtém-se melhores resultados à medida que mais amostras de áudio vão sendo recebidas. Por conta disto, uma rampa inicial é observada no primeiro segundo de filtragem, e logo após ocorre uma leve queda, que pode ser explicado devido ao áudio de entrada possuir um momento de pausa na fala entre 1,6 e 2,4 segundos. A partir de aproximadamente 3,2 segundos, o filtro tem uma melhor convergência com as variações tendendo a manterem um valor médio linear.

4.2.3 Lateralização

Para mensurar as distorções da lateralização do sinal analisado, serão apresentados os resultados obtidos para as métricas ΔILD_N, ΔILD_S, ΔIPD_N e ΔIPD_S, sendo os sufixos $_N$ para ruído (*noise*) e $_S$ para sinal de fala (*speech*). Para estes resultados, em ambas métricas, quanto menor for a variação, maior é a preservação das pistas binaurais do sinal processado, o que reflete em uma melhor reprodução do cenário acústico original pelo cérebro do usuário.

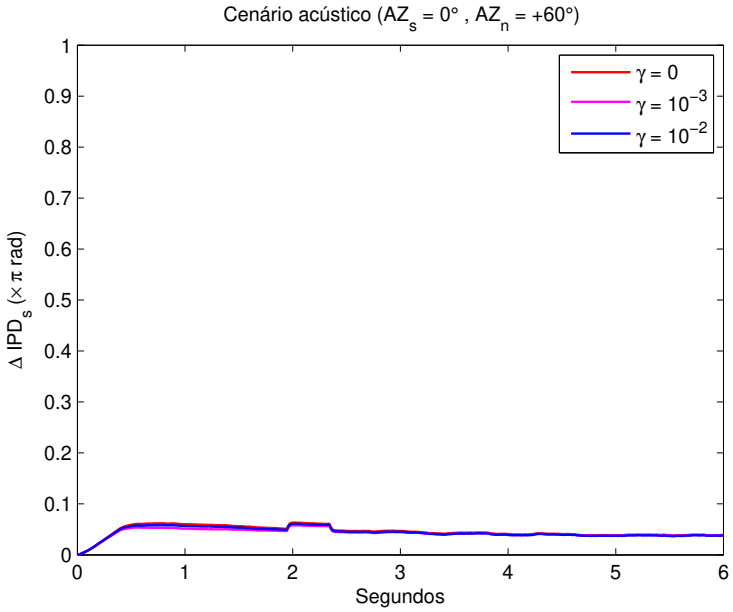
Figura 4.10: Variação de IPD do ruído.



Fonte: Arquivo pessoal

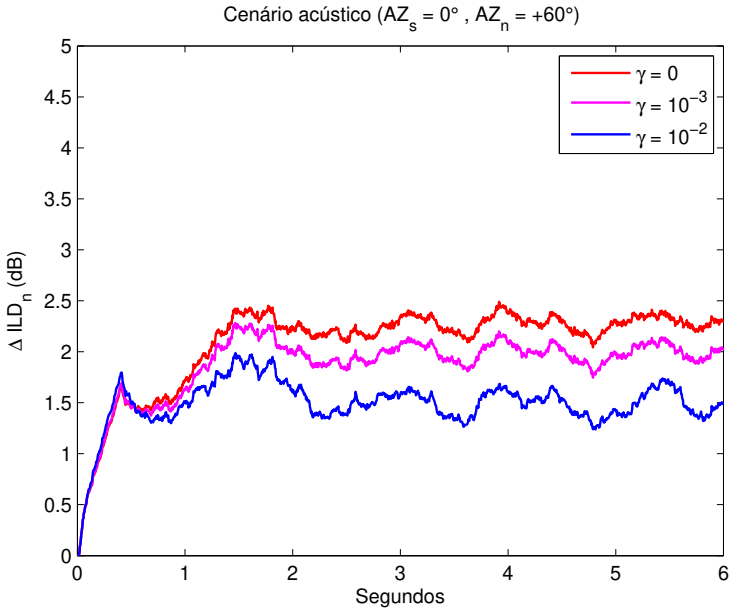
Para o cenário acústico 1, nas figuras 4.10 e 4.12, que mostram as variações da ILD

Figura 4.11: Variação de IPD da fala.



Fonte: Arquivo pessoal

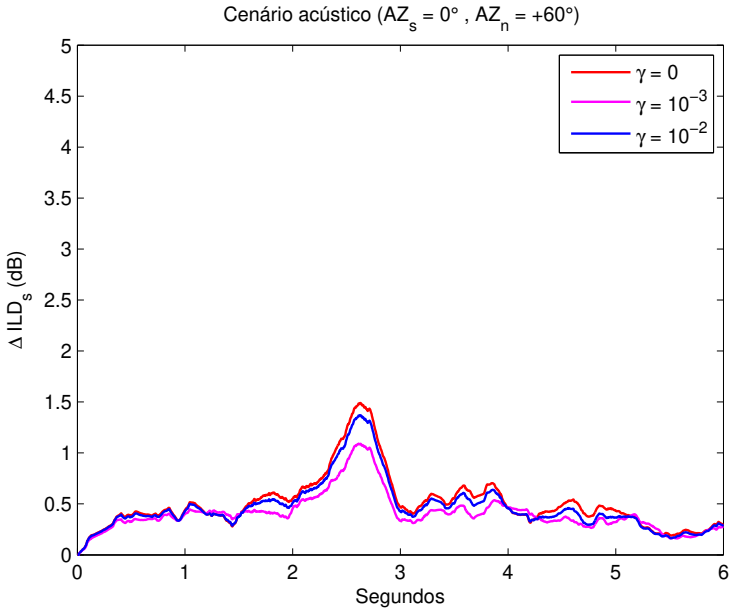
Figura 4.12: Variação de ILD do ruído.



Fonte: Arquivo pessoal

e IPD do ruído em relação ao áudio de entrada, podemos observar que o filtro que não utiliza a técnica de preservação da lateralização do ruído, $\gamma = 0$, obteve a maior variação observada, indicando perdas na lateralização original. Já para os valores observados que utilizam a técnica ITF, tem uma variação menor, sendo o maior γ com a menor variação em ambas métricas, indicando a preservação da posição do ruído. Já para a variação referente ao sinal de fala, mostrados nas figuras 4.11 e 4.13, as variações em todos os valores de γ foram próximas de 0, não modificando o sinal de fala do áudio resultante,

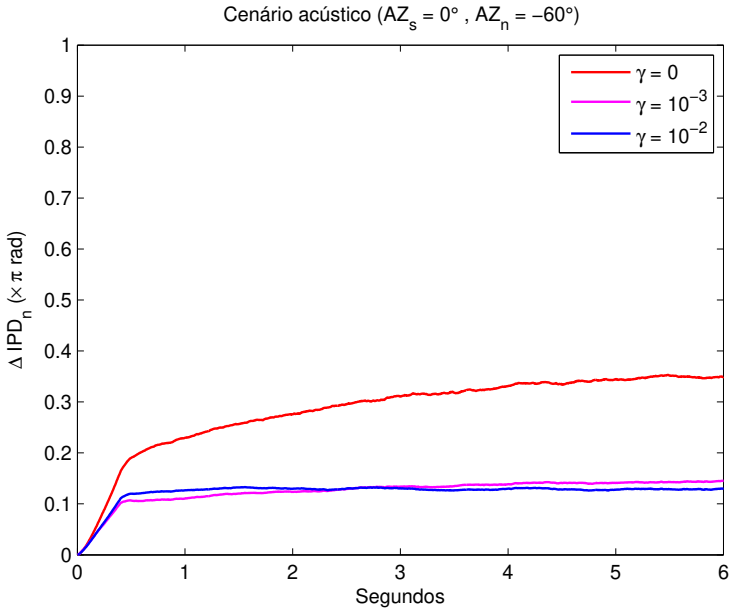
Figura 4.13: Variação de ILD da fala.



Fonte: Arquivo pessoal

conforme esperado.

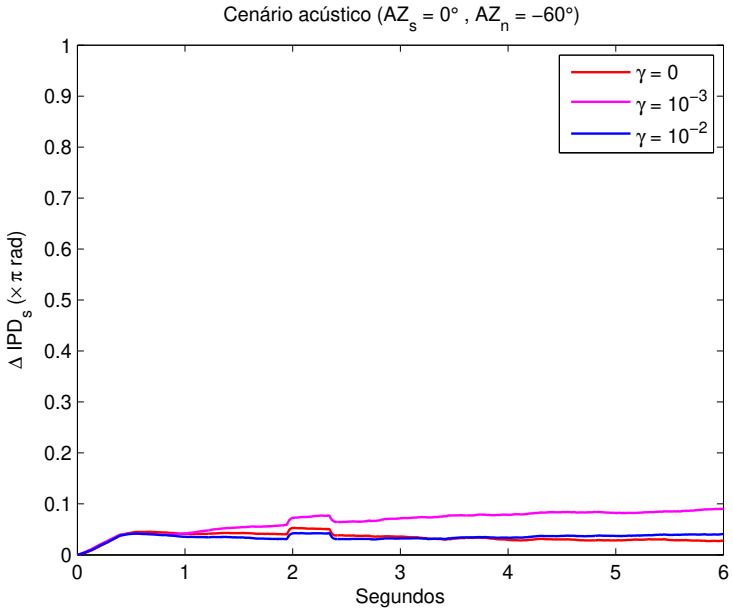
Figura 4.14: Variação de IPD do ruído.



Fonte: Arquivo pessoal

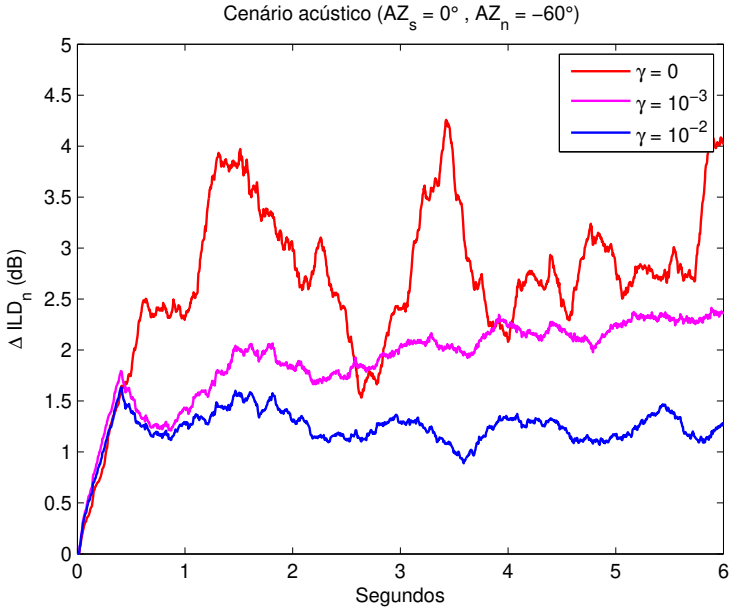
Para o cenário acústico 2, nas figuras 4.14 e 4.16, que mostram as variações de ILD e IPD referente ao ruído em relação ao áudio de entrada, as variações de ILD para γ = 0 não apresentaram tanta linearidade quanto para o cenário acústico 1, ainda que acima dos filtros que utilizaram a técnica de preservação, indicando perdas na lateralização do sinal de ruído original. Novamente, para os valores observados que utilizam a técnica

Figura 4.15: Variação de IPD da fala.



Fonte: Arquivo pessoal

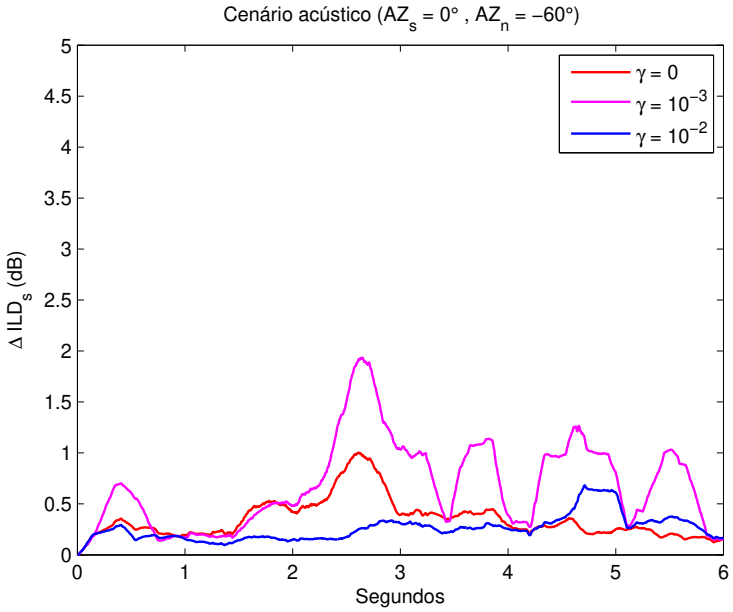
Figura 4.16: Variação de ILD do ruído.



Fonte: Arquivo pessoal

ITF, atingiram uma menor variação, sendo o maior γ com as mais baixas variações em ambas métricas, indicando a preservação da posição do ruído. Já para a variação referente ao sinal de fala, mostrados nas figuras 4.15 e 4.17, as variações em todos os valores de γ foram próximas de 0, não modificando o sinal de fala do áudio resultante, conforme esperado. Ainda que observado uma maior variação da ILD para $\gamma = 10^{-3}$, a posição da fala no áudio resultante não foi alterado.

Figura 4.17: Variação de ILD da fala.



Fonte: Arquivo pessoal

4.2.4 Resultados de *hardware*

Agora, serão apresentados os resultados obtidos de *hardware* pela ferramenta de síntese da Xilinx, como latência, frequência de operação e uso de recursos.

Foi utilizada a frequência de operação sugerida pela ferramenta de síntese de alto nível de 100MHz, e utilizado como ponto de partida o FPGA UltraScale Plus da Xilinx, modelo VU35P, que dispõe de 1,907k FFs (*Flip Flops*), 872k LUTs (*Lookup Tables*), 5,952 DSPs e 2268 BRAMs (*Block RAM*). O FPGA foi escolhido de modo que facilitasse a instanciação do core, onde o espaço utilizado não fosse um limitante. Após a análise de ocupação do core, a exploração de FPGAs de menor porte é um trabalho futuro importante.

Em relação à latência, por conta de o circuito efetuar a filtragem a cada 64 amostras de áudio a uma frequência de 16kHz, possuímos então uma restrição de 4ms por operação. A latência máxima obtida pelo circuito, do momento que recebe que as 64 amostras de áudio até chegar ao resultado final para o *frame*, foi de 1,535ms, o que proporciona um tempo ocioso de 2,465ms entre um *frame* e outro. O atraso entre o recebimento da primeira amostra até a reprodução filtrada nos alto-falantes é de 5,535ms. De acordo com alguns estudos (BURWINKEL; MCKINNEY; GALSTER, 2017), estima-se que em média o atraso aceitável que não causa desconfortos aos usuários de aparelhos auditivos é de aproximadamente 10ms. Desse modo, a latência obtida, levando em consideração

as baixas latências previstas para a tecnologia 5G, conforme demonstrado em (SLALMI et al., 2020), se adequa aos requisitos de tempo que um aparelho auditivo necessita, possibilitando a utilização do circuito como por exemplo um dispositivo de borda.

Os recursos utilizados ficaram em 64,141 FFs, 67,890 LUTs, 297 DSPs e 98 BRAMs. A potência consumida em média teve o valor de 3,433W. A tabela 4.3 contém os dados de utilização de recursos, juntamente com a porcentagem de uso do FPGA utilizado para simulação.

Tabela 4.3: Recursos utilizados pelo FPGA.

	Utilizado	Disponível	(%)
FF	64,141	1907k	3.36
LUT	67,890	827k	8.21
DSP	297	5952	4.99
BRAM	98	2268	4.32

5 CONCLUSÃO

Este trabalho apresentou a implementação, em FPGA, de um método adaptativo de redução de ruído em tempo real para aparelhos auditivos binaurais, com preservação de cenário acústico. A partir dos resultados obtidos das métricas objetivas, pode-se comprovar que o algoritmo implementado de redução de ruído conseguiu satisfazer sua teoria. Também pode-se perceber que de fato o filtro de Wiener, sem adição de outras técnicas, conseguiu preservar as pistas acústicas da fonte de fala. No entanto, acaba por alterar as informações binaurais, utilizadas pelo cérebro, para localização da fonte de ruído.

Essa alteração conseguiu ser revertida com a utilização da técnica de preservação de cenário acústico ITF. Ajustando o parâmetro γ, que permite regular a utilização da técnica na filtragem, pode-se perceber uma diminuição nas variações das pistas acústicas tanto do sinal de fala quanto do ruído, porém sendo penalizado com a diminuição de ganho no aspecto de redução de ruído. Porém, a utilização de γ da ordem de 10^{-1} ou superior, acabou por comprometer a qualidade e inteligibilidade da fala, gerando áudios resultantes piores em relação aos áudios de entrada.

Com relação a utilização do algoritmo em tempo real, os resultados obtidos pela ferramenta de síntese mostram que a latência atingida para o processamento do sinal é satisfatória, permitindo a utilização da implementação em dispositivos de borda, por conta de possuir tempo suficiente de transmissão entre os dispositivos sem apresentar desconfortos aos usuários. Há de se ressaltar que o presente trabalho não demonstrou a implementação em *hardware* do algoritmo de detecção de atividade de voz, que é utilizado no sistema, onde possivelmente pode apresentar impactos à latência obtida, que podem ser minimizados utilizando técnicas de implementação de *hardware*, como pipeline e paralelismo.

Devido ao fato de a utilização da implementação ocorrer como dispositivo de borda, as restrições de projeto impostas para utilização diretamente nos aparelhos auditivos, como consumo energético e tamanho do dispositivo, não foram exigidas neste trabalho.

6 TRABALHOS FUTUROS

Os resultados e conclusões apresentados nesse trabalho abrem uma série de possibilidades de trabalhos envolvendo a circuito apresentado. Dentre elas podemos elencar:

- Implementar e utilizar *on-board* algoritmos de detecção de atividade de voz (VAD)
- Alterar o tipo de dado utilizado, de ponto flutuante para ponto fixo, para comparação do áudio processado e do impacto nas métricas de avaliação objetivas e nas métricas de avaliação do *hardware*.
- Explorar diretivas de otimização do HLS e outras formas de implementação para aprimorar o circuito.
- Explorar os resultados obtidos para variação de número de microfones de entrada.
- Substituir o filtro de Wiener multicanal por outros algoritmos de redução de ruído existentes.
- Prototipação da implementação proposta, para refinar os resultados obtidos.
- Desenvolver um classificador de cenários acústicos para o dispositivo de borda, visando melhorar a performance dos algoritmos de redução de ruído.

REFERÊNCIAS

BEERENDS, John G; STEMERDINK, Jan A. A perceptual speech-quality measure based on a psychoacoustic sound representation. **Journal of the Audio Engineering Society**, Audio Engineering Society, v. 42, n. 3, p. 115–123, 1994.

BLAUERT, Jens. The Psychophysics of Human Sound Localization. **MIT press**, jan. 1997.

BRILLINGER, D.R. **Time Series: Data Analysis and Theory**. [S.l.]: Society for Industrial e Applied Mathematics, 2001. (Classics in Applied Mathematics). ISBN 9780898715019. Disponível em: <<https://books.google.com.br/books?id=PX5HExMKER0C>>.

BURWINKEL, Justin; MCKINNEY, Martin; GALSTER, Jason. **Acceptable Hearing Aid Throughput Delay for Listeners with Hearing Loss Under Noisy Conditions**. [S.l.: s.n.], mar. 2017. DOI: <10.13140/RG.2.2.36471.73122>.

CARMO, Diego; COSTA, Márcio. Online approximation of the multichannel Wiener filter with preservation of interaural level difference for binaural hearing-aids. **Computers in Biology and Medicine**, v. 95, fev. 2018. DOI: <10.1016/j.compbiomed.2018.02.017>.

CORNELIS, Bram; MOONEN, Marc; WOUTERS, Jan. Performance analysis of multichannel Wiener filter-based noise reduction in hearing aids under second order statistics estimation errors. **IEEE Transactions on Audio, Speech, and Language Processing**, IEEE, v. 19, n. 5, p. 1368–1381, 2010.

COUSSY, Philippe et al. An Introduction to High-Level Synthesis. **IEEE**, IEEE, v. 26, n. 4, p. 8–17, 2009.

CROCHIERE, R. A weighted overlap-add method of short-time Fourier analysis/Synthesis. **IEEE Transactions on Acoustics, Speech, and Signal Processing**, v. 28, n. 1, p. 99–102, 1980. DOI: <10.1109/TASSP.1980.1163353>.

DOCLO, Simon; MOONEN, Marc. On the output SNR of the speech-distortion weighted multichannel Wiener filter. **IEEE Signal Processing Letters**, IEEE, v. 12, n. 12, p. 809–811, 2005.

DOCLO, Simon; SPRIET, Ann et al. Speech Distortion Weighted Multichannel Wiener Filtering Techniques for Noise Reduction. In: SPEECH Enhancement. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005. p. 199–228. ISBN 978-3-540-27489-6. DOI: <10.1007/3-540-27489-8_9>. Disponível em: <<https://doi.org/10.1007/3-540-27489-8_9>>.

DRESCHLER, Wouter et al. ICRA Noises: Artificial Noise Signals With Speech-Like Spectral and Temporal Properties for Hearing Aid Assessment. **Audiology**, v. 40, p. 148–157, jan. 2001.

HARTMANN, William; MACAULAY, Eric. Anatomical limits on interaural time differences: An ecological perspective. **Frontiers in neuroscience**, v. 8, p. 34, fev. 2014. DOI: <10.3389/fnins.2014.00034>.

HARTMANN, William; RAKERD, Brad et al. Transaural experiments and a revised duplex theory for the localization of low-frequency tones. **The Journal of the Acoustical Society of America**, v. 139 2, p. 968–85, 2016.

HEARING Aids improve Hearing - and a LOT more. [S.l.], 2020 (Acessado Out 20, 2021). Disponível em: <<https://www.ehima.com/wp-content/uploads/2020/07/EuroTrak-Trends-2009-2020-June-2020.pdf>>.

ITTURRIET, Fábio Pires. **Preservação perceptualmente relevante da diferença de tempo interaural em aparelhos auditivos binaurais**. 2019. f. 159. Tese (Doutorado) – Universidade Federal de Santa Catarina, Santa Catarina.

ITTURRIET, Fábio Pires; COSTA, Márcio Holsbach. Perceptually relevant preservation of interaural time differences in binaural hearing aids. **IEEE/ACM Transactions on Audio, Speech, and Language Processing**, IEEE, v. 27, n. 4, p. 753–764, 2019.

ITU. Mean opinion score (MOS) terminology. **Recommendation ITU-T P. 800.1. ITU-T Telecommunication Standardization Sector of ITU Geneva**, 2016.

KAYSER, H. et al. Database of Multichannel In-Ear and Behind-the-Ear Head-Related and Binaural Room Impulse Responses. **EURASIP Journal on Advances in Signal Processing**, v. 2009, p. 6, dez. 2009. DOI: <10.1155/2009/298605>.

KENT, Ray D. et al. Toward Phonetic Intelligibility Testing in Dysarthria. **Journal of Speech and Hearing Disorders**, v. 54, n. 4, p. 482–499, 1989. DOI: <10.1044/jshd.5404.482>. eprint:

<https://pubs.asha.org/doi/pdf/10.1044/jshd.5404.482>. Disponível em: <<https://pubs.asha.org/doi/abs/10.1044/jshd.5404.482>>.

KIESER, Robert; REYNISSON, Pall; MULLIGAN, Timothy J. Definition of signal-to-noise ratio and its critical role in split-beam measurements. **ICES Journal of Marine Science**, v. 62, n. 1, p. 123–130, jan. 2005. ISSN 1054-3139. DOI: <10.1016/j.icesjms.2004.09.006>. eprint: <https://academic.oup.com/icesjms/article-pdf/62/1/123/29150237/62-1-123.pdf>. Disponível em: <<https://doi.org/10.1016/j.icesjms.2004.09.006>>.

KLASEN, T.J. et al. Binaural Multi-Channel Wiener Filtering for Hearing Aids: Preserving Interaural Time and Level Differences. In: v. 5, p. v–v. DOI: <10.1109/ICASSP.2006.1661233>.

KLASEN, Thomas J et al. Binaural multi-channel Wiener filtering for hearing aids: preserving interaural time and level differences. In: IEEE. 2006 IEEE International Conference on Acoustics Speech and Signal Processing Proceedings. [S.l.: s.n.], 2006. v. 5, p. v–v.

LOIZOU, Philipos C. **Speech Enhancement: Theory and Practice**. 2. ed. [S.l.]: CRC Press, 2013.

PLENGE, Georg. On the differences between localization and lateralization. **The Journal of the Acoustical Society of America**, v. 56 3, p. 944–51, 1974.

R.S., Lord Rayleigh O.M. Pres. XII. On our perception of sound direction. **The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science**, Taylor Francis, v. 13, n. 74, p. 214–232, 1907. DOI: <10.1080/14786440709463595>. eprint: <https://doi.org/10.1080/14786440709463595>. Disponível em: <<https://doi.org/10.1080/14786440709463595>>.

RAMÍREZ, Javier; GORRIZ, Juan; SEGURA, José. Voice Activity Detection. Fundamentals and Speech Recognition System Robustness. In: [s.l.: s.n.], jun. 2007. 6(9). ISBN 978-3-902613-08-0. DOI: <10.5772/4740>.

REYS, Arthur et al. Implementação em tempo real de um sistema de redução de ruído binaural com preservação da função de transferência interaural. In: DOI: <10.14209/sbrt.2019.1570559059>.

RIX, A.W. et al. Perceptual evaluation of speech quality (PESQ)-a new method for speech quality assessment of telephone networks and codecs. In: 2001 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings (Cat. No.01CH37221). [S.l.: s.n.], 2001. v. 2, 749–752 vol.2. DOI: <10.1109/ICASSP.2001.941023>.

SHI, Weisong et al. Edge Computing: Vision and Challenges. **IEEE Internet of Things Journal**, v. 3, n. 5, p. 637–646, 2016. DOI: <10.1109/JIOT.2016.2579198>.

SLALMI, Ahmed et al. On the Ultra-Reliable and Low-Latency Communications for Tactile Internet in 5G Era. **Procedia Computer Science**, v. 176, p. 3853–3862, jan. 2020. DOI: <10.1016/j.procs.2020.09.003>.

TAAL, Cees H. et al. A short-time objective intelligibility measure for time-frequency weighted noisy speech. In: 2010 IEEE International Conference on Acoustics, Speech and Signal Processing. [S.l.: s.n.], 2010. p. 4214–4217. DOI: <10.1109/ICASSP.2010.5495701>.

VAN DEN BOGAERT, Tim et al. Speech enhancement with multichannel Wiener filter techniques in multimicrophone binaural hearing aids. **The Journal of the Acoustical Society of America**, Acoustical Society of America, v. 125, n. 1, p. 360–371, 2009.

VORAN, Stephen. Objective estimation of perceived speech quality. I. Development of the measuring normalizing block technique. **IEEE Transactions on speech and audio processing**, IEEE, v. 7, n. 4, p. 371–382, 1999.

WERNER, Johnny; COSTA, Marcio Holsbach. A Noise-Reduction Method With Coherence Enhancement for Binaural Hearing Aids. **Journal of Communication and Information Systems**, v. 35, n. 1, p. 338–348, 2020.

WIENER, Norbert. **Extrapolation, Interpolation, and Smoothing of Stationary Time Series**. [S.l.]: The MIT Press, 1964. ISBN 0262730057.

WORLD HEALTH ORGANIZATION. 1 in 4 people projected to have hearing problems by 2050. OMS, 2021. Disponível em: <<https://www.who.int/news/item/02-03-2021-who-1-in-4-people-projected-to-have-hearing-problems-by-2050>>.

XILINX. **Fast Fourier Transform v9.1**. [S.l.: s.n.]. <https://www.xilinx.com/support/documentation/ip_documentation/xfft/v9_1/pg109-xfft.pdf>.