

UNIVERSIDADE FEDERAL DO AMAZONAS
INSTITUTO DE COMPUTAÇÃO
PROGRAMA DE PÓS-GRADUAÇÃO EM INFORMÁTICA

**COMPOSIÇÃO MUSICAL COLABORATIVA BASEADA EM
ESPACIALIZAÇÃO SONORA EM TEMPO REAL**

MAURO JÚNIOR BATISTA AMAZONAS

MANAUS, AM
2017

MAURO JÚNIOR BATISTA AMAZONAS

**COMPOSIÇÃO MUSICAL COLABORATIVA BASEADA EM
ESPACIALIZAÇÃO SONORA EM TEMPO REAL**

Texto de dissertação apresentado ao Instituto de Computação da Universidade Federal do Amazonas como parte dos requisitos para obtenção do título de Mestre em Informática pelo Programa de Pós-Graduação em Informática-PPGI.

UNIVERSIDADE FEDERAL DO AMAZONAS

Orientador: Prof. Dr. Bruno Freitas Gadelha

MANAUS, AM
2017

Ficha Catalográfica

Ficha catalográfica elaborada automaticamente de acordo com os dados fornecidos pelo(a) autor(a).

A489c Amazonas, Mauro Júnior Batista
Composição musical colaborativa baseada em espacialização sonora em tempo real / Mauro Júnior Batista Amazonas. 2017
100 f.: il. color; 31 cm.

Orientador: Prof. Dr. Bruno Freitas Gadelha
Dissertação (Mestrado em Informática) - Universidade Federal do Amazonas.

1. Espacialização Sonora. 2. Sistemas Colaborativos. 3. Música Eletroacústica. 4. Projeto de Interação Sônica. I. Gadelha, Bruno Freitas II. Universidade Federal do Amazonas III. Mestre em Informática.



PODER EXECUTIVO
MINISTÉRIO DA EDUCAÇÃO
INSTITUTO DE COMPUTAÇÃO

PROGRAMA DE PÓS-GRADUAÇÃO EM INFORMÁTICA



UFAM

FOLHA DE APROVAÇÃO

"Composição Musical Colaborativa Baseada em Especialização
Sonora em Tempo Real"

MAURO JÚNIOR BATISTA AMAZONAS

Dissertação de Mestrado defendida e aprovada pela banca examinadora constituída pelos Professores:

Prof. Bruno Freitas Gadelha - PRESIDENTE

Profa. Thaís Helena Chaves de Castro - MEMBRO INTERNO

Profa. Rosiane de Freitas Rodrigues - MEMBRO INTERNO

Prof. Hugo Fuks - MEMBRO EXTERNO

Manaus, 01 de Dezembro de 2017

Agradecimentos

Quero agradecer a Deus por sua infinita misericórdia para comigo, à minha esposa Angélica e ao meu amado filho João, os meus bens mais preciosos, por alegrarem meu coração em meio aos momentos difíceis. Agradeço aos meus sogros, Afonso e Meire, pelo incentivo e apoio que me deram ao longo dos meus estudos, bem como aos amigos que fiz durante esta jornada e que me deram força nas adversidades, também responsáveis por eu ter chegado aqui. Por fim, quero expressar minha profunda gratidão ao meu orientador, Prof. Dr. Bruno Gadelha, por sua paciência e fé depositadas em mim no decurso desta etapa.

Este trabalho é dedicado à minha amada mãe que, infelizmente, já não está mais aqui; contudo, seu sonho era me ver alcançar lugares jamais imaginados. Dedico também à minha pequena família, a qual amo muito. É por eles que fui até o fim.

*“Meu amado filho, estude, porque podem lhe tirar tudo nesta vida,
mas o seu conhecimento é a única coisa que ninguém pode roubar.”*
(Antônia Íris, minha mãe)

Resumo

A localização sonora é uma habilidade que o ser humano possui, a qual lhe permite determinar a direção de uma certa fonte sonora. Para explorar essa habilidade, atualmente, o som espacializado é amplamente utilizado na arte, na indústria cinematográfica, em jogos e na realidade virtual. Em muitos desses cenários, um engenheiro ou um projetista de som é o ator central e responsável por dar emoção às imagens projetadas. É o projetista de som que orquestra a distribuição de cada elemento de uma cena sonora pelas caixas de som espalhadas em um ambiente, sendo sua participação obrigatória na espacialização, até o momento. Existem várias aplicações que proporcionam uma espacialização sonora em tempo real, mas, na maioria dos casos, o processo ainda depende desse personagem. Na arte, quando se trata de um concerto de música eletroacústica, esse personagem também está presente, como responsável pelo controle do som. Nos jogos e na realidade virtual, a sonorização dos elementos e seus comportamentos seguem um roteiro definido. Com o propósito de suprir a demanda por ferramentas que proporcionem uma espacialização sonora, considerando diferentes fontes sonoras a serem controladas colaborativamente por atores distintos, este trabalho apresenta um *framework* escalável que proporciona espacialização de sons em tempo real, baseado em Projeto de Interação Sônica, aplicável a múltiplos contextos. De maneira oposta a outros trabalhos existentes voltados para a espacialização sonora em tempo real, esse *framework*, desenvolvido no ambiente Pure Data, traz uma flexibilidade da configuração de alto-falantes a serem utilizados, permitindo também o controle total e individual da localização de várias fontes sonoras sendo processadas em tempo real. Por fazer uso de geração de código personalizado, esse *framework* traz independência, escalabilidade e performance para as aplicações, possibilitando, assim, a eliminação da figura centralizadora do projetista de som. Sendo aplicável em performances de música eletroacústica, ou mesmo em performances colaborativas em que cada usuário se torna plateia e compositor, ao mesmo tempo. Há, também, a facilitação do desenvolvimento de aplicações independentes que explorem o processamento de som em tempo real, como em jogos, simuladores e realidade virtual. Como base para a resolução do problema de espacialização sonora foi utilizado o *Design Thinking* (DT), uma abordagem focada em soluções para problemas complexos de maneiras inovadoras e criativas. Os protótipos gerados a partir do processo de DT deram origem ao framework; uma vez desenvolvido, foi realizada uma análise em pelo menos três contextos diferentes: Análise do ponto de saturação do som, composição colaborativa de música eletroacústica espacializada e espacialização sonora de jogos.

Palavras-chaves: Espacialização Sonora, Sistemas Colaborativos, Música Eletroacústica, Projeto de Interação Sônica.

Abstract

Sound localization is a skill that the human being possesses, which allows him to determine the direction of a certain sound source. To exploit this ability today, spatialized sound is widely used in art, film industry, gaming and virtual reality. In many of these scenarios, an engineer or sound designer is the central actor responsible for giving emotion to the projected images. He is the sound designer who orchestrates the distribution of each element of a sound scene by the sound boxes scattered in an environment, being his obligatory participation in the spatialization, so far. There are several applications that provide a spatial sound in real time, but in most cases, the process still depends on this character. In art, when it comes to an electroacoustic music concert, this character is also present, as responsible for controlling the sound. In games and in virtual reality, the sonorization of the elements and their behaviors follow a defined script. In order to supply the demand for tools that provide a sound spatialization, considering different sound sources to be controlled collaboratively by different actors, this work presents a scalable framework that provides real time spatialization, based on Project of Sonic interaction, applicable to multiple contexts. As opposed to other existing works aimed at real-time sound spatialization, this framework, developed in the Pure Data environment, provides a flexibility in the configuration of the speakers to be used, also allowing total and individual control of the location of various sound sources being processed in real time. By making use of custom code generation, this framework provides independence, scalability and performance for the applications, thus eliminating the centralizing figure of the sound designer. Being applicable in performances of electroacoustic music, or even in collaborative performances in which each user becomes an audience and composer at the same time. There is also the facilitation of the development of independent applications that exploit sound processing in real time, such as games, simulators and virtual reality. As a basis for solving the problem of spatial spatialization, Design Thinking (DT) was used, a solution-focused approach to complex problems in innovative and creative ways. The prototypes generated from the DT process gave rise to the developed framework; an analysis was made in at least three different contexts: Sound saturation point analysis, collaborative composition of spatialized electroacoustic music and sound spatialization of games.

Key-words: Spatial Sound, Collaborative Systems, Electroacoustic Music, Sonic Interaction Design.

Sumário

1	Introdução	14
1.1	Objetivos	16
1.2	Metodologia	17
1.3	Sumário Estruturado da Dissertação	18
2	Fundamentação Teórica	19
2.1	Espacialização Sonora	19
2.1.1	Espacialização Sonora na Música	20
2.1.2	Técnicas de Espacialização	22
2.1.2.1	Binaural	22
2.1.2.2	Estéreo	23
2.1.2.3	Dolby Surround	24
2.1.2.4	<i>Ambisonics</i>	25
2.2	Tecnologias e Linguagens de Programação para o Áudio	27
2.2.1	MAX e Pure Data	28
2.2.2	AUDIENCE	29
2.2.3	SuperCollider	31
2.2.4	ChucK	32
2.3	Projeto de Interação Sônica	33
2.4	Trabalhos Relacionados	35
2.5	Conclusão	39
3	Processo de Design Thinking na Composição Musical Espacializada	42
3.1	Compreendendo o Problema	42
3.2	Projetando Soluções	43
3.3	Prototipação	45
3.3.1	Primeiro Protótipo	45
3.3.1.1	Discussão	46
3.3.2	Segundo Protótipo	49
3.3.2.1	Estudo Empírico	51
3.3.2.2	Planejamento	51
3.3.2.3	Execução	52
3.3.2.4	Discussão	53
3.3.3	Terceiro Protótipo	54
3.3.3.1	Discussão	55
3.3.4	Ajustes de Protótipo	56
3.3.4.1	Discussão	58
3.4	Implementação da Melhor Solução	59

3.5	Conclusão	59
4	ARSUS Framework	61
4.1	Requisitos para Espacialização Sonora	61
4.2	Arquitetura	63
4.2.1	Módulo de Comunicação	63
4.2.2	Módulo de Criação de <i>Subpatches</i>	65
4.2.3	Objeto de Interação e Módulo de Espacialização	65
4.3	Requisitos de Uso do ARSUS	67
4.4	Conclusão	68
5	Avaliação do Framework	69
5.1	ARSUS <i>Framework</i> na Avaliação do Ponto de Saturação do Som	69
5.1.1	Conceitos Sonoros	70
5.1.2	Execução	72
5.1.3	Discussão	72
5.2	ARSUS <i>Framework</i> na Composição Colaborativa de Música Eletroacústica Espacializada	75
5.2.1	Planejamento	75
5.2.2	Execução	76
5.2.3	Discussão	78
5.3	ARSUS <i>Framework</i> no Contexto de Jogos	80
5.3.1	Execução	80
5.3.2	Discussão	81
5.4	Conclusão	83
6	Conclusão	85
	Referências	89
	Apêndices	93
	APÊNDICE A Questionário de experiência de uso do Compomus	94
	APÊNDICE B Questionário de experiência de uso do Compomus e ARSUS Framework	97

Lista de Figuras

Figura 1 – Exemplo de fontes sonoras em disposições diferentes que chegam com intensidades e tempos diferentes a cada ouvido	20
Figura 2 – Ambientação não espacializada: mesmo com o posicionamento no caso da orquestra, a fonte sonora será percebida como única pelo ouvinte, assim como no quarteto de cordas.	21
Figura 3 – Sistema de espacialização sonora desenvolvido para a reprodução de <i>Répons</i> , de Pierre Boulez.	22
Figura 4 – Funções de transferências para os ouvidos direito e esquerdo.	23
Figura 5 – À esquerda, Sistema Binaural partindo de uma única fonte sonora, e, à direita, dois alto-falantes simulam uma fonte sonora “fantasma”.	24
Figura 6 – Padrão surround 5.1, estabelecido pela norma (ITU-RBS.755-1).	25
Figura 7 – Coordenadas esféricas utilizadas pelas equações de codificação <i>Ambisonics</i>	27
Figura 8 – Implementação de um reproduzidor de áudio simples em MAX.	28
Figura 9 – Implementação de um reproduzidor de som em <i>loop</i> no Pure Data.	29
Figura 10 – Arquitetura de camadas do AUDIENCE (FARIA; ALVES, 2005).	30
Figura 11 – Trecho de código na linguagem SuperCollider.	31
Figura 12 – Trecho de código na linguagem ChuckK.	33
Figura 13 – iPod <i>Classic</i> com roda mecânica e iPod com sensor tátil <i>click wheel e feedback</i> sonoro.	35
Figura 14 – <i>atmoSphere</i> sendo utilizado no teste de experiência do usuário (FUSHIMI et al., 2017).	36
Figura 15 – Artista em uma piscina carregando as imagens a serem lidas pelo sistema que, por sua vez, executa o som; Cubos sonoros sendo testados em frente ao computador (NAS; LOPES, 2016).	36
Figura 16 – Reproduzindo três trajetórias e controlando três fontes sonoras (GARCIA; FAVORY; BRESSON, 2016).	37
Figura 17 – À esquerda, <i>Big Buoy</i> em uma interação com crianças e, à direita, <i>Music Troll</i> em uma Performance para o público (JENSENIUS; VOLDSUND, 2012).	38
Figura 18 – Espaço de interação virtual no TUIO Simulator.	46
Figura 19 – Versão de testes do primeiro protótipo em que cada usuário foi modelado manualmente um a um.	48
Figura 20 – Segundo protótipo da aplicação servidora com melhorias de performance e código encapsulado.	49
Figura 21 – Primeira versão funcional do Compomus.	50

Figura 22 – Diagrama de funcionamento do espaço de interação.	51
Figura 23 – Terceiro protótipo da aplicação do lado servidor.	54
Figura 24 – Teste de espacialização quadrafônica com participantes.	56
Figura 25 – Tela C do Compomus para o quarto protótipo.	57
Figura 26 – Avaliação do quarto protótipo.	58
Figura 27 – Arquitetura proposta para o <i>Framework</i>	64
Figura 28 – Disposições de alto-falantes suportadas pelo ARSUS.	66
Figura 29 – Aplicação do ARSUS para testar o ponto de saturação do som.	70
Figura 30 – Características de uma onda sonora.	71
Figura 31 – Exemplos de ondas sonoras: à esquerda, dentro do limite de distorção; à direita, onda sonora ultrapassando o limite de distorção (IAZZETTA, 2017).	71
Figura 32 – Em cima, a amostra de som gerada com cem instâncias e, em baixo, com duzentas instâncias.	73
Figura 33 – Em cima, a amostra de som gerada com trezentas instâncias e, em baixo, com quinhentas instâncias.	73
Figura 34 – Em cima, a amostra de som gerada com seiscentas instâncias e, em baixo com mil instâncias.	74
Figura 35 – Ponto de saturação aproximado em duas mil instâncias.	75
Figura 36 – Avaliação do ARSUS em um contexto de composição musical interativa e espacializada em tempo real.	77
Figura 37 – Percentual de respostas dos usuários à pergunta realizada acerca da es- pacialização sonora. Apuração feita através das respostas do formulário aplicado após o estudo.	78
Figura 38 – Percentual de satisfação dos usuários acerca de sua experiência de uso do sistema e de sua colaboração para uma composição coletiva. O eixo Y representa a quantidade de alunos e o eixo X, o nível de satisfa- ção/experiência relatados.	79
Figura 39 – Percentual de usuários que tiveram alguma dificuldade em utilizar a interface do Compomus.	80
Figura 40 – Avaliação do ARSUS aplicado à espacialização do jogo <i>Pac-Man</i>	81
Figura 41 – Funções inseridas no código original do jogo com as mensagens OSC a serem enviadas ao ARSUS.	82

Lista de Tabelas

Tabela 1 – Equações de codificação <i>Ambisonics</i> até a 3 ^a ordem (FARIA et al., 2005).	26
Tabela 2 – Comparação entre as técnicas de espacialização sonora.	40
Tabela 3 – Comparação entre os ambientes e linguagens de programação voltados para o processamento de Áudio.	41

1 Introdução

A forma de interagir e como desenvolver o projeto de som de produtos ou artefatos, são estudados pelo Projeto de Interação Sônica, ou, em inglês, *Sonic Interaction Design* (SID). SID é um campo de estudo emergente situado na interseção de exibição auditiva, design de interação, computação ubíqua e artes interativas.

A exploração da interação com o som baseado em SID é aplicada de diferentes formas em interfaces simples, como bolas com microfones, sensores de pressão, etc. Um exemplo dessas aplicações é o *Big Buoy*, uma boia com sensores de pressão e microfones embutidos, e o *Music Troll* (JENSENIUS; VOLDSUND, 2012), um “instrumento de instalação” independente com quatro cabeças que saem de uma caixa de madeira que foi desenvolvido para a interação com o som através do toque.

A sociedade contemporânea tem demandado diálogos com as novas tecnologias e, por este motivo, muitos artistas e projetistas exploram esse campo de maneira consciente, buscando diálogos dentro do que seria “a tensão da luta entre o humano e máquina num sistema aberto e instável” (BROECKMANN, 2009). Pesquisadores defendem, no campo conceitual, a necessidade de se criarem experiências lúdicas, artísticas e criativas, dentro das relações entre arte, ciência e tecnologia, como forma de aproximar as pessoas em geral de um debate sobre as nossas relações com as tecnologias, as máquinas, os ambientes artificiais e os fenômenos científicos.

Pesquisas baseadas nos conceitos de SID têm levado os autores a desenvolverem interfaces interativas, fazendo uso de tecnologias emergentes, como o software ReactiVision, que utiliza a visão computacional para visualizar *tags*. Com essa tecnologia (NAS; LOPES, 2016), experimentam vários conceitos do estudo em cenários diferentes, como performances solo, em grupo e colaborativas, apresentadas em escolas e exposições de arte.

A evolução dessas tecnologias tem proporcionado novas formas de colaboração neste universo de criação e execução musical. A música eletroacústica, uma das várias vertentes da música, mergulha profundamente nas potencialidades que a colaboração e a tecnologia possibilitam. Desde o seu surgimento no final dos anos 40, a música eletracústica demonstra ter uma “intimidade” com a tecnologia, seja ela na vertente acusmática, mista ou derivada da eletrônica em tempo real (ROCHA, 2010). A colaboração se tornou parte integrante da arte como fonte de originalidade para a composição de música eletroacústica, os artistas têm combinado as tecnologias emergentes com a arte para produzir novas expressões musicais (BLAINE; FELS, 2003). Várias pessoas contribuindo com sons diferentes em um espaço de interação resulta, em sua essência, em música produzida de forma colaborativa; ao interagirem umas com as outras, não é necessário palavras para se comunicar, o som é o meio de comunicação.

Em um outro cenário também artístico, como o cinema, o som é trabalhado com o propósito de fazer com que os espectadores imaginem que o que estão vendo é real, porém, o público tem dificuldade em dissociar a atuação do som das informações visuais as quais está submetido, resultando em uma “percepção do som como imagem”. Essa interação indissociável entre o som e a imagem é responsável pela imersão do espectador, envolvendo-o em um campo sonoro proporcionado pelos sistemas de áudio¹ multicanais e a espacialização sonora (COSTA, 2013).

As tecnologias atuais de realidade virtual requerem também o áudio em 3D, e a espacialização sonora tem sido bastante explorada em jogos como no clássico F.E.A.R (IGN US, 2006) já preparado para realidade virtual com uso de dispositivos como o InfinaDeck (Road to VR, 2012) e Vue VR (Vue VR, 2017), que trazem uma imersão ao jogador. Em todos esses casos a espacialização sonora é um pré-requisito para uma imersão completa.

O fenômeno da espacialização sonora se dá devido a habilidade do ser humano em determinar a localização da direção de um certo som. De acordo com (MOORE, 2003), essa capacidade funciona por meio de indicações auditivas que são percebidas pelos ouvidos. A diferença de tempo interaural (DTI) e a diferença de intensidade interaural (DII) são as principais indicações, e vem do fato do ser humano possuir dois ouvidos posicionados separadamente, o que faz com que o som chegue de forma diferente a cada um.

A espacialização sonora há muito tempo é usada no cinema, em que os filmes projetados possuem, em seu rolo, o áudio codificado para reprodução em sistemas de áudio multicanais e tudo segue roteiro determinado pelo projeto criado pelo projetista de som.

Todos os aspectos sonoros de uma cena de filme passam pelas mãos do projetista de som, ele é o ator central, responsável por dar emoção às imagens projetadas na tela do cinema ou movimento para os sons de uma cena. Em performances eletroacústicas é possível fazer a espacialização sonora em tempo real, porém o processo é manual e depende do projetista de som para executar o direcionamento das fontes sonoras. Existem também aplicativos que fazem a espacialização sonora em tempo real, mas na maioria dos casos, depende desse personagem. Isto é, a interação com o som já está pré-estabelecida, devendo seguir um roteiro ou estando sujeita a um personagem central (COSTA, 2013).

A motivação desta pesquisa é suprir a demanda por tecnologias que possibilitem a espacialização sonora em tempo real, proporcionando ferramentas para que produtores

¹ Há uma diferença entre som e áudio. Som é a propagação de uma onda mecânica; é uma onda longitudinal, que se propaga de forma circuncêntrica, apenas em meios materiais (que têm massa e elasticidade), como os sólidos, líquidos ou gasosos. O áudio, consiste na representação digital de uma onda sonora por meio de código binário. O processo que envolve, na captação ou gravação, a conversão do som analógico para digital (ADC, Analog to digital converter) e, na reprodução, a conversão do som digital para analógico (DAC, Digital to analog converter) permite que o som seja armazenado e reproduzido por meio de arquivos de áudio em diversos formatos, como WAV, AIFF, MP3, OGG, e de outros meios.

de filmes, jogos, realidade virtual, projetistas de som, compositores, regentes e artistas performáticos possam espacializar fontes sonoras sem a necessidade de conhecimentos profundos acerca de programação ou teoria do som.

Assim, o problema considerado para esta pesquisa é: Como dar suporte computacional à espacialização sonora realizada em tempo real em um espaço definido, considerando diferentes fontes sonoras sendo espacializadas por atores distintos de forma colaborativa, espontânea e aleatória, sem a necessidade da presença de uma figura centralizadora para orquestração.

Para guiar esta pesquisa foram elaboradas as seguintes questões:

- Como espacializar diferentes sons manipulados por diversos usuários em tempo real?
- O desafio, neste caso, seria criar uma estrutura computacional que identifique esses sons, considerando que as trajetórias percorridas por eles não devem seguir um comportamento predefinido. Os usuários devem estar livres para interagir com os sons e definir o trajeto que estes irão percorrer. O que nos leva a segunda questão de pesquisa.
- Como proporcionar uma interação dos usuários com o som? - Neste caso, essa interação não deve ser complexa e nem exigir esforço do usuário para entendê-la. Para projetar uma forma interação do usuário com o som de uma maneira facilitada, devem ser levados em conta a utilização de objetos do dia a dia. O usuário deve se sentir no controle de sua interação, e esta interação deve estar explícita, o que deve nos levar a considerar a próxima questão.
- Os usuários serão capazes de se sentir contribuindo em um ambiente de interação?
- O resultado gerado pela interação com o som deve ser uma composição, o que leva ao desafio de como proporcionar ao usuário a sensação de colaboração em um ambiente de interação.

Em vista das questões de pesquisa levantadas, a seguir estão definidos os objetivos do trabalho.

1.1 Objetivos

Proporcionar um suporte computacional para espacialização sonora em tempo real, escalável para utilização em múltiplos contextos.

Para atingir o objetivo geral foi necessário:

- Identificar o conjunto de requisitos necessários para espacialização sonora.
- Projetar um modelo interativo de espacialização sonora, escalável e sensível à localização.

- Definir métodos de avaliação dos resultados.

1.2 Metodologia

Esta pesquisa teve origem a partir da demanda de uma parceria entre a Faculdade de Artes e o Instituto de Computação (IComp), ambos da Universidade Federal do Amazonas (UFAM). A ideia desta parceria consiste em unir a arte, mais especificamente a música, com a tecnologia, abordando o problema da composição colaborativa de música eletroacústica com o uso de tecnologias emergentes.

Foi realizada uma revisão da literatura para conhecimento das tendências de estudos e o estado da arte na área de espacialização sonora, interação com o som, colaboração na música e interação humano computador, a qual retornou várias ferramentas, abordagens e novas possibilidades para o projeto, incluindo o campo de estudo Projeto de Interação Sônica e o *Design Thinking* (BONINI; ENDO, 1995), uma abordagem diferenciada para a condução de projetos que buscam soluções inovadoras.

Em seguida, foi executada uma busca por tecnologias e ferramentas disponíveis para o desenvolvimento de projetos áudio. Dentre as ferramentas disponíveis, está o Pure Data (PUCKETTE, 2016), um ambiente e linguagem de programação visual com paradigma de conexão de blocos de código, para facilitar o desenvolvimento de programas voltados à cena musical por quem não é, de fato, um programador. Os programas escritos em Pure Data são conhecidos como *patches*, e seus objetos são conhecidos como *subpatches* e abstrações. Outra ferramenta disponível e escrita para o Pure Data é o *Audience*, uma biblioteca para processamento de áudio espacial proposto por (FARIA, 2011).

O *Design Thinking* foi utilizado com o objetivo de gerar ideias de soluções para o problema da composição colaborativa de música eletroacústica. Essa abordagem considera quatro atividades principais, a saber: compreensão do problema, projeto de soluções, prototipação e implementação da melhor solução. As etapas de projeto de soluções e prototipação foram realizadas de forma iterativa, onde a cada projeto de solução era gerado um protótipo que, por sua vez, era avaliado através de estudos experimentais.

Durante processo de prototipação, verificou-se que a solução mais adequada (melhor solução) envolvia a espacialização sonora. Assim, a pesquisa mudou o foco, onde o problema a ser pesquisado deixou de ser somente o da composição colaborativa e passou para o suporte à composição musical colaborativa baseada em espacialização sonora em tempo real. Cada uma dessas atividades está descrita em detalhes no Capítulo 3.

Os protótipos desenvolvidos deram origem ao ARSUS, o *framework* que encapsula as dificuldades da espacialização sonora, sendo aplicável também em outros cenários. Uma vez desenvolvido, fez-se necessário avaliar o *framework* em diferentes contextos: análise do ponto de saturação, composição colaborativa de música eletroacústica, espacialização sonora de jogos. Cada avaliação está descrita em detalhes no Capítulo 5.

1.3 Sumário Estruturado da Dissertação

Capítulo 1 - Introdução à espacialização sonora, projeto de interação sônica, problemas, necessidades, definições dos objetivos e aplicações do trabalho desenvolvido.

Capítulo 2 - Apresenta a fundamentação teórica, indispensável para o entendimento do trabalho, e uma descrição dos trabalhos relacionados ao tema desta pesquisa.

Capítulo 3 - Descreve o processo de desenvolvimento da solução do problema de espacialização sonora através do Design Thinking.

Capítulo 4 - Apresenta, em detalhes, o resultado da evolução da solução para o problema de espacialização sonora. O ARSUS *Framework* surgiu a partir da implementação da melhor solução desenvolvida e descrita no Capítulo anterior.

Capítulo 5 - Apresenta a avaliação do framework proposto, bem como a sua aplicação em outros contextos.

Capítulo 6 - Apresenta as considerações da pesquisa realizada, assim como os resultados obtidos e perspectivas de trabalhos futuros.

2 Fundamentação Teórica

Este capítulo apresenta os conceitos e técnicas de espacialização sonora mais difundidos nos últimos anos. Em seguida, são apresentadas as tecnologias e linguagens de programação voltadas para o processamento de áudio mais difundidas da atualidade. O campo de estudos que guiou este projeto é então apresentado juntamente com seus conceitos e aplicações e, por fim, os trabalhos relacionados ao tema abordado são discutidos.

2.1 Espacialização Sonora

A habilidade de localização da direção do som funciona por meio de indicações auditivas que são percebidas pelo sistema auditivo. Essas indicações se dão pelo fato do ser humano possuir os ouvidos posicionados um em cada lado da cabeça, o que faz com que o som chegue de forma diferente a cada um. A diferença de tempo interaural (DTI) e a diferença de intensidade interaural (DII) são as indicações mais utilizadas.

A DTI está associada à diferença de tempo que o som leva para chegar, a partir de uma fonte sonora localizada em um espaço tridimensional, a cada um dos ouvidos. Um exemplo: se uma fonte sonora está mais próxima fisicamente de um dos ouvidos, ela chegará primeiro a esse mesmo ouvido e, em seguida, ao mais distante. Portanto, existe aí uma diferença de tempo entre o momento em que o som chegou ao ouvido mais próximo e o momento que chegou ao ouvido mais distante. Essa diferença de tempo é calculada pelo cérebro, que indica de onde vem a localização aproximada e tridimensional da fonte sonora.

A DII está relacionada à propagação do som em um ambiente, que pode chegar com diferença de intensidade aos ouvidos de um ser humano. Novamente, devido à diferença das distâncias envolvendo a fonte sonora e os ouvidos. A intensidade do som diminui de acordo com a distância percorrida, devido à perda de energia na propagação do som. A exemplo, em uma câmara de som tratada para haver baixa reverberação (múltiplas reflexões de uma onda sonora), o ouvido mais próximo da fonte sonora recebe a onda com uma intensidade maior que a do ouvido mais distante (MOORE, 2003), como é ilustrado na Figura 1.

Entende-se que o sistema auditivo, a partir dessas indicações e de outras mais específicas (como a ordem de chegada das ondas e o espectro), considerando um som complexo, determine a posição (direção e distância) da fonte mediante um sistema de votação (GERZON, 1974). Uma análise aprofundada dessas indicações, bem como de outros fatores e avanços na área sobre os mecanismos auditivos de localização espacial foram relatados por (BLAUERT, 1996).

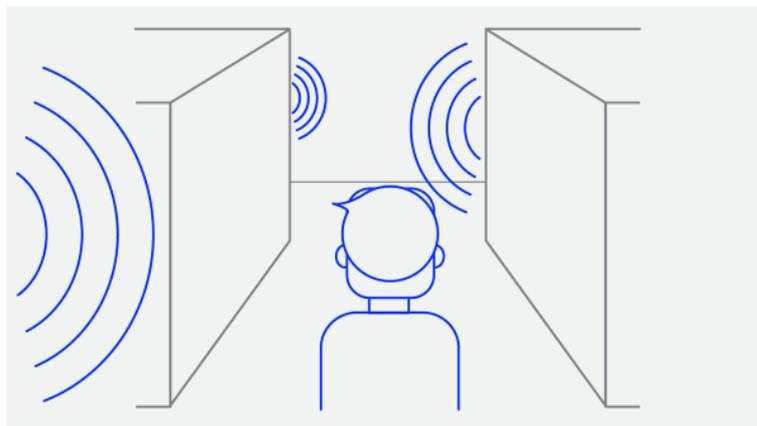


Figura 1 – Exemplo de fontes sonoras em disposições diferentes que chegam com intensidades e tempos diferentes a cada ouvido.¹

2.1.1 Espacialização Sonora na Música

A execução de peças musicais pode ser considerada espacializada, pois o ouvinte utiliza a sua percepção espacial para identificar cada fonte sonora. Sendo assim, ele consegue apontar a origem dos sons, desde um instrumento solista ou até uma orquestra. Usualmente, essa ambientação está relacionada à forma e ao local da propagação do som, dependendo mais de questões físicas do que artísticas.

Por exemplo, em uma orquestra, a escolha da posição dos instrumentos se dá pelo volume sonoro que cada instrumento emite, de forma que todos sejam ouvidos claramente. Em formações menores, como quarteto de cordas, esse posicionamento de fontes é irrelevante para o resultado final. Nos dois casos a percepção do público é que essas são uma única fonte sonora, vinda da frente, como mostrado na Figura 2.

Um dos primeiros a fazer uso da espacialização em suas peças musicais foi o compositor Adrian Willaert, que ainda no século XVI criou uma das primeiras peças utilizando o espaço como elemento composicional, fazendo uso de dois coros posicionados em lados opostos na Basílica de São Marcos, em Veneza.

Já em Meados do século XX, a espacialização sonora veio sendo explorada sistematicamente por compositores e regentes. Artistas como Stockhausen, Iannis Xenakis e Pierre Boulez, iniciaram suas experimentações com a espacialização utilizando instrumentos acústicos ou auxiliados pela revolução tecnológica da música eletroacústica (THOMAZ, 2007).

Uma obra eletroacústica é composta de sons previamente gravados, que são tratados posteriormente em um computador e trabalhados musicalmente para se construir um produto final, podendo, aliás, fazer uso de sons sintetizados ou da natureza (FRITSCH, 2008).

¹ Figura extraída de: <<https://medium.com/method-perspectives/interacting-with-the-virtual-a-mix-of-realities-5d83547c96d2>>

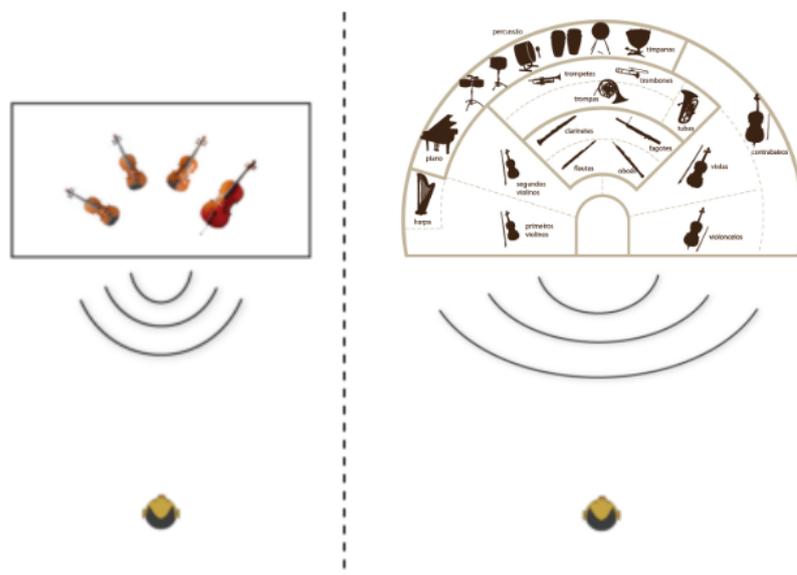


Figura 2 – Ambientação não espacializada: mesmo com o posicionamento no caso da orquestra, a fonte sonora será percebida como única pelo ouvinte, assim como no quarteto de cordas.²

Ao contrário de uma música composta para ser executada por instrumentos, a qual o compositor cria uma partitura que será interpretada e executada em uma apresentação, a música eletroacústica é produzida em estúdio (mediada por computador) e armazenada em algum tipo de mídia, como um CD ou mesmo o disco rígido do computador. Essa forma de música é também conhecida como acusmática (GARCIA, 1998).

Em uma apresentação de música acusmática não se tem músicos posicionados em um palco, a obra é executada em um arranjo de alto-falantes posicionados no local de apresentação e ao redor dos ouvintes. Na maioria das apresentações é o compositor quem controla uma mesa de som colocada ao centro do local da apresentação, processando e direcionando o som em tempo real para os alto-falantes e, desta maneira, moldando o espaço.

Desta forma, tem-se uma mobilidade do som ao entorno da plateia, criando assim um efeito de som envolvente. A plateia normalmente sente que está imersa em sons. A essa forma de controlar o direcionamento e a mobilidade dos sons em um espaço por meio de alto-falantes, dá-se o nome de difusão sonora, difusão eletroacústica ou mesmo espacialização.

A banda inglesa de rock Pink Floyd, procurando novas sonoridades, realizou experimentos com o som envolvente em seus shows. Uma forma rudimentar de mixagem foi montada de modo que cercasse o público com caixas de som e, a partir disso, efeitos sonoros controlados por um *joystick* eram projetados de um lado a outro ou em movimentos circulares. Com a evolução da computação e do processamento de sinais digitais, as possi-

² Figura Adaptada de (THOMAZ, 2007)

bilidades de manipulação do som cresceram vertiginosamente. Na Figura 3, equipamento à esquerda foi desenvolvido por Giuseppe Di Giugno no *Institut de Recherche et Coordination Acoustique/Musique* (IRCAM), que era composto por oito processadores digitais de áudio. Já o Matrix 32, à direita, controlava o tráfego de áudio entre os microfones, o 4X e os alto-falantes. Para controlar esse sistema de processamento de sinais de áudio, Miller Puckette, também pesquisador do (IRCAM), desenvolveu um software chamado Patcher³, que mais tarde viria a evoluir para os softwares MAX/MSP⁴ e no Pure Data.⁵

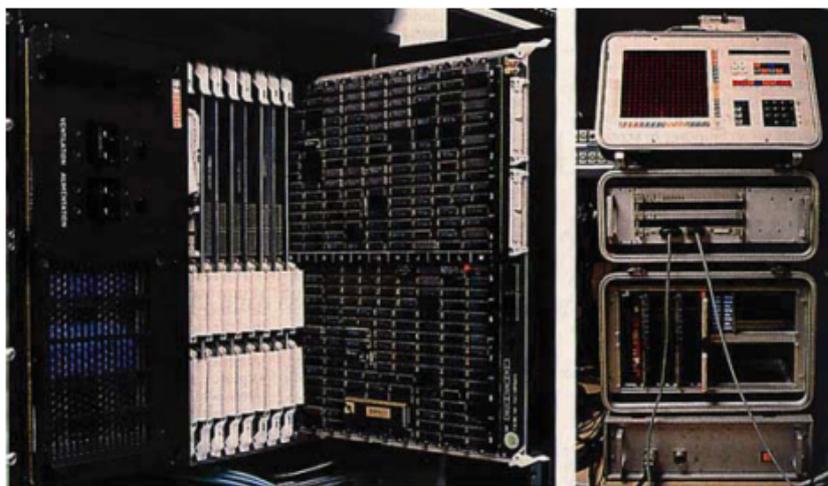


Figura 3 – Sistema de espacialização sonora desenvolvido para a reprodução de *Répons*, de Pierre Boulez.⁶

2.1.2 Técnicas de Espacialização

Várias abordagens e técnicas para a espacialização sonora foram desenvolvidas ao longo dos anos para tentar recriar campos sonoros tridimensionais, através de modelagens físicas e matemáticas (FARIA; ALVES, 2005).

Vários fatores são considerados em cada técnica desenvolvida, como custo computacional, exigência de equipamentos, número de alto-falantes, número de pessoas inseridas e qualidade de recriação da cena sonora. As técnicas mais tradicionais são listadas a seguir com uma breve descrição de seu funcionamento.

2.1.2.1 Binaural

A técnica de espacialização binaural é uma das mais simples formas de espacialização sonora, por possuir poucos requisitos para o seu funcionamento. Essa abordagem explora o fato do som chegar de forma diferente em cada ouvido, uma vez que cada um fica

³ Disponível em: <<http://msp.ucsd.edu/Publications/icmc88.pdf>>. Acesso em: 20 de out de 2017

⁴ Disponível em: <<https://cycling74.com/products/max/>>. Acesso em: 20 de out de 2017

⁵ Disponível em: <<https://puredata.info/>>. Acesso em: 20 de out de 2017

⁶ Figura Retirada de (THOMAZ, 2007)

posicionado em um lado da cabeça como citado anteriormente. O Sistema Binaural tem como base de processamento as funções de transferência conhecidas como *Head Related Transfer Function* (HRTF), Função de Transferência em Relação à Cabeça, em tradução livre.

A HRTF consiste em uma representação matemática da transformação sofrida pelo som desde a fonte até o ouvido. A geração de um som $X(t)$ em sua posição é ouvida como $X_L(t)$, no ouvido esquerdo, e $X_R(t)$, no ouvido direito, onde t é o tempo percorrido pelo som até chegar aos ouvidos; $h_L(t)$ e $h_R(t)$ são funções de resposta de impulso relacionadas à cabeça, em inglês, *head-related impulse response* (HRIR), dadas a partir da distância da fonte sonora até os ouvidos (SAVIOJA et al., 2015), vide Figura 4.

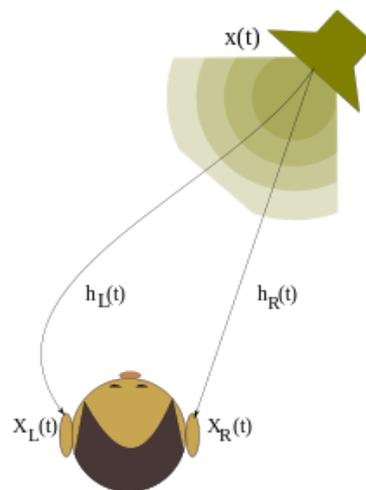


Figura 4 – Funções de transferências para os ouvidos direito e esquerdo.⁷

Ao contrário de outras técnicas que reproduzem uma cena sonora em três dimensões, o Sistema Binaural requer apenas dois alto-falantes ou um par de fones de ouvidos, no entanto, esta técnica possui dificuldades de desempenho quando aplicada a mais de um ouvinte com dois alto-falantes, tornando-se invasiva quando é utilizado com fones de ouvido.

2.1.2.2 Estéreo

Em 1931, o inventor Britânico Alan Blumlein patenteou o que seria o conceito de estereofonia, sistema que utiliza dois canais para reproduzir áudio. Essa técnica faz uso de dois alto-falantes separados por distâncias variáveis, cada um com o seu canal de reprodução. A divisão de som entre os dois canais é feita de forma artificial, utilizando um panorama por amplitude, ou fazendo uso de um arranjo de microfones. Desta maneira, cria-se uma “fonte sonora fantasma” entre os alto-falantes, o que aumenta a sensação de espacialização do som (FARIA; ALVES, 2005).

⁷ Figura Retirada de: <https://en.wikipedia.org/wiki/Head-related_transfer_function>

Vale destacar que existe uma grande diferença entre estereofonia e áudio Binaural no que tange à percepção espacial. Na estereofonia temos o fato de os dois ouvidos conseguirem detectar um som parecido chegando de direções diferentes, podendo ainda ter algum atraso ou uma diferença de amplitude entre ambos. Tendo em conta a diferença de tempo que o sinal sonoro leva para chegar aos ouvidos, pode-se perceber que a fonte sonora está em um local (uma posição fantasma) em relação ao primeiro sinal detectado. No áudio Binaural, o sinal de áudio é gerado por uma única fonte sonora e chega aos ouvidos em tempos e intensidades diferentes. Dependendo do ângulo e incidência do sinal, haverá um atraso de milissegundos entre os dois ouvidos, mas que é percebido pelo cérebro (RUMSEY; MCCORMICK, 2012); Figura 5.

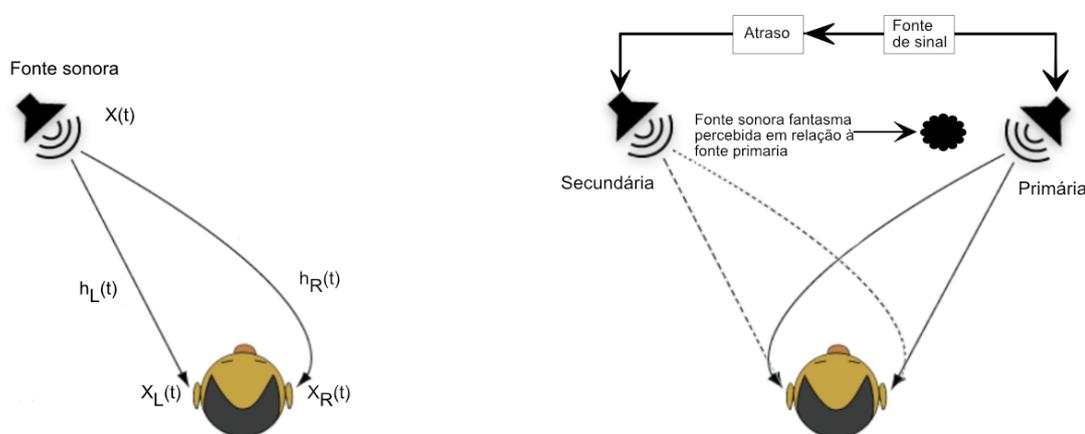


Figura 5 – À esquerda, Sistema Binaural partindo de uma única fonte sonora, e, à direita, dois alto-falantes simulam uma fonte sonora “fantasma”.⁸

Em 1971, a fim de ultrapassar as restrições do som estéreo, a indústria fonográfica criou o Som Quadrafônico, também conhecido como estéreo 4.0. O sistema faz uso de quatro canais de áudio, dos quais os alto-falantes utilizados são colocados nos quatro cantos de um ambiente, reproduzindo sons inteira ou parcialmente independentes em cada caixa de som. Contudo, essa técnica se mostrou um fracasso de mercado para época, devido a sua complexidade técnica para instalação e alto custo de implantação, sendo logo substituída pelo *Surround Sound* (WIKIPÉDIA, 2017).

2.1.2.3 Dolby Surround

Em sua evolução, o áudio saiu de um canal, foi para dois e, a partir daí, evoluiu constantemente até chegar aos sistemas multicanais, também conhecidos como sistemas de som envolvente ou *surround*. O *surround* expande o campo sonoro monoaural e estéreo

⁸ Figura adaptada de (RUMSEY; MCCORMICK, 2012)

para duas ou três dimensões. Nesse sistema, o ouvinte é envolto pelo campo sonoro, o que permite um ambiente de áudio muito mais complexo e realista (FARIA; ALVES, 2005).

Em meados da década de 70, a Dolby lançou o Dolby Stereo, um sistema de áudio envolvente para cinemas que era embutido nos rolos de filmes e eram codificados em um sinal ótico estéreo, com quatro canais de áudio: esquerda(L), centro(C), direita(R) e surround(S), sendo este padrão encontrado até hoje em muitos cinemas. Nos anos 80, a Dolby lançou o Dolby Surround, com a proposta de trazer o som envolvente para o ambiente doméstico, contudo, era muito caro e logo foi substituído pelo Dolby Pro Logic. Com a chegada da era digital, o Dolby Pro Logic recebeu atualizações e passou a chamar-se Dolby Digital. Esse sistema atende a norma ITU-RBS.755-1, que padronizou os sistemas de som *surround* com a configuração 5.1 (RECOMMENDATION, 1994); Figura 6.

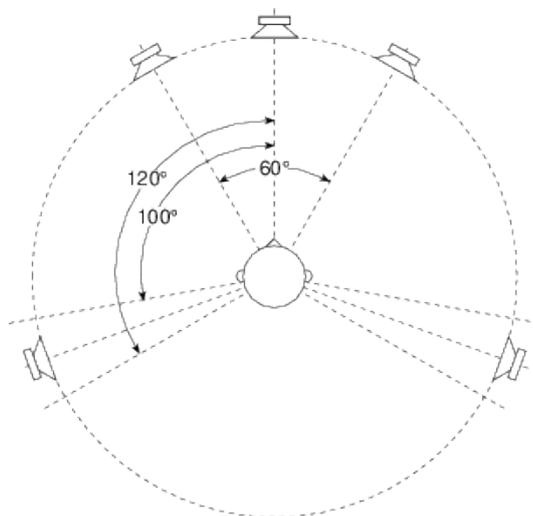


Figura 6 – Padrão surround 5.1, estabelecido pela norma (ITU-RBS.755-1).⁹

Os sistemas baseados na norma ITU-RBS.755-1 são, na realidade, um sistema estéreo com três canais e um canal *surround* traseiro usado para efeitos especiais. Não se consegue obter um campo sonoro estável ao entorno do ouvinte devido à existência de espaços vazios, pelos lados e atrás, na formação do campo sonoro, como pode ser verificado ainda na Figura 6.

2.1.2.4 Ambisonics

Ambisonics é uma associação de várias técnicas para reproduzir e gravar um campo sonoro. Seu objetivo principal é registrar todas as informações possíveis do campo sonoro de um ambiente para, em seguida, reproduzi-lo com o máximo de fidelidade em um conjunto de alto-falantes. Nas informações de reprodução, consta a localização e o espaço da fonte sonora, fornecendo a ilusão de uma imagem verdadeiramente tridimensional (GERZON, 1974).

⁹ Figura extraída de (RECOMMENDATION, 1994)

Ordem(m)	Canal <i>Ambisonics</i>	Equação de Codificação
0	W	$\sqrt{2} \cdot S$
1	X	$\cos \phi \cdot \cos \theta \cdot S$
	Y	$\text{sen } \phi \cdot \cos \theta \cdot S$
	Z	$\text{sen } \theta \cdot S$
2	R	$(3\text{sen}^2\theta - 1)/2 \cdot S$
	S	$\cos \phi \cdot \cos 2\theta \cdot S$
	T	$\text{sen } \phi \cdot \cos 2\theta \cdot S$
	U	$\cos 2\phi \cdot \cos^2 \theta \cdot S$
	V	$\text{sen } 2\phi \cdot \cos^2 \theta \cdot S$
3	K	$\text{sen}^2\theta \cdot (5 \cdot \text{sen}^2\theta - 3)/2 \cdot S$
	L	$8 \cdot \cos \phi \cdot \cos \theta \cdot (5 \cdot \text{sen}^2\theta - 1)/11 \cdot S$
	M	$8 \cdot \text{sen } \phi \cdot \cos \theta \cdot (5 \cdot \text{sen}^2\theta - 1)/11 \cdot S$
	N	$\cos 2\phi \cdot \text{sen } \theta \cdot \cos^2 \theta \cdot S$
	O	$\text{sen } 2\phi \cdot \text{sen } \theta \cdot \cos^2 \theta \cdot S$
	P	$\cos 3\phi \cdot \cos^3 \theta \cdot S$
	Q	$\text{sen } 3\phi \cdot \cos^3 \theta \cdot S$

Tabela 1 – Equações de codificação *Ambisonics* até a 3ª ordem (FARIA et al., 2005).

Durante os anos 70, o sistema de som envolvente mais utilizado era o quadrafônico, o qual fazia uso de quatro canais tanto para reprodução quanto para gravação, contudo, seu custo era muito alto e possuía muitos problemas técnicos, como os espaços entre os alto-falantes e a complexidade da gravação do campo sonoro. Existem, portanto, dois sistemas concorrentes no som *surround*: o DTS, técnica similar de auralização multicanal que também se baseia na norma ITU- R BS.775.1 para distribuição dos alto-falantes, e o Dolby. No entanto, ambos são bastante distintos no seu modo de funcionamento.

A maior diferença entre o quadrafônico, DTS e Dolby está no fato do sistema quadrafônico necessitar de gravar (ou produzir) os quatro canais discretos que serão reproduzidos por cada um dos alto-falantes.

No sistema *Ambisonics*, é gravado o campo sonoro em um determinado ponto que permite uma separação entre gravação e reprodução. Uma das vantagens desse sistema é sua versatilidade quanto à disposição dos alto-falantes, não necessitando ser reproduzido somente em 5.1, possibilitando outras configurações para a reprodução de primeira ordem, como: estéreo, quadrafônico, hexagonal e octogonal. Quando a ordem é superior, mais alto-falantes são necessários e sua disposição também sofre alterações (SANTOS, 2012).

A codificação em *Ambisonics* pode ser realizada através de síntese, na qual um sinal monofônico é posicionado no espaço por meio de manipulações com o ganho de cada canal. Esses ganhos se dão por equações de codificação *Ambisonics* que são provenientes diretos do formalismo matemático/físico da técnica.

As equações *Ambisonics* vão até a terceira ordem como lista a Tabela 1, onde θ

representa a elevação e ϕ representa a rotação da fonte em relação aos eixos, como mostra a Figura 7, e S é o sinal monofônico ao qual será aplicado o ganho.

Como o cálculo desses ganhos é extremamente complicado, o Ambisonics adota decodificação de canais através de matrizes de ganho pré-calculadas para facilitar o uso em configurações mais comuns, dependendo da disposição de alto-falantes empregada (THOMAZ, 2007).

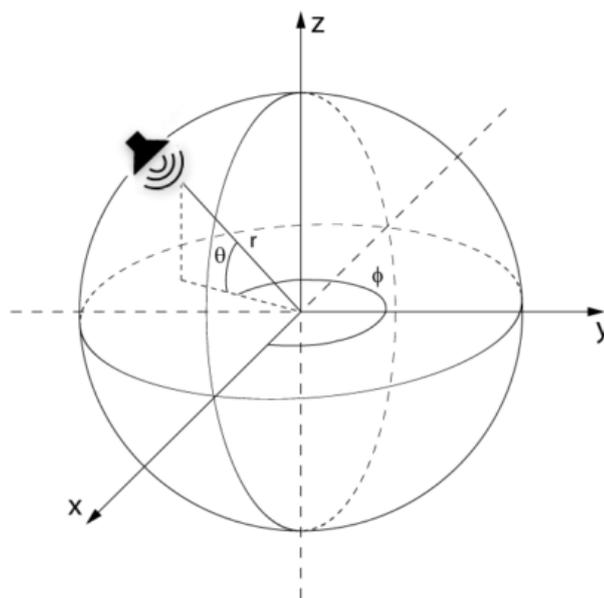


Figura 7 – Coordenadas esféricas utilizadas pelas equações de codificação *Ambisonics*.¹⁰

A tendência de pesquisa atual é focada na segunda e terceira ordem do sistema *Ambisonics*, na utilização em sistemas de realidade virtual (FARIA; ALVES, 2005) e na sua integração com outros sistemas e formatos de som envolventes, assim como transcodificações para os sistemas 5.1.

2.2 Tecnologias e Linguagens de Programação para o Áudio

Nesta seção, são apresentadas as principais tecnologias, linguagens e ambientes de programação voltados para a manipulação e processamento de áudio disponíveis atualmente, dentre elas estão o MAX, Pure Data, SuperCollider, ChucK e a biblioteca AUDI-ENCE4PD.

¹⁰ Figura Adaptada de (THOMAZ, 2007).

2.2.1 MAX e Pure Data

Miller Puckette escreveu originalmente Max no *Institut de Recherche et de Coordination Acoustique/Musique* IRCAM de Paris em meados dos anos 80. A codificação no MAX é realizada por meio de conexões gráficas de objetos que executam uma determinada função, criando um *patch*,¹¹ esses objetos atuam como programas autônomos (na realidade, são bibliotecas vinculadas dinamicamente), cada um dos quais pode receber entrada (através de uma ou mais entradas visuais), gerar saída (através de saídas visuais) ou ambas. Os objetos passam mensagens de suas saídas para as entradas de objetos conectados.

MAX é tipicamente aprendido através da aquisição de um vocabulário de objetos e como eles funcionam dentro de um *patch*; por exemplo, o objeto do “sfplay” funciona como um leitor do arquivo de áudio aberto pelo objeto “open”, e o objeto com a figura de alto-falante é um conversor de sinal digital para analógico responsável por externar o áudio gerado, como ilustra a Figura 8. A maioria dos objetos não são gráficos, consistindo apenas no nome de um objeto e vários argumentos-atributos (em propriedades de classe essencial) digitados em uma caixa de objetos. Outros objetos são gráficos, incluindo controles deslizantes, caixas de números, discagem, editores de tabelas, menus suspensos, botões e outros objetos para executar o programa de forma interativa.

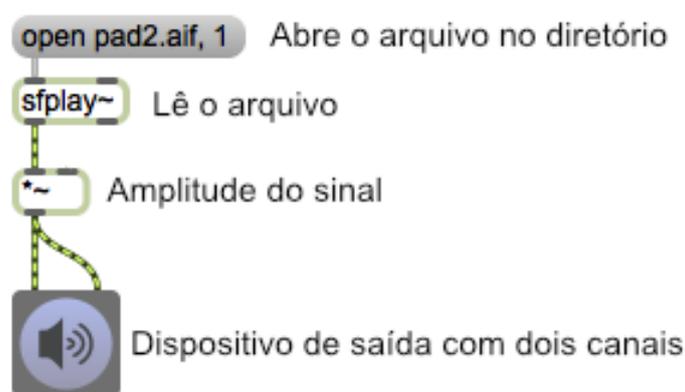


Figura 8 – Implementação de um reprodutor de áudio simples em MAX.

Posteriormente, Puckette lançou um programa de software gratuito totalmente redesenhado, em 1996, denominado Pure Data (PD) que, apesar de várias diferenças fundamentais do original do IRCAM, é superficialmente semelhante e continua sendo uma alternativa de fonte aberta para o MAX.

O Pure Data (PD), assim como o MAX, é uma linguagem de programação visual para criação de música eletrônica, música eletroacústica, música interativa e trabalhos multimídia. Apesar de Miller Puckette ser o seu autor principal, o PD é uma linguagem

¹¹ Os programas desenvolvidos nesta linguagem são conhecidos como *patches* e seus objetos *externals*, abstrações e *subpatches*, todos são executados dentro do ambiente que pode também invocar bibliotecas externas.

opensource com uma base de colaboradores extensa, que trabalham ininterruptamente em novos *externals*, abstrações e outras formas de *add-ons*, passíveis de estenderem as capacidades do sistema. O código do PD foi publicado sobre uma licença BSD e é compatível com os sistemas operacionais GNU/Linux, Mac OS, iOS, Android e Windows (PUCKETTE, 2016).

A dinâmica dos dados entre esses objetos funcionam de maneira similar ao MAX e dão-se por meio de mensagens que são enviadas da saída do bloco (*outlet*) para a entrada conectada do outro (*inlet*). As mensagens podem ser de sinais de áudio, que são processados quando o áudio está ligado, ou de controle (números, listas de números e strings), que são processadas quando ocorre algum estímulo no objeto.

A Figura 9 ilustra a implementação de um reproduzidor de áudio com repetição do som, onde o primeiro bloco “start” é o responsável por dar um comando de confirmação ou início que é enviado através de seu *outlet* para *inlet* do bloco de baixo. O bloco “open miracle.aiff, 1” carrega o som para a memória do computador e envia-o para a leitura através de seu *outlet* para o bloco abaixo.

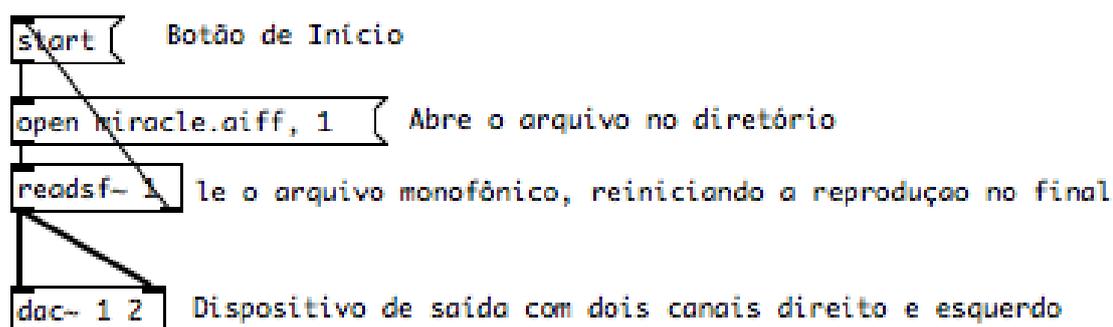


Figura 9 – Implementação de um reproduzidor de som em *loop* no Pure Data.

O bloco “*readsf 1*”, que possui um *inlet* e dois *outlets*, ao reproduzir o arquivo envia o sinal de áudio para o dispositivo de saída abaixo pelo *outlet* da esquerda e, ao final da reprodução do som, envia um sinal de confirmação de que terminou de reproduzir o som através de seu *outlet* da direita para o *inlet* do primeiro bloco proporcionando desta forma um *loop*. O Sinal sonoro que é enviado do bloco de reprodução para o dispositivo de saída não possui amplificação e é duplicado com o intuito de prover um sinal de áudio com dois canais.

2.2.2 AUDIENCE

O AUDIENCE for PD (Audience4PD) é uma biblioteca com itens proprietários feita de objetos Pure Data (externos) e abstrações (*patches*) concebidos para construir aplicações de áudio espaciais em geral. O software permite trabalhar de forma independente uma composição de cena sonora, sua renderização acústica, a codificação de seu

campo sonoro em um formato multicanal adequado e, finalmente, sua audição usando modos de alto-falante 2D/3D.

O *Audience* está baseado em uma abordagem modular de quatro camadas funcionais, ilustradas na Figura 10. A primeira camada é responsável pela composição de uma cena acústica. A segunda camada consiste em um simulador acústico onde são feitos tratamentos no sinal sonoro para melhorar a percepção espacial do som. A terceira camada faz a codificação do áudio e parâmetros de espacialização para enviar para a última camada. A quarta e última camada decodifica o sinal recebido e reproduz o campo sonoro nos canais correspondentes. Essa, por sua vez, requer equipamentos externos como sistemas de som multicanal para a propagação do som. O AUDIENCE possui uma versão básica gratuita chamada *OpenAudience* que não conta com os itens da segunda e primeira camada da versão completa (FARIA et al., 2005).

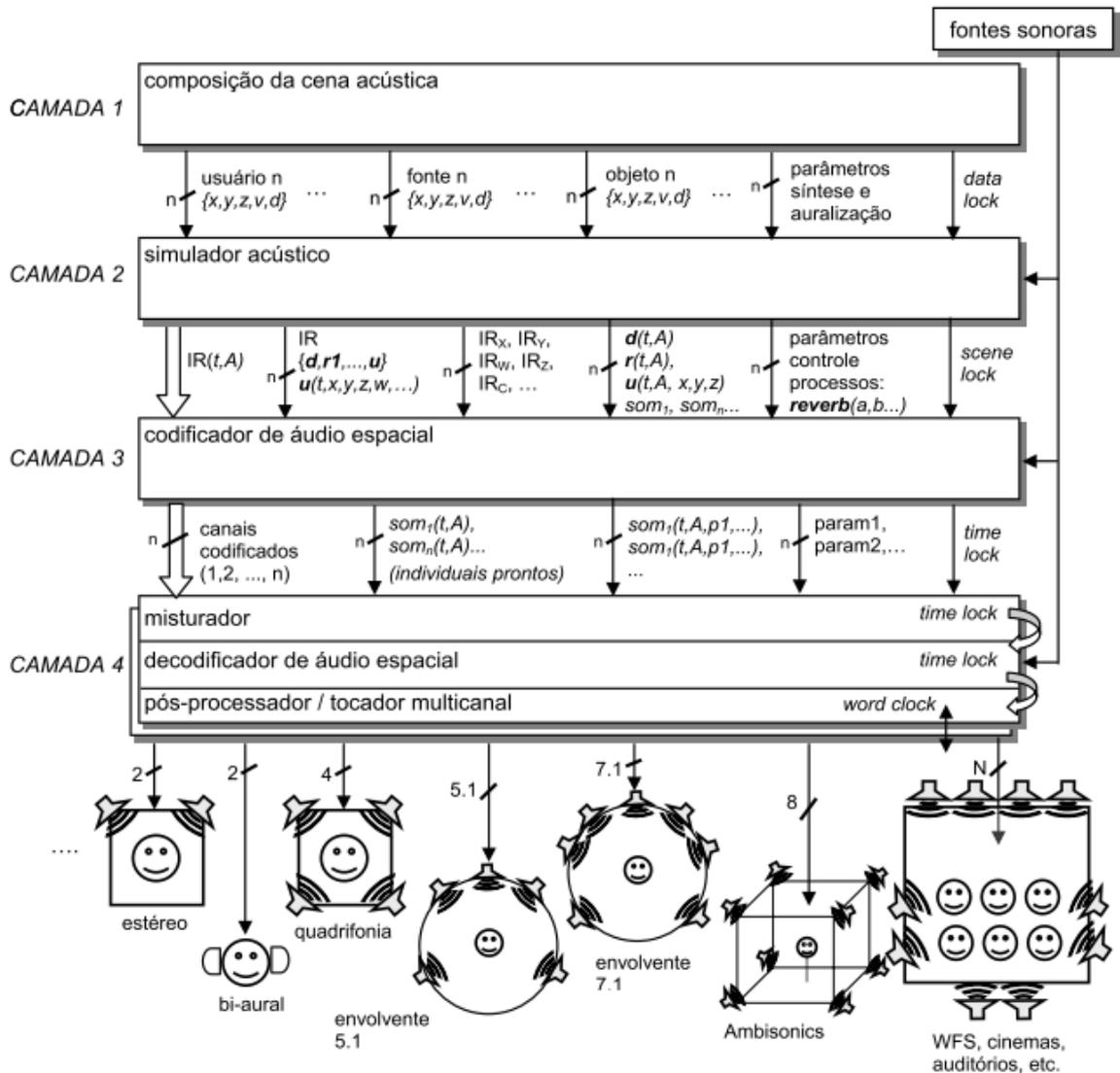


Figura 10 – Arquitetura de camadas do AUDIENCE (FARIA; ALVES, 2005).

2.2.3 SuperCollider

SuperCollider é uma linguagem e ambiente de programação, lançado originalmente em 1996 por James McCartney para síntese de áudio e composição algorítmica em tempo real. É um software livre, lançado sob os termos da *General Public License* (GNU). Naquela época, estava evoluindo para um sistema que seria usado e futuramente desenvolvido tanto por cientistas quanto por artistas que trabalham com som. Sua linguagem de programação dinâmica, expressiva e eficiente fez com que ele se tornasse um ponto importante para a pesquisa acústica e a programação interativa. A linguagem de programação SuperCollider é uma linguagem de escrita dinâmica orientada a objetos, com herança unilateral, coletor de lixo e programação funcional, similar a Smalltalk¹² com uma sintaxe similar a Lisp¹³ ou a linguagem de programação C (SuperCollider, 2017).

O Supercollider apresenta dois componentes dissociados: o servidor *scsynth* e o cliente *sclang*; o primeiro responsável por comunicações de rede e interpretação de código e o segundo responsável por gerar sinal de áudio através de um complexo e sofisticado motor de síntese, sendo que estes componentes se comunicam usando OSC (*Open Sound Control*).

A Figura 11 ilustra um trecho de código na linguagem de programação *SuperCollider* em que é criado um modulador de frequência senoidal, com amplitude do Ruído Rosa,¹⁴ combinado com outra senoidal, a qual depende da posição horizontal do ponteiro do mouse.

```
1 {
2     var x = SinOsc.ar(MouseX.kr(1, 100));
3     SinOsc.ar(300 * x + 800, 0, 0.1)
4     +
5     PinkNoise.ar(0.1 * x + 0.1)
6 }.play;
7
```

Figura 11 – Trecho de código na linguagem SuperCollider.

O SuperCollider permite que sejam criadas interfaces gráficas de usuário multiplataforma para aplicativos. A biblioteca de classes padrão com componentes de interface de usuário pode ser estendida por uma série de estruturas disponíveis. Para programação

¹² Smalltalk é a linguagem de programação que tornou popular o paradigma de orientação a objetos. Inspirou as linguagens mais populares da atualidade: Java, Ruby, Python, etc.

¹³ Lisp é uma família de linguagens de programação concebida por John McCarthy em 1958. Num artigo, ele mostra que é possível usar exclusivamente funções matemáticas como estruturas de dados elementares (o que é possível a partir do momento em que há um mecanismo formal para manipular funções: o Cálculo Lambda de Alonzo Church).

¹⁴ Ruído Rosa ou Ruído de 1/f é um sinal ou um processo onde o espectro de frequências como a densidade espectral de potência é inversamente proporcional à frequência do sinal.

interativa, o sistema oferece acesso programático a arquivos de código de texto a serem usados para gerar gráficos vetoriais algoritmicamente.

As características principais da linguagem SuperCollider são:

- *scsynth*, um servidor de áudio em tempo real, é o núcleo da plataforma;
- Granularidade, síntese de aditivos e subtrações, FM, síntese granular, FFT e modelagem física;
- Síntese em tempo real ou não (processar samples e gravar num arquivo, por exemplo);
- Programação textual (em oposição a, por exemplo, Pure Data);
- *Unit Generators* ("UGens") para análise, síntese e processamento. Permite a escrita de novos UGens em C++;
- Suporte a comunicação por MIDI e OSC;
- *scide*, um editor para *sclang* com um sistema de ajuda integrado;
- Suporte a entrada de vários periféricos (mouse, teclado, *joysticks*, acelerômetros, etc).

2.2.4 Chuck

Chuck é uma linguagem de programação de propósito geral voltada para o nicho de programação sônica. Um programa Chuck é um simples *script* que roda em uma máquina virtual. O grande atrativo é o suporte à execução concorrente dos programas facilitada pela máquina virtual, que também se encarrega do sincronismo temporal (*strongly-timed*) (ZAWACKI, 2017).

Existem também várias facilidades para modificar os programas que estão atualmente na máquina virtual sem que ela tenha que ser reiniciada, o que permite que seja feita programação *on-the-fly* ou *live-coding*.

Foi criada no *Princeton Sound Lab* por Ge Wang e Perry R. Cook e é usada exaustivamente na *Princeton Laptop Orchestra (Plork)*. Chuck é distribuída sob uma licença *opensource* e o código está disponível no site oficial.¹⁵

As características principais da linguagem ChuK são:

- Programação Concorrente;
- Mecanismo de sincronismo temporal embutido na máquina virtual;

¹⁵ Disponível em: <<http://chuck.cs.princeton.edu/>>

- Síntese em tempo real ou não (processar samples e gravar num arquivo, por exemplo);
- Programação textual (em oposição a, por exemplo, Pure Data);
- *Unit Generators* (geradores de função) padrão e extensíveis;
- Suporte a comunicação por MIDI e OSC;
- Suporte a entrada de vários periféricos (mouse, teclado, *joysticks*, acelerômetros, etc).

Na Figura 12, o código na linguagem ChuckK, referente à linha 2, cria um oscilador e guarda na variável (s). A linha 3 é responsável por mudar a frequência da onda para (440Hz). A linha 5 do código executa por dois segundos o som a ser gerado e, por último, a linha 7 imprime “Hello World!” na tela.

```
1 // SinOsc é um oscilador de onda senoidal
2 SinOsc s => dac;
3 440 => s.freq;
4
5 2::second => now;
6
7 <<< "Hello World!">>>
8
```

Figura 12 – Trecho de código na linguagem ChuckK.

Ao rodar este exemplo é possível ouvir uma onda senoidal na frequência da nota Lá padrão (440Hz), parecido com um assovio bem suave.

ChuckK atualmente se encontra na versão (1.2.1.3), e possui distribuições para as plataformas Linux, MacOS e Windows.

Também pode ser encontrado, no site oficial, um software chamado MiniAudicle, uma pequena IDE para escrever código ChuckK (contando com *syntax highlighting*), executando-o na máquina virtual gerenciando os diversos programas que rodam concorrentemente.

2.3 Projeto de Interação Sônica

A forma de interagir e como desenvolver o projeto de som de produtos ou artefatos, são estudados pelo Sonic Interaction Design (SID) ou, em português, Projeto de Interação Sônica, que é um campo emergente de estudo que está situado na interseção de exibição auditiva, projeto de interação, computação ubíqua e artes interativas. SID pode

ser usado para descrever a prática e pesquisa em qualquer uma das várias funções que o som pode desempenhar no ciclo de interação entre usuários, artefatos, serviços ou ambientes. Aplicações que vão desde a funcionalidade crítica de um alarme até o significado de uma criação Musical (ROCCHESSO et al., 2008).

SID explora maneiras pelas quais o som pode ser usado para transmitir informações, significados, qualidades estéticas e emocionais em contextos interativos. Um dos objetivos finais do SID é a capacidade de fornecer diretrizes de projeto e avaliação para produtos interativos com um comportamento sonoro saliente. O SID aborda os desafios de criar interações sonoras interativas e adaptativas que respondam continuamente aos gestos de um ou mais utilizadores. Ao mesmo tempo, o SID investiga como os gestos projetados e o *feedback* sonoro são capazes de transmitir emoções e engajar experiências expressivas e criativas (FRANINOVIĆ; SERAFIN, 2013).

O SID também visa identificar novas funções que o som pode desempenhar na interação entre usuários e artefatos, serviços ou ambientes. Ao explorar tópicos como a experiência multisensorial com artefatos sonoros, ilusões perceptivas, som como meio de comunicação em um *loop* de percepção-ação e aprendizado sensorio-motriz através do som, os trabalhos baseados em SID estão abrindo novos domínios de pesquisa e prática para engenheiros e projetistas de som, interação e projetistas de interface, artistas de mídia e projetistas de produtos, entre outros.

Na computação sonora e musical, os pesquisadores têm se afastado da mera reprodução de engenharia de instrumentos musicais existentes e sons cotidianos em um contexto passivo para investigar princípios e métodos de forma a auxiliar na concepção e avaliação de sistemas interativos sonoros. Essa é considerada uma das áreas mais promissoras para a investigação e experimentação (SERRA; WIDMER; LEMAN, 2007).

Além disso, a concepção e implementação de novas interfaces para controlar tais sons, juntamente com a capacidade de aumentar instrumentos musicais existentes e objetos cotidianos com sensores e *feedback* auditivo, é atualmente uma área ativa de exploração na comunidade *New Interfaces for Musical Expression* (NIME) (COOK, 2001).

Hoje muitos sons, para os produtos, estão sendo projetados. Isso inclui tanto os sons que são produzidos através de fenômenos físicos quanto os sons que são criados digitalmente. Como exemplo de ambos os tipos, a manipulação física de materiais e o ajuste fino de componentes internos têm sido utilizados, por exemplo, para criar o som distinto do motor da marca Harley Davidson.

Os projetistas de som se baseiam nesse campo de estudo para desenvolver os sons dos produtos. Um clássico exemplo da aplicação do SID foram as versões posteriores ao primeiro modelo do iPod que utilizou, como dispositivo de entrada, uma roda de rolagem mecânica: uma roda que se gira para permitir a rolagem entre os itens do menu. As versões correspondentes do iPod substituíram a roda mecânica pela roda de clique: uma superfície sensível ao toque na qual os usuários deslizam o dedo no sentido horário e no

sentido anti-horário, como se estivesse em uma roda móvel, como ilustra a Figura 13.



Figura 13 – iPod *Classic* com roda mecânica e iPod com sensor tátil *click wheel* e *feedback* sonoro.

Um elemento que foi introduzido na roda de clique é o *clicker*: um clique de som que fornece *feedback* para o movimento entre os itens do menu. Esse recurso dá uma sensação tátil à roda de clique (uma ilusão pseudo-háptica), um pouco semelhante à mesa giratória em telefones antigos, tornando a rolagem mais expressiva e mais informativa. Uma vez que a rolagem reage à aceleração (quanto mais rápido você interagir com a roda de clique, mais rápido você percorre os itens do menu), o *clicker* fornece informações que não são evidentes a partir da ação de rolagem em si. O som de clique é o único som produzido pelo iPod fora dos fones de ouvido e é gerado através de um pequeno alto-falante piezoelétrico dentro do dispositivo (HERMANN; HUNT; NEUHOFF, 2011).

2.4 Trabalhos Relacionados

Em (FUSHIMI et al., 2017) os autores usam uma técnica chamada correspondência de modalidade cruzada, a interação entre duas ou mais modalidades sensoriais para criar uma experiência envolvente para o usuário. O *atmoSphere* é um sistema que fornece aos usuários experiências de música imersivas usando o áudio espacial e o *feedback* tátil.

Nesse trabalho os autores se concentraram na modalidade cruzada de sensações auditivas e táteis para aumentar o ambiente sonoro. O *atmoSphere* consiste em uma música espacializada e um dispositivo em forma de esfera que fornece *feedback* tátil, proporcionando aos usuários a imaginação de um ambiente sonoro grande e, ao mesmo tempo, a capacidade de sentir a interação desse ambiente na palma de suas mãos.

No *atmoSphere*, a música utilizada é composta por áudio Binaural previamente gravado de tipos de sons variados, como palmas de mão, voz de sussurros e sons de água. Esses sons foram registrados de 8 direções; a localização deles está ligada ao *feedback* tátil para que os usuários possam sentir o movimento representativo do som, como mostra a Figura 14.



Figura 14 – *atmoSphere* sendo utilizado no teste de experiência do usuário (FUSHIMI et al., 2017).

Baseados fortemente nos conceitos de SID, os autores em (NAS; LOPES, 2016) desenvolveram interfaces interativas para o software *ReactiVision*. Esse software permite sua adaptação em diferentes contextos, e foi criado inicialmente para fazer parte do sistema *Reactable*, um instrumento que ficou conhecido em 2007 através da cantora islandesa Björk. Por meio do seu sistema de reconhecimento ótico, o *ReactiVision* permitiu que a *performer* pudesse tocar sua programação sonora de dentro D'Água; Figura 15.

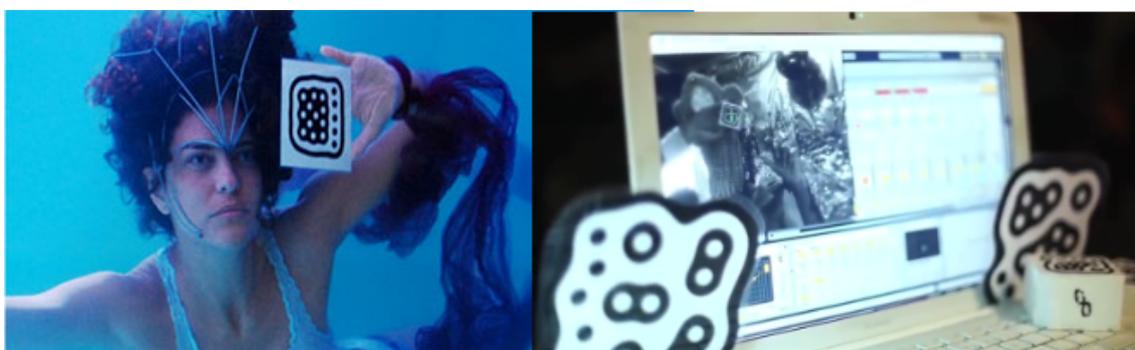


Figura 15 – Artista em uma piscina carregando as imagens a serem lidas pelo sistema que, por sua vez, executa o som; Cubos sonoros sendo testados em frente ao computador (NAS; LOPES, 2016).

Ainda em (NAS; LOPES, 2016), um outro experimento consiste em um ambiente descontraído, um ateliê coletivo aberto ao público. O Projeto de Som desenvolvido pode ser

acessado pelas imagens desenhadas à mão e reconhecidas pelo *Reactivision*, manipulando uma sequência de sons que se multiplicam a partir de uma frequência fundamental. Dessa maneira, as pessoas puderam manusear os cubos em frente a câmera do computador e lançar sons para o ambiente; Figura 15.

Este artigo apresenta *Trajectoires*, um aplicativo *Web* que funciona em computadores de mesa, *smartphones* ou *tablets* que permite aos compositores desenhar e editar trajetórias para controlar a espacialização do som com interações com base em toque na tela. A Figura 16 mostra um usuário que reproduz três trajetórias que controlam movimentos de fontes de som (GARCIA; FAVORY; BRESSON, 2016) .



Figura 16 – Reproduzindo três trajetórias e controlando três fontes sonoras (GARCIA; FAVORY; BRESSON, 2016).

O objetivo dessa pesquisa é projetar ferramentas interativas capazes de suportar trabalhos de compositores profissionais com espacialização de som. Os autores se concentram em dispositivos móveis que apresentam modalidades de entrada de toque e gesto, permitindo mover-se no espaço a ser renderizado para suportar uma configuração rápida e avaliação de cenas de som espacial.

Trajectoires permite aos compositores desenhar, reproduzir e editar trajetórias de fontes de som em dispositivos móveis. Enquanto o usuário está desenhando ou reproduzindo trajetórias existentes, o aplicativo envia mensagens Open Sound Control (OSC, 2016) para controlar remotamente qualquer mecanismo de espacialização de som, como *Spat* (CARPENTIER, 2015). O usuário pode mover-se livremente no espaço de interação enquanto usa o aplicativo para avaliar o resultado musical. O aplicativo também suporta a configuração da orientação das fontes de som manipulando fisicamente o dispositivo móvel.

Em (ASKA; RITTER, 2016), os autores estudam o uso da captura de movimento para controlar a espacialização sonora e os parâmetros para difusão multicanal em tempo

real. Nesse trabalho, vários programas foram desenvolvidos para o software MAX¹⁶ para facilitar o controle gestual da espacialização. Os sistemas de rastreamento de movimento usando câmera e dispositivos periféricos como o *Leap Motion*¹⁷ são explorados como meios viáveis e expressivos para fornecer a localização do som. Desta forma, permitindo que o usuário do sistema use o movimento através do espaço para controlar a localização do som em um ambiente. Ainda são discutidas três implementações diferentes: uma abordagem derivada de práticas de difusão do som, uma abordagem usando sonificação e uma abordagem na qual os gestos que controlam a espacialização são o foco. Essas abordagens estão fortemente ligadas a duas das mais importantes linhas de pesquisa da prática e performance de música eletroacústica e informática: primeira, a deslocação geográfica entre fontes sonoras e o som real percebido, segunda, a deslocação da casualidade física do som.

A exploração de formas diferentes de interação com o som baseadas em projeto de interação sônica são aplicadas nesse trabalho. O desenvolvimento de interfaces simples, como bolas dotadas de microfones ou sensores de pressão para interação de crianças com som, é o foco principal desse estudo. Um dos resultados da pesquisa é o *Music Troll*, um “instrumento de instalação” independente com quatro “cabeças” saindo de uma caixa de madeira; Figura 17.



Figura 17 – À esquerda, *Big Buoy* em uma interação com crianças e, à direita, *Music Troll* em uma Performance para o público (JENSENIUS; VOLDSUND, 2012).

A caixa contém todos os componentes eletrônicos (computador, placa de som, amplificador, alto-falantes); as quatro “cabeças” penduradas fora da caixa são suportadas por braços, cada uma com uma forma e interação sonora diferentes (JENSENIUS; VOLDSUND, 2012).

Nesse trabalho foi feita uma pesquisa de técnicas para sonorização espacial musical e o desenvolvimento de um sistema de espacialização sonora sobre uma arquitetura de áudio multicanal (AUDIENCE) para a auralização imersiva. A técnica de auralização *Ambisonics* foi escolhida para a utilização nesse trabalho devido a sua fácil implemen-

¹⁶ Disponível em: <<https://cycling74.com/products/max/>>. Acesso em: 20 de out de 2017

¹⁷ *Leap Motion* consiste em um pequeno dispositivo com um sensor capaz de captar movimentos dos 10 dedos das mãos do usuário.

tação, baixo custo computacional, configuração flexível de alto-falantes, capaz de suportar algumas pessoas simultaneamente e boa direcionalidade das fontes sonoras virtuais. Utilizou-se o Pure Data como plataforma para a construção dos blocos de software do sistema e suas conexões. O objetivo principal do sistema é espacializar fontes sonoras previamente gravadas através do posicionamento virtual dessas fontes e do usuário, podendo este movimentar essas fontes e sua posição durante a execução da peça (THOMAZ, 2007).

Apesar dos trabalhos relacionados acima terem alguns atributos em comum, interação com o som e a espacialização de fontes sonoras em tempo real, nenhum destes pode, ao mesmo tempo, ser escalável, proporcionar uma flexibilidade de configurações de alto-falantes, ser utilizado por vários usuários, permitir controle total e individual da espacialização sonora à vários controladores em tempo real e ser aplicável a vários contextos. Sendo assim, este trabalho tem como objetivo suprir a demanda por uma ferramenta que contemple todos estes atributos.

2.5 Conclusão

Dentre as técnicas para a espacialização sonora disponíveis atualmente, as mais difundidas foram apresentadas durante este capítulo. A escolha de uma técnica de auralização depende do foco do projeto desenvolvido, podendo ser destacadas algumas premissas que levam à escolha da técnica mais adequada ao projeto, como:

- Realismo na recriação da paisagem sonora ou precisão do efeito da espacialização;
- Quantidade de usuários utilizando o sistema simultaneamente;
- Facilidade de implementação do sistema, se a montagem de *hardware* e a programação do *software* são complexos;
- Custo computacional, ou seja, se a técnica exige grande processamento computacional em consequência dos cálculos para a reprodução da onda sonora;
- Custo de equipamentos: necessidade de alto-falantes, amplificadores, computadores e placas de áudio para a montagem do sistema.

Todas as técnicas de espacialização sonora (Binaural, Estéreo, Dolby Surround, Ambisonics) foram avaliadas por (THOMAZ, 2007) por meio de uma escala de um a cinco, sendo cinco a situação mais favorável, tendo como critérios os fatores acima; resultados podem ser conferidos na Tabela 2. Ao levar em consideração os requisitos da aplicação a ser desenvolvida, a tabela auxilia na escolha da técnica mais adequada para o contexto do projeto.

Os ambientes e as linguagens de programação apresentados neste capítulo estão dentre os mais utilizados pela comunidade de desenvolvedores de aplicações voltadas para

Técnica de Espacialização	Realismo	Quantidade de Usuários.	Facilidade de Implementação.	Custo Computacional	Custo de Equipamentos
Binaural	2 Boa reconstrução da onda se utilizadas boas HRTFs.	1 Suporta apenas um usuário.	4 HRTFs prontas para a utilização. Porém, complicadas no caso de HRTFs personalizadas.	3 No caso de síntese exige um processamento razoável.	4 Necessita apenas de fones de ouvido.
Estéreo	2 Sem os problemas de interferência do estéreo comum, mas só reproduz som frontal.	1 Suporta apenas um usuário.	4 Montagem simples, requer cálculo de filtros.	3 Processamento de filtros anti-interferência.	4 Necessita de apenas dois alto-falantes.
<i>Dolby Surround</i>	1 Reconstrução com problemas nas laterais e atrás.	2 Suporta poucos usuários.	5 Codificadores amplamente disponíveis.	4 Existem <i>hardware</i> e <i>software</i> específicos.	4 Equipamento 5.1 e <i>receiver</i> amplamente disponível.
<i>Ambisonics</i>	5 Ótima reconstrução da onda para altas ordens.	4 É capaz de suportar vários usuários simultaneamente.	3 Montagem depende do número e disposição dos alto-falantes.	3 Não requer grande processamento.	3 Depende do número de alto-falantes.

Tabela 2 – Comparação entre as técnicas de espacialização sonora.

o áudio. Apesar da linguagem MAX ser similar ao Pure Data e SuperCollider ter características parecidas com a linguagem Chuck no comparativo ilustrado pela Tabela 3, o que vai determinar a escolha de uma delas são os objetivos do projeto. Contudo, vale destacar alguns fatores que são determinantes, como:

- Uso de software gratuito *open source* ou software proprietário;
- Facilidade na implementação, pois tanto a linguagem de programação textual padrão tem vantagens quanto à incorporação do software produzido como *plugin* ou *add-on* quanto a linguagem de programação visual tem a vantagem do aprendizado rápido da sintaxe, além de possuir muitos módulos ou blocos de código prontos;
- Performance, se projeto requer a resolução de problemas complexos com múltiplas instâncias sendo executadas ao mesmo tempo;
- Multiplataforma, se o projeto requer a execução em várias plataformas;
- Necessidade de conhecimento em programação, pois as linguagens visuais têm a vantagem de não requererem um profundo conhecimento de codificação, já as linguagens textuais requerem uma experiência nesta área.

Como discutido acima, as decisões de escolha tanto para a técnica de espacialização sonora quanto para o ambiente e linguagem de programação a serem utilizadas no projeto dependem de alguns fatores. Para esta pesquisa, o ambiente e a linguagem de programação escolhidos foram o Pure Data, pela simplicidade e rapidez na aprendizagem da linguagem, além de ser multiplataforma. Quanto a técnica de espacialização, foi escolhida a *Ambisonics*, devido a maior flexibilidade na configuração de alto-falantes e número de utilizadores simultaneamente suportados.

<i>Ambiente/ Linguagem de programação</i>	<i>Tipo de programação Textual/Visual</i>	Programação Concorrente/ Paralela	Síntese em tempo real	Comunicação externa MIDI/OSC	Suporte a periféricos Mouse/Joystick	Suporte a bibliotecas e ou funcionalidades de terceiros
Pure Data	Visual	Paralela	Sim	MIDI e OSC	Sim	Sim
SuperCollider	Textual	Paralela	Sim	MIDI e OSC	Sim	Sim
ChucK	Textual	Concorrente	Sim	MIDI e OSC	Sim	Sim
MAX/MSP	Visual	Paralela	Sim	MIDI e OSC	Sim	Sim

Tabela 3 – Comparação entre os ambientes e linguagens de programação voltados para o processamento de Áudio.

No que diz respeito ao *Sonic Interaction Design* (SID), ou Projeto de Interação Sônica, este é o ponto de partida desta pesquisa, e guia para compreender o processo de sonificação e desenvolvimento de interfaces e ambientes interativos. O SID deve ser considerado pré-requisito, ou ponto de partida para pesquisadores que pretendem desenvolver projetos focados na sonificação de artefatos e interação com o som. O SID aumenta as possibilidades de desenvolvimento de novas tecnologias, formas de interação com o som e experiências multissensoriais criativas e expressivas para seus utilizadores.

3 Processo de Design Thinking na Composição Musical Espacializada

O *Design Thinking* (DT) é um enfoque humanista de inovação e criatividade que surge de uma visão multidisciplinar alicerçada nos princípios de engenharia, design, ciências sociais, artes e descobertas do mundo corporativo (PLATTNER; MEINEL; LEIFER, 2010). Em uma tradução mais literal, DT se refere à maneira do designer (neste caso, o profissional que está buscando soluções de qualquer natureza), no meio corporativo, pensar utilizando o pensamento abduutivo, ou seja, um tipo de raciocínio “fora da caixa”. Dessa forma, ele busca formular questionamentos através da compreensão dos fenômenos, com observação do universo do problema (MJV, 2017).

Segundo (COOPER; JUNGINGER; LOCKWOOD, 2009), o DT é uma ferramenta que proporciona a imaginação de soluções inovadoras, o pensar através do processo de design, assim como geração de produtos, experiências reais e serviços. Essa abordagem é formada por um processo não linear, alternado e que é desenvolvido a partir do trabalho colaborativo, do entendimento da necessidade do outro, da geração rápida de ideias e da criação e avaliação de protótipos (IDEO, 2017).

(CAVALCANTI; FILATRO, 2017) propõem uma abordagem baseada em vertentes existentes em DT com uma aplicação colaborativa, flexível, interativa e alinhada com propostas da IDEO e da *Design School* (PLATTNER, 2017). Ao adaptar o DT para o campo educacional, os autores têm o intuito de proporcionar aos educadores ferramentas para inovar e solucionar problemas de forma criativa. Nessa proposta, os autores definem quatro etapas que devem ser seguidas para o desenvolvimento de uma pesquisa ou produção de um artefato baseado em DT: compreender o problema, projetar soluções, prototipar e implementar a melhor solução.

Dada a eficiência na resolução de problemas complexos com foco em soluções inovadoras, com flexibilidade, o DT foi adotado como abordagem para a solução do problema de composição colaborativa identificado através da parceria do Instituto de Computação e Faculdade de Artes, a qual visava a integração de tecnologias emergentes com a arte, em especial a música. Esse problema está melhor descrito na seção seguinte. As seções posteriores deste capítulo estão organizadas com base nas demais etapas do DT, conforme mencionado anteriormente.

3.1 Compreendendo o Problema

Esta pesquisa teve origem a partir da demanda de uma parceria entre a Faculdade de Artes (FAARTES) e o Instituto de Computação (IComp), ambos da Universidade

Federal do Amazonas (UFAM). A ideia dessa parceria foi de unir a arte e a tecnologia, abordando o problema da criação de paisagens sonoras¹ para permitir que as pessoas possam brincar com os sons de forma colaborativa, gerando uma composição resultante destas interações, ou de maneira mais formal, uma composição de música eletroacústica com o uso de tecnologias emergentes.

Após quatro reuniões, para a discussão do problema no contexto musical, verificou-se a necessidade de dar suporte computacional a um ambiente de interação entre as pessoas e seus respectivos sons.

Então, como um problema inicial, foi considerada a seguinte questão: **Como dar suporte computacional a um ambiente no qual as pessoas possam interagir com outras, com o próprio ambiente e com sons? Isto é, que estruturas podem ser criadas para promover essa interação, de forma que o resultado desta colaboração resulte em uma composição musical?**

Para compreender melhor o problema foi organizado, na Faculdade de Artes, um *workshop* com especialistas na área de música para a apresentação de tendências artísticas, fundamentos teóricos da música, conceitos de música contemporânea, música colaborativa e de composição de música eletroacústica. Em seguida, foi realizada uma pesquisa na literatura para entender como são geradas as paisagens sonoras, como produzir a combinação de sons, como se dá a colaboração no âmbito musical, como projetar a interação com o som e quais são as tecnologias empregadas para proporcionar todos estes processos.

Sendo assim, foram organizados todos os conhecimentos prévios e adquiridos durante a etapa anterior, para em seguida ser feita a análise dos dados coletados na revisão da literatura. Definido o problema em ambas as esferas, tanto na musical quanto na computacional, seguimos para a próxima etapa de desenvolvimento segundo os conceitos de DT, projetar soluções.

3.2 Projetando Soluções

Para projetar as soluções foram levantadas algumas questões acerca do problema de espacialização sonora, como:

- O espaço deve ser dotado de sensores para identificar a presença de pessoas?
- Que tipo de sensores podem ser aplicados?
- Quão caro pode ser prover um ambiente dotado de sensores que determinem a localização de uma pessoa em um ambiente?

Vários tipos de sensores foram cogitados para a utilização no ambiente, como sensor de movimento, acelerômetros, *tags* NFC, etc. Contudo, a aplicação de sensores

¹ Uma paisagem sonora é composta pelos diferentes sons que compõem um determinado ambiente, sejam esses sons de origem natural, humana, industrial ou tecnológica.

dedicados no ambiente é onerosa e muitos destes sensores encontram-se difundidos em larga escala em dispositivos móveis, então a solução encontrada para este problema foi o uso dos sensores que os participantes em potencial já têm embutidos em seus *smartphones*.

Nesta etapa, várias soluções para o problema de composição de música colaborativa foram propostas e classificadas em dois segmentos: o primeiro, consistia na interação sem *feedback* de áudio e, o segundo, e mais explorado, consistia em soluções de interação sempre com a presença do *feedback* de áudio. Neste seguimento, foram projetadas: uma solução usando *bluetooth* como forma de determinar a presença do usuário, uma solução utilizando GPS como delimitador do ambiente e uma solução utilizando redes WIFI para a criação do ambiente virtual. Cada uma dessas soluções são discutidas a seguir.

No primeiro segmento de soluções, foi proposto que os participantes interagissem com o ambiente e com outros participantes, mas não recebessem *feedback* de áudio, somente teriam sua interação registrada e, em seguida, seria concebida uma composição com o resultado desta interação. Essa solução logo foi descartada e este segmento descontinuado, pois a falta de *feedback* de áudio não fez muito sentido, uma vez que não proporcionava uma interação com o som propriamente dita. Existia a colaboração e uma composição, mas a interação dos participantes estava limitada ao silêncio, o que não era interessante para o projeto.

No segundo segmento de soluções, o *feedback* de áudio se tornou pré-requisito e a primeira solução proposta foi utilizar o *bluetooth* dos *smartphones* para determinar se o usuário estava dentro do ambiente ou não. Neste caso, essa solução requeria um ponto de referência como núcleo central do ambiente e, ao se aproximar, deste ponto central, o som escolhido pelo usuário era reproduzido em seu smartphone como *feedback* na interação. Contudo, a tecnologia possui limitações e a principal limitação encontrada nessa tecnologia foi o tempo de varredura dos dispositivos ser muito alta, ou seja, um usuário poderia entrar e sair do ambiente em um curto espaço de tempo e este nem sequer ser notado, comprometendo o desempenho e confiabilidade do sistema.

Então, a segunda solução proposta seria usar o GPS como forma de determinar se o usuário estava ou não dentro do ambiente delimitado virtualmente. Ao se aproximar de um ponto central definido via software, assim como na proposta anterior, o som escolhido pelo usuário era reproduzido e sua interação e trajetória eram registrados para posteriormente ser construída a composição musical com os dados obtidos. Esta solução, apesar de interessante, não chegou a ser prototipada devido à dificuldade em determinar um espaço virtual através da geolocalização em locais pequenos.

Ao analisar a forma de *feedback* de áudio nas soluções anteriores, foi verificado que este poderia ser melhor aproveitado em um contexto geral, se fosse reproduzido em caixas de som espalhadas pelo ambiente, em vez de somente ser reproduzido no *smartphone*. Algumas situações que poderiam atrapalhar a interação foram identificadas, por exemplo: no momento da interação, um participante poderia não conseguir ouvir o som do

outro caso este esteja baixo demais, ou mesmo um som muito alto poderia atrapalhar a percepção dos demais.

Adotada a forma de *feedback* em caixas de som espalhadas pelo ambiente, surgiu a ideia de utilizar uma rede sem fio para determinar este espaço. Utilizando um roteador como ponto central, e com a potência de transmissão do sinal reduzido ao remover as antenas do dispositivo, foi possível delimitar um espaço virtual pequeno para o ambiente de interação. O tempo de varredura para encontrar novas redes nos *smartphones* é muito menor que no *bluetooth*, diminuindo então o tempo de resposta das ações de entrar e sair do espaço de interação. Uma aplicação servidora foi proposta para se comunicar com os *smartphones*, executar o som em alto-falantes posicionados no ambiente, registrar toda a interação dos participantes e gerar, ao final da composição, o resultado desta interação.

Nesse caso, a dinâmica de funcionamento da solução é a seguinte: ao enxergar o ID do roteador colocado como ponto central, o *smartphone* inicia a reprodução do som escolhido pelo usuário. Ao deixar de detectar a rede do roteador, o sistema deve entender que o usuário saiu do espaço de interação, encerrando, assim, a reprodução do som escolhido.

O projeto de interação sonora dessas soluções seguiu as etapas de DT, onde várias propostas de solução foram analisadas e as soluções mais viáveis seguiram para serem prototipadas. Na etapa de prototipagem, o processo de cada protótipo é avaliado, são observados erros e acertos da tecnologia proposta com o intuito de observar pontos fortes a serem explorados na solução.

3.3 Prototipação

O processo de desenvolvimento das soluções propostas resultou em três protótipos, a saber: (1) protótipo para testar, através do uso de simuladores, a ideia inicial da composição através da execução simultânea de sons de participantes que entram e saem em um ambiente; (2) protótipo para avaliação com usuários reais; (3) protótipo para considerar a localização espacial dos participantes no ambiente de interação e ajustes no protótipo para que possa ser testável com usuários em ambientes pequenos de interação e que inviabilizam o uso da geolocalização. Cada um dos protótipos são detalhados a seguir:

3.3.1 Primeiro Protótipo

A fim de avaliar a viabilidade da implementação de uma aplicação para um lado do servidor, foi construído um protótipo rudimentar para verificar se o ambiente de programação Pure Data poderia lidar com várias solicitações de rede, processar e reproduzir muitos sons diferentes, ao mesmo tempo e em tempo real.

Dado que esse estudo tinha um objetivo técnico e seus resultados nos guiariam a decisões tecnológicas sobre o projeto, um teste com participantes reais, neste momento,

não seria viável. Então, um protótipo foi construído utilizando o Pure Data foi testado em conjunto com a simulação de um espaço de interação utilizando o software, TUIO Simulator (TUIO, 2016), representado na Figura 18.

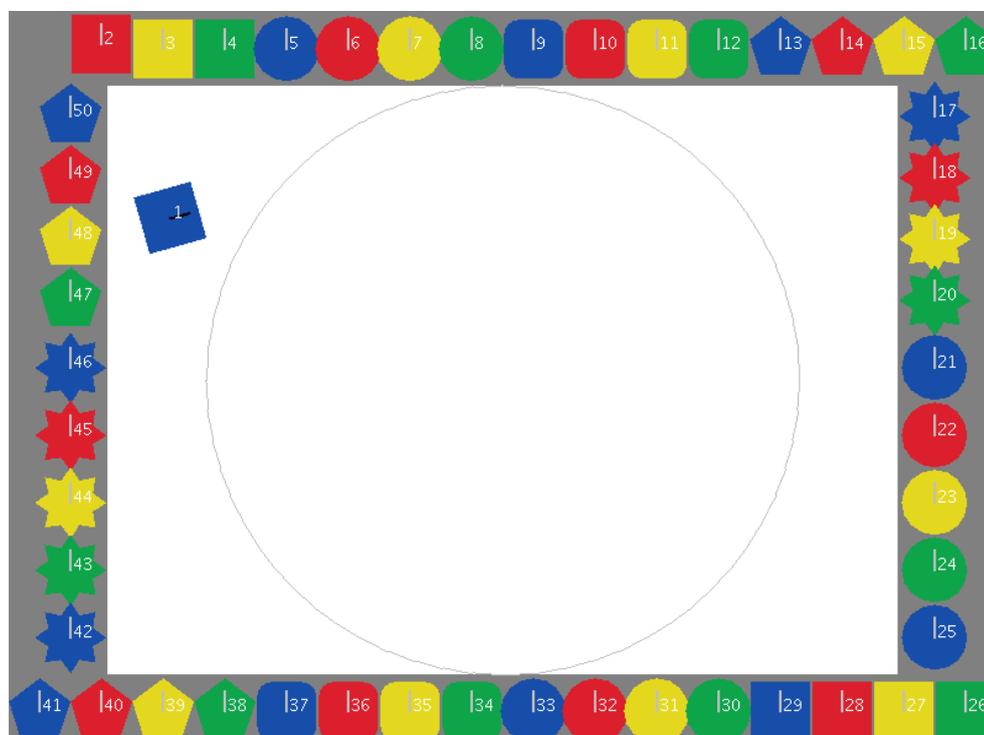


Figura 18 – Espaço de interação virtual no TUIO Simulator.

O simulador TUIO teve de ser adaptado para dar suporte a um maior número de figuras geométricas. Cada uma dessas figuras geométricas representa um usuário no espaço de interação, sendo que estas também podem ser giradas pelo traço em branco no centro. Essa rotação seleciona um dos vinte sons disponíveis.

O círculo no meio do simulador representa o espaço de interação. Quando uma figura é colocada dentro do círculo, uma mensagem é enviada para a aplicação do lado do servidor com Pure Data e o som indicado pelo ângulo da figura começa a ser reproduzido. Quando o usuário deixa o espaço de interação, outra mensagem é enviada para aplicação do lado servidor que encerra a reprodução do som. Essa comunicação, através de mensagens entre a aplicação servidora e o simulador, utiliza um protocolo de mensagens próprio do simulador.

3.3.1.1 Discussão

Com o resultado da avaliação do primeiro protótipo, verificou-se a viabilidade de adotar o Pure Data como plataforma para o ambiente. Foi possível reproduzir com sucesso, e sem sobrecarga, vários sons ao mesmo tempo. A plataforma foi robusta o suficiente para lidar com múltiplas solicitações de vários usuários.

Nesta primeira versão, todos os usuários foram modelados um por um, impondo um limite de 50 participantes. Caso fosse necessário usar a aplicação para um número maior de participantes, o procedimento seria incluir manualmente cada novo usuário. Se houvessem erros de implementação, seria quase impossível de encontrar e corrigi-los, pois a quantidade de elementos na tela era consideravelmente grande e complexa. Um maior número de elementos causa no Pure Data algumas dificuldades gráficas no manuseio de tantos objetos. A Figura 19 ilustra somente o módulo de reprodução e cada bloco de código representava um objeto de interação do ambiente.



Figura 19 – Versão de testes do primeiro protótipo em que cada usuário foi modelado manualmente um a um.

Outro ponto crítico deste protótipo foi o uso do protocolo de comunicação TUIO. Esse protocolo tem uma sintaxe própria que deve ser adotada para estabelecer a comunicação entre um dispositivo e o servidor. Quando utilizado com o simulador, o protocolo funcionou bem, mas quando utilizado para implementar a comunicação usando um dispositivo móvel, surgiram alguns problemas de suporte do protocolo. Como resultado, decidiu-se adotar o protocolo *Open Sound Control* (OSC), um protocolo melhor suportado pela comunidade.

3.3.2 Segundo Protótipo

Após a discussão acerca do primeiro protótipo, o segundo recebeu várias correções de problemas identificados na versão anterior, bem como melhorias de performance, mas ainda estava limitado a 50 participantes. Este protótipo da aplicação servidora pode ser conferido na Figura 20. A esta solução, foi integrada uma aplicação cliente desenvolvida para dispositivos móveis representada pela Figura 21. Um estudo empírico foi realizado durante o teste do protótipo com participantes reais um cenário real de uso, somente identificando os participantes que estavam dentro ou fora do espaço de interação.



Figura 20 – Segundo protótipo da aplicação servidora com melhorias de performance e código encapsulado.

A aplicação do lado cliente foi desenvolvida por um aluno de iniciação científica ligado ao projeto, consistindo em um aplicativo móvel desenvolvido na plataforma Android e chamado de Compomus (**COMPO**sição **MUS**ical). Essa plataforma foi adotada devido ao seu baixo custo para o desenvolvimento e implantação. O Android é também

uma das plataformas mais populares entre usuários móveis. Compomus requer a versão 6.0 do sistema operacional Android, devido a problemas com as permissões das versões anteriores. O *App* funciona em conjunto com um servidor web com banco de dados para armazenamento de informações dos participantes e seus sons escolhidos.

Atualmente, o aplicativo está disponível apenas em português-brasileiro (PT-BR). A interface do aplicativo, descrita na Figura 21, será explicada em detalhes a seguir.

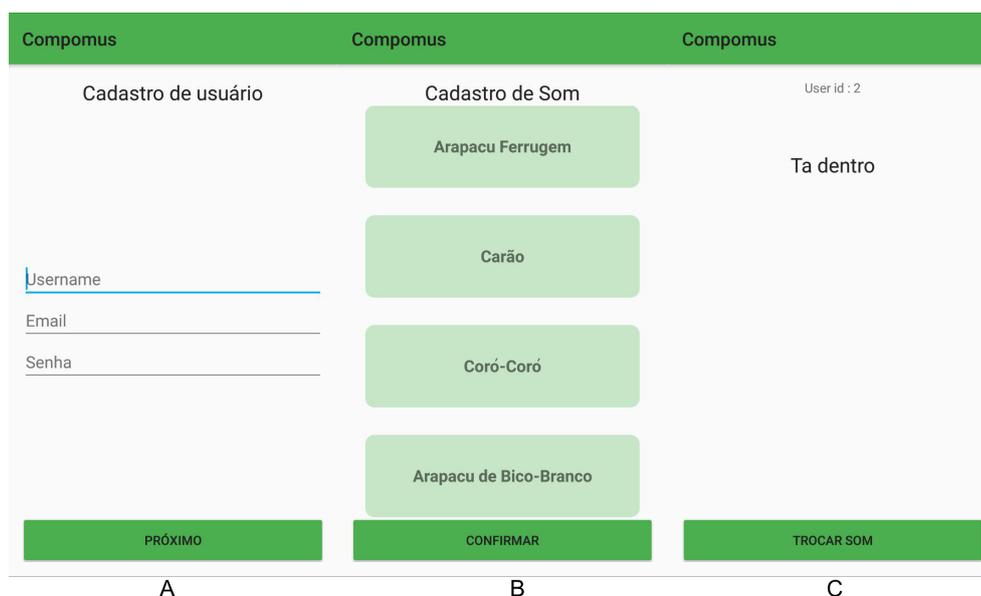


Figura 21 – Primeira versão funcional do Compomus.

O projeto do Compomus consiste em três telas principais. A tela (A) descreve o formulário de registro que um novo usuário deve preencher. Os campos obrigatórios desse formulário são: nome de usuário, e-mail e uma senha. Uma vez registrado, o usuário é conduzido para a Tela (B). É apresentada a tela (B), que consiste em uma lista de sons onde o usuário deve escolher aquele que irá representá-lo no espaço de interação. O usuário pode ouvir uma pré-visualização dos sons disponíveis e escolher qual deles mais gosta. Nesse contexto, os sons disponíveis são cantos de pássaros da floresta amazônica: Arapacu Ferrugem, Carão, Coró-Coró e Arapacu de Bico-Branco são nomes de aves típicas da fauna amazônica. Essas amostras foram utilizados com o intuito de recriar a cena sonora de uma floresta, assim disponibilizando um total de vinte sons no sistema.

Uma vez escolhido o som, é fornecida uma tela (C) ao usuário. Nessa tela, o *App* informa o número de identificação, status que indica se o usuário está dentro ou fora do espaço de interação e um botão para troca de som que pode ser ativado a qualquer momento. Se quiser alterar o som escolhido, basta pressionar o botão disponível e selecionar um novo, e após selecionar um novo som o usuário é levado de volta à tela (C).

Neste segundo protótipo, o funcionamento do ambiente, que pode ser acompanhado na Figura 22, dá-se da seguinte maneira: o smartphone do usuário deve estar conectado ao roteador (A) para permitir a comunicação do Compomus com o aplicativo do lado

1. Questões:

- Os participantes perceberão sua interação no espaço de interação?
- Os participantes perceberão o *feedback* de áudio no alto-falante do ambiente?
- Os participantes perceberão a composição colaborativa em tempo real?

2. Contexto e Assuntos: O estudo foi realizado na universidade com dez alunos matriculados em uma disciplina denominada Sistemas Colaborativos, oferecida aos alunos do curso de Ciência da Computação e Sistemas de Informação. Os sujeitos já estavam familiarizados com os principais conceitos do projeto, mas não estavam envolvidos em seu desenvolvimento. Um ponto a ser destacado é que os alunos participantes do estudo não tinham nenhum conhecimento avançado sobre composição, produção ou teoria musical.

Considerando o espaço de interação para este contexto, definimos uma pequena área em uma sala de aula onde a disciplina ocorre. Para limitar esse espaço, o intervalo do roteador que desempenha o papel de ponto central tinha de ser limitado. Foi disponibilizado somente um alto-falante amplificado no meio do espaço de interação para o *feedback*.

3. Definição das tarefas: Os sujeitos deste estudo tiveram que realizar seis tarefas principais: (A) Instalar o Compomus em seus dispositivos móveis Android; (B) Preencher o formulário de inscrição e registrar a candidatura; (C) Selecionar um som para representá-los no espaço de interação; (D) Interagir livremente entrando e saindo do espaço de interação; (E) Alterar o som; (F) Completar um questionário após o estudo.

3.3.2.3 Execução

Aos participantes foi solicitado que baixassem e instalassem o Compomus em seus dispositivos móveis, fizessem o registro no aplicativo e escolhessem um som para se identificarem. Em um primeiro momento, atribuímos um som diferente a cada usuário para poder identificá-los facilmente no alto-falante. Foi aconselhado que eles colocassem seus dispositivos em modo silencioso, a fim de facilitar a percepção de seu próprio som no alto-falante no meio do espaço de interação. Várias fontes de som podem prejudicar a percepção dos participantes.

Os participantes foram capazes de interagir livremente uns com os outros e com o sistema, entrando ou saindo do espaço de interação. Os participantes também foram capazes de mudar o som que lhes foi atribuído para o som que mais se identificaram, com o intuito de verificar se poderiam perceber a mudança na composição musical resultante reproduzida no alto-falante. Eles puderam interagir com o ambiente durante 15 minutos.

No final do processo, foram convidados a preencher um questionário com perguntas sobre sua experiência durante o estudo.

3.3.2.4 Discussão

A execução deste estudo nos permitiu observar a colaboração na tarefa de composição de música em tempo real. Esta colaboração foi analisada de acordo com o Modelo de Colaboração 3C (FUKS et al., 2007). A ideia principal deste modelo é que a colaboração seja alcançada com a interação de três dimensões diferentes: (C)omunicação, (C)oordenação e (C)ooperação: Foi possível observar cada “C” do modelo durante a interação dos sujeitos no espaço de interação.

(C)omunicação: No contexto deste estudo, a comunicação ocorreu de forma não verbal. Os sujeitos estavam se comunicando uns com os outros através de seus sons no ambiente. Quando um novo som era tocado no alto-falante, isso significava que um novo sujeito entrava no espaço de interação. Quando um som parava no alto-falante, significava que um sujeito deixara o espaço de interação.

(C)oordenação: No contexto deste projeto, o objetivo da colaboração é compor uma música em tempo real, mas a coordenação acontece espontaneamente com base no *feedback* de áudio dado pelo alto-falante. Não há um “diretor de arte” ou uma figura central orquestrando a interação para produzir a música resultante.

(C)ooperação: Neste estudo, o espaço compartilhado é o espaço de interação, onde os sujeitos devem entrar, e sair e mudar os sons se quiserem. A operação conjunta é feita individualmente, interagindo no espaço de interação. O resultado dessas interações individuais múltiplas é gravado no servidor e reproduzido em tempo real no alto-falante.

As três questões que orientaram este estudo tiveram a ver com a percepção da interação, o *feedback* de áudio e a tarefa colaborativa durante o desempenho. Analisando as repostas do questionário (disponível no Apêndice A) aplicado após o teste, notamos que todos os sujeitos foram unânimes em dizer quão importante eles foram na composição global durante a interação no ambiente. Isso leva a concluir que eles foram capazes de perceber todos os três pontos já mencionados.

Embora eles tenham relatado uma boa experiência usando Compomus e que claramente se sentiram parte de uma composição de música colaborativa, os participantes levantaram pontos importantes, como os problemas que enfrentaram durante a experiência. Um exemplo é a falta de existência de *feedback* visual na tela de status, em vez de apenas texto indicativo. Os participantes também relataram a ineficiência da aplicação em alguns casos, por exemplo: aplicativo mostrou na tela de status que o dispositivo estava dentro do espaço de interação, mas o som não foi reproduzido.

Algumas das respostas de alguns usuários podem ser verificadas através das citações: “Aplicação cumpre o objetivo que foi repassado, fácil de ser utilizada, apesar de existir problemas com a localização do seu usuário”; “A aplicação é boa e divertida, mas

tem alguns problemas com a localização. Por exemplo, às vezes a pessoa está fora do intervalo em que o som deve reproduzido, mas do nada o dispositivo acha que a pessoa ainda está dentro do intervalo”; “Com relação ao quesito (8): Embora o *feedback* seja suficiente, tem como ser melhorado com efeitos visuais (além de um texto)”.

Há um atraso de dez segundos na busca de novas redes do Sistema Operacional Android. O problema também ocorreu quando um sujeito deixou o espaço de interação e seu som continuou a tocar por algum tempo, até que o sistema perceba que o usuário estava fora de alcance.

Um fato importante que podemos notar foi que a composição musical resultante será sempre diferente, mesmo que as configurações de cenário mantenham o mesmo. Isso ocorre devido à natureza colaborativa do ambiente que é totalmente dependentes de interações humanas. Essas interações nunca serão as mesmas. Outro ponto observado é que, apesar de os participantes terem relatado no geral uma boa experiência e interação, essa interação ficou limitada somente à entrada e saída dos participantes, sendo uma interação pobre e que necessita de uma reformulação.

3.3.3 Terceiro Protótipo

Mais uma vez realizados os estudos a partir das versões anteriores, correções apontadas pelos participantes foram implementadas. Para lidar com os problemas relatados, aumentar o desempenho e adicionar escalabilidade ao sistema, o aplicativo do lado servidor foi completamente reestruturado. Utilizou-se técnicas para gerar código executável, *subpatches* e *patches* dinâmicos, o que reduziu drasticamente a quantidade de elementos gráficos na tela como mostrado na Figura 23.

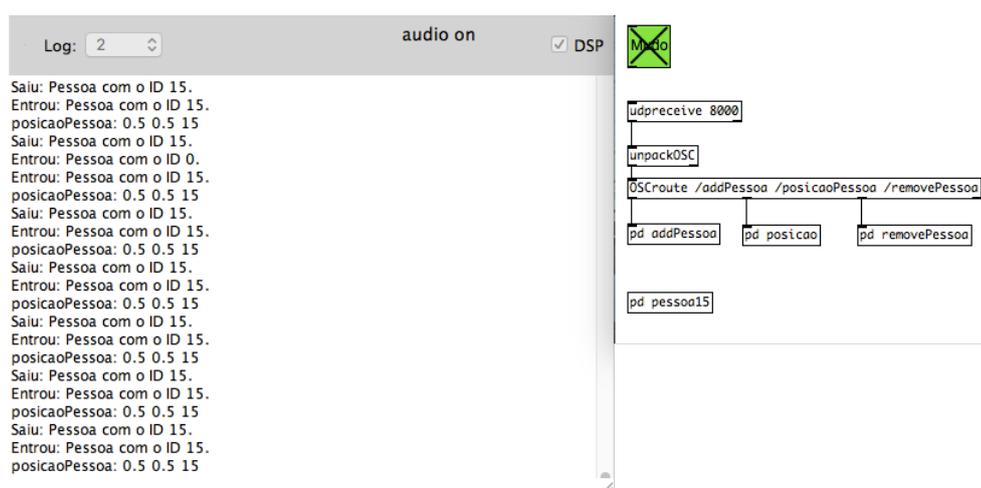


Figura 23 – Terceiro protótipo da aplicação do lado servidor.

Nesta versão da aplicação servidora, além das melhorias citadas acima, foi adicionado o suporte de espacialização sonora em 4 canais, utilizando uma técnica de panorama

cruzado ou quadrafônica. Como apresentado na Subseção 2.1.2.2 do Capítulo de Fundamentação Teórica, a técnica de panorama por amplitude proporciona uma fonte sonora “Fantasma” ao projetar uma fonte sonora virtual entre dois alto-falantes. Nessa aplicação, para simular que o som está mais para um lado, o alto-falante oposto tem o volume de reprodução reduzido, trazendo assim a sensação de movimento.

Na técnica aplicada nesta solução, o som monofônico passa por dois filtros de ganho (panorama), onde cada um possui uma entrada de sinal sonoro monofônico, uma entrada para a localização da posição da fonte sonora entre os dois alto-falantes e duas saídas de sinal de áudio. As informações da coordenada X são enviadas para um filtro e as informações da coordenada Y são enviadas para o outro filtro, juntamente com uma cópia do sinal monofônico para cada um. Então, o sinal sonoro é tratado nesses filtros e enviado para o canal de saída de áudio correspondente de acordo com a localização recebida.

No Compomus, foi implementada a detecção da localização do usuário no espaço de interação que, nesta versão, dava-se através do módulo de GPS do smartphone do usuário, os dados são enviados para o servidor de áudio que interpreta as informações e faz a espacialização em tempo real como citado acima.

Além das questões do estudo anterior, uma nova foi adicionada às questões a serem respondidas:

- Os participantes perceberão o seu som sendo espacializado de acordo com a sua localização no ambiente?

A execução deste estudo se deu da mesma maneira do anterior, participaram deste teste os mesmos dez alunos participantes do teste anterior. Contudo, o local de testes foi alterado para um dos espaços de convivência da universidade, devido à necessidade de usar mais caixas de som e precisar de mais espaço. O tamanho do espaço de interação foi delimitado unicamente por questões técnicas, isto é, a quantidade de cabos disponíveis para conexão dos equipamentos de som não permitiu que o espaço de interação fosse maior. Os participantes ficaram livres para poder interagir com o ambiente durante o período de 15 minutos, como mostra a Figura 24.

3.3.3.1 Discussão

A execução deste teste teve significativa importância para o projeto, pois confirmou as especulações a cerca da imprecisão do GPS nos smartphones. A performance do sistema no local utilizado para o teste não foi muito boa, de certa forma, devido ao local ter pouco espaço e possuir alguns pavimentos acima, dificultando a localização do GPS.

Alguns participantes relataram ter conseguido mesmo, que por pouco tempo, espacializar o som com a sua localização de forma correta. Contudo, vários problemas acerca da precisão foram relatados pelos participantes, problemas como saltos de localização, erros de localização, exemplo: um usuário estava em um ponto e o GPS indicava que o



Figura 24 – Teste de espacialização quadrafônica com participantes.

ele estava no lado oposto. Com base nos relatos dos participantes verificou-se que seria necessário um espaço maior e descoberto para uma melhor performance da localização via GPS.

Alguns dos problemas relatados pelos participantes neste protótipo está relacionado à imprecisão do GPS que, apesar dos problemas com a localização, relataram uma boa experiência quanto a espacialização e acharam que tiveram uma grande imersão no ambiente devido ao maior número de caixas e o som ser direcionado de acordo com a sua localização, mesmo que imprecisa.

A experiência de interagir com o ambiente segundo os participantes não foi muito prejudicada e foi possível perceber com facilidade a espacialização acontecer, e que também eram responsáveis pelo movimento do som no ambiente ao se posicionarem em outros locais. Os participantes relataram que a interação com o ambiente era mais interessante que a proporcionada no teste do segundo protótipo, onde só era possível entrar e sair do espaço de interação.

3.3.4 Ajustes de Protótipo

Ainda no terceiro protótipo, foram consideradas todas as melhorias propostas pelos participantes nos testes anteriores e foram executadas em sua maioria na aplicação do lado servidor, incluindo melhorias de performance e a substituição da técnica de auralização quadrafônica pela técnica *Ambisonics* (LECOMTE; GAUTHIER, 2015), que proporciona

uma flexibilidade do número de alto-falantes e precisão na virtualização da fonte sonora com o uso da biblioteca de código aberto *OpenAudience*.

O Compomus, neste momento, recebeu uma versão de depuração adicionada com o único intuito de testar a nova técnica de auralização e aperfeiçoar a espacialização sonora, e uma nova tela “C” foi adicionada, como mostra a Figura 25. A tela “C” agora possui um “*Joystick*” onde o usuário pode projetar o som para o ponto que quiser dentro do espaço de interação. Devido ao pouco espaço no local de testes, foram adicionados dois botões onde o usuário poderia se colocar dentro ou fora do ambiente delimitado.



Figura 25 – Tela C do Compomus para o quarto protótipo.

Nesse teste, o estudo foi realizado no Centro de Artes da Universidade, com dois alunos de doutorado e dois professores voluntários do IComp; Figura 26. Os sujeitos já estavam familiarizados com os principais conceitos do projeto, mas não envolvidos com seu desenvolvimento. Outra questão sobre os temas é que assim como os outros voluntários, eles não tinham nenhum conhecimento avançado sobre composição, produção ou teoria musical.

Considerando o espaço de interação, foi definido uma pequena área no estúdio de ensaios. Nesse caso, o Roteador (A), que definiria o espaço de interação, não foi necessário, pois, os participantes teriam a possibilidade de entrar ou sair do ambiente como foi



Figura 26 – Avaliação do quarto protótipo.

detalhado nas modificações do projeto do Compomus acima. A definição de tarefas segue o modelo utilizado no segundo protótipo que, ao final dos testes, os participantes devem preencher um questionário sobre a sua experiência com o Compomus.

3.3.4.1 Discussão

Como mencionado anteriormente, o foco deste protótipo é a espacialização utilizando a técnica de auralização *Ambisonics* e a espacialização do som utilizando a biblioteca *OpenAudience*.

Após o uso do Compomus, os participantes responderam sete questões acerca de sua experiência com o sistema. Um dos participantes relatou que sentiu alguma dificuldade no início da utilização do Compomus e que o aplicativo chegou a travar algumas vezes. O restante afirmou ter tido uma boa experiência com o Compomus e o sistema em geral.

Quando indagados se tinham percebido a mistura dos sons e se ainda era possível identificar o seu som, um usuário relatou que tinha notado que ao posicionar seu som exatamente em uma caixa de som e outro usuário também o fizesse, o som escolhido parava de ser reproduzido e vice-versa. Os outros participantes afirmaram que sim, foi possível perceber a mistura dos sons e identificar o som escolhido dentre os outros.

Todos os participantes afirmaram que conseguiram identificar a origem do som corretamente e com boa precisão. Todos os participantes também afirmaram que foi possível sim interagir com outros participantes para tentar adivinhar o som escolhido pelo outro ou mesmo para tentar perceber de onde o som estava sendo reproduzido.

A mistura dos sons, segundo os participantes, não afetou a percepção do seu som e dos sons escolhidos pelos outros. Apesar de alguns problemas relacionados à espacialização, relataram uma boa experiência com o ambiente, e que acharam interessante a forma

como foi aplicada a tecnologia na arte de compor música com sons aleatórios.

3.4 Implementação da Melhor Solução

A melhor solução, de acordo com a análise dos protótipos descritos na seção anterior, foi o terceiro protótipo do ambiente de interação após as últimas correções, a solução possui *feedback* de áudio espacializado em tempo real, utiliza o GPS como forma de determinar a localização do usuário combinado com uma rede *WIFI* para delimitar o espaço de interação. A aplicação servidora, integrante dessa solução, ainda sofreu modificações, tornando-se um *Framework* para espacializar sons em tempo real sendo aplicável a múltiplos contextos.

Essa implementação foi testada e avaliada em três contextos diferentes: em conjunto com uma aplicação que visava encontrar o ponto de saturação do som, em um ambiente com participantes reais na Universidade Federal do Amazonas Campus de Itacoatiara, com o intuito de espacializar o som em tempo real e avaliar o desempenho da espacialização do *framework* e, por último, foi testado em conjunto com um jogo. Os detalhes desta avaliação podem ser conferidos no Capítulo 5, Avaliação do *Framework*.

3.5 Conclusão

Este capítulo apresentou o desenvolvimento do processo de *Design Thinking* como abordagem para resolução do problema de espacialização sonora tratado por este trabalho. Esse processo possui quatro etapas:

A primeira etapa consiste em conhecer o problema abordado, em que quatro reuniões com os participantes do projeto foram realizadas para a delimitação do problema e apresentação de conceitos e tendências na área da computação e música.

A segunda etapa consiste em projetar soluções, onde é organizada uma tempestade de ideias, as quais devem propor soluções candidatas para o problema e as soluções mais viáveis são prototipadas.

Na terceira etapa do processo do DT, acontece a prototipação e é onde as soluções candidatas são testadas. Durante essa etapa, algumas tecnologias foram utilizadas e outras foram abandonadas. Dentre as tecnologias utilizadas e posteriormente foram abandonadas, estão:

- O protocolo de comunicação TUIO;
- O TUIO Simulator;
- A técnica de espacialização sonora quadrafônica;
- O bluetooth e GPS como delimitadores do espaço de interação.

Dentre as abordagens e tecnologias adotadas estão:

- Pure Data como ambiente e linguagem de programação;
- O Open Sound Control (OSC) como protocolo de comunicação;
- O Compomus como aplicação cliente;
- O uso combinado de GPS e Wi-Fi para a determinação da localização do utilizador do Compomus;
- As camadas três e quatro da biblioteca *OpenAudience* para o módulo de espacialização sonora contido na aplicação servidora, trazendo o suporte aos modos de espacialização *Ambisonics*;
- A técnica de espacialização *Ambisonics*;
- O paradigma de *patching* dinâmico trazendo escalabilidade para aplicação servidora;
- A utilização de objetos e abstrações permitindo maior performance para a aplicação servidora.

A última etapa do processo de DT é a implementação da melhor solução, esta inclui todas as correções e ressalvas apontadas pelos utilizadores durante os testes. Nesse caso, a melhor solução encontrada possui todos os itens destacados anteriormente como abordagens e tecnologias adotadas.

Durante o desdobramento do processo descrito neste capítulo, notou-se um potencial da aplicação servidora para ser utilizada não somente com o Compomus, mas também com outras aplicações cliente. Sendo assim, decidiu-se generalizar o código da aplicação servidora para que esta se tornasse um *framework* para a espacialização sonora em tempo real, utilizável em múltiplos contextos. Foram projetadas uma arquitetura e novas funcionalidades para a aplicação, que podem ser conferidas no capítulo seguinte.

4 ARSUS *Framework*

Como resultado da avaliação dos protótipos concebidos como candidatos à resolução do problema de espacialização sonora utilizando tecnologias emergentes descrito na Seção 3.3, chegou-se à conclusão de que a melhor versão resultante das etapas de *Design Thinking* tinha um potencial para ser utilizado em múltiplos contextos.

Na etapa de levantamento de trabalhos na literatura, verificou-se que o conceito de espacialização sonora não era aplicado somente no âmbito artístico. Muitos trabalhos envolvendo a espacialização sonora eram voltados para a realidade virtual, assistência para a navegação de pessoas com deficiência visual, jogos, ambientes de treinamentos, performances artísticas bem como composição de música colaborativa, eletroacústica e outros.

Durante a avaliação do protótipo 3 e seu refinamento, percebeu-se que a aplicação servidora poderia ser generalizada como um *framework* e que este poderia dar suporte computacional não somente ao contexto do problema inicial, mas poderia também ser aplicado aos contextos acima citados, se este fosse alimentado com as entradas requeridas.

Sendo assim, a aplicação servidora desenvolvida inicialmente para espacializar somente os sons do Compomus recebeu melhorias e generalizações em seu código para se tornar uma solução independente denominada como ARSUS *Framework* (*scALable Real-time SoUnd Spatialization*), uma tecnologia que dá suporte computacional para espacialização sons em tempo real, sem a necessidade prévia de conhecimento de programação ou das teorias musicais que estão por trás do conceito de som espacial.

Apesar da aplicação servidora ter se tornado o ARSUS Framework, essa não deixou de dar suporte para o Compomus, pois ambos se tornaram softwares distintos e não um sistema como um todo.

Devido a grande parte da implementação do ARSUS ter sido concebida enquanto este fazia parte do sistema de espacialização proposto como solução do problema de espacialização sonora, não se faz necessário que seja repetido neste capítulo, portanto, nas seções seguintes serão detalhados somente os aspectos ainda não apresentados durante este trabalho.

4.1 Requisitos para Espacialização Sonora

Na engenharia de Software, a análise de requisitos é a primeira atividade técnica no desenvolvimento do software, e pode ser entendida como responsável por definir os serviços que um sistema deve realizar, sua interface com os demais elementos e sob quais restrições o sistema deve operar (GIOVANELLA, 2013).

Para a criação de um ambiente de som espacializado é necessário que se cumpram alguns requisitos:

- Caixas de som - Para promover a espacialização sonora é necessário que se tenha mais de uma caixa de som. Conforme dito na Subseção 2.1.2, a espacialização sonora pode ser realizada em diversas modalidades como: estéreo, binaural, quadrafônico, hexagonal, octogonal e octogonal cúbico. Todas essas modalidades necessitam de um certo número de caixas específico para serem utilizadas de acordo o método de escolhido. Há dois tipos de caixa de som, as acústicas que necessitam de um amplificador de som e amplificadas que já possuem amplificação sonora integrada.
- Amplificador de som - É um equipamento eletrônico que a partir de um pequeno sinal de áudio colocado na sua entrada, fornece na sua saída este sinal ampliado e adequado ao funcionamento de uma ou mais caixas de som acústicas (sem amplificação).¹ Como dito anteriormente, algumas caixas de som já possuem esse aparelho integrado e são conhecidas como caixas de som amplificadas. Outro tipo de equipamento que já possui esse aparelho em seu interior e que foi desenvolvido para a reprodução de áudio multicanais é o *Home Theater*.
- Dispositivos de Áudio Multicanal - São dispositivos capazes de reproduzir áudio em mais de dois canais de forma interdependente. O objetivo deste tipo de dispositivo, é envolver o ouvinte com caixas de som e simular a ambiência da gravação. Existem alguns tipos de dispositivos multicanais como placas de som, interfaces de áudio e amplificadores multicanais. Esses dispositivos geralmente são utilizados em cinemas, estúdios de gravação e integrados em *Home Theaters*.
- Softwares de Mixagem - O som espacializado requer uma estrutura computacional constituída de hardware e software, pois a espacialização sonora multicanais exige uma manipulação do som que somente os softwares de mixagem podem proporcionar. Os softwares de mixagem são projetados para manipulação avançada do som, e são bastante utilizados na produção de música, filmes e dublagens e outros segmentos da música.

Há atualmente vários tipos de softwares disponíveis no mercado que podem proporcionar o som espacializado como: ProTools², Audacity³, GarageBand⁴, Logic Pro X⁵, Sound Forge⁶. Além de ferramentas de manipulação do som como Pure Data e MAX, os quais podem manipular o som a nível de sinal sonoro.

¹ Definição extraída de: <<http://www.studior.com.br/beaba.htm>>

² Disponível em: <<http://www.avid.com/pro-tools>>

³ Disponível em: <<http://www.audacityteam.org/download/>>

⁴ Disponível em: <<https://www.apple.com/br/mac/garageband/>>

⁵ Disponível em: <<https://www.apple.com/br/logic-pro/>>

⁶ Disponível em: <<http://www.magix-audio.com/us/>>

Os requisitos mencionados acima são necessários para promover a espacialização sonora para filmes, música, jogos e shows de música. Em alguns desses casos, é possível proporcionar uma espacialização sonora em tempo real através de manipulação de ganhos ou uso aplicativos que são construídos para tal. Contudo, em todos esses casos há uma figura centralizadora dessa espacialização, o engenheiro de som é quem decide qual o comportamento que o som deve ter, quais trajetórias deve seguir, o que normalmente é determinado a partir de um roteiro.

As tecnologias existentes não proporcionavam uma solução para o problema identificado por este trabalho, por isso a necessidade de se criar um framework que espacializasse vários sons em tempo real, onde cada som pudesse ser espacializado de forma independente promovendo assim uma descentralização. Sendo assim, para possibilitar a exploração do espaço como elemento composicional e proporcionar uma imersão sonora sem a necessidade de uma figura centralizadora, foi projetada a arquitetura do ARSUS que será detalhada a seguir.

4.2 Arquitetura

A fim de prover um suporte computacional completo para a espacialização sonora e sonificação de forma ágil e descomplicada, o ARSUS passou por modificações em seu código para que a sua utilização se tornasse a mais simples possível. Para que isto fosse possível, fez-se necessária a generalização da sua arquitetura representada na Figura 27, que possui três módulos principais e seus funcionamentos são detalhados a seguir:

4.2.1 Módulo de Comunicação

Este é o módulo responsável por receber as mensagens de entrada do *framework* e encaminhá-las para os outros módulos existentes. Este módulo espera como entrada cinco tipos de mensagens que devem estar no formato do protocolo OSC.

O *Open Sound Control* (OSC) é um protocolo aberto, de transporte independente, baseado em mensagens, desenvolvido para comunicação entre computadores, sintetizadores de som e outros dispositivos multimídia. Sua sintaxe é composta de um endereço, ou destino da mensagem a ser enviada e os parâmetros desejados que podem ser do tipo inteiro, ponto flutuante e caractere, mais detalhes podem ser conferidos em (OSC, 2017).

A primeira entrada esperada pelo *framework* deve conter o “endereço” de entrega da mensagem e dois números do tipo inteiro e separados por um espaço, exemplo: “/addObjeto 20 2”, onde “addObjeto” é o tipo de mensagem a ser enviada e representa o destino final da mensagem, que é encaminhada para outro módulo e significa que um novo *subpatch* será criado; “20” seria o ID do som a ser reproduzido e “2” é o ID do *subpatch*, que a partir deste momento será chamado somente de objeto de interação, pois é este quem realiza a interação do som no campo sonoro.

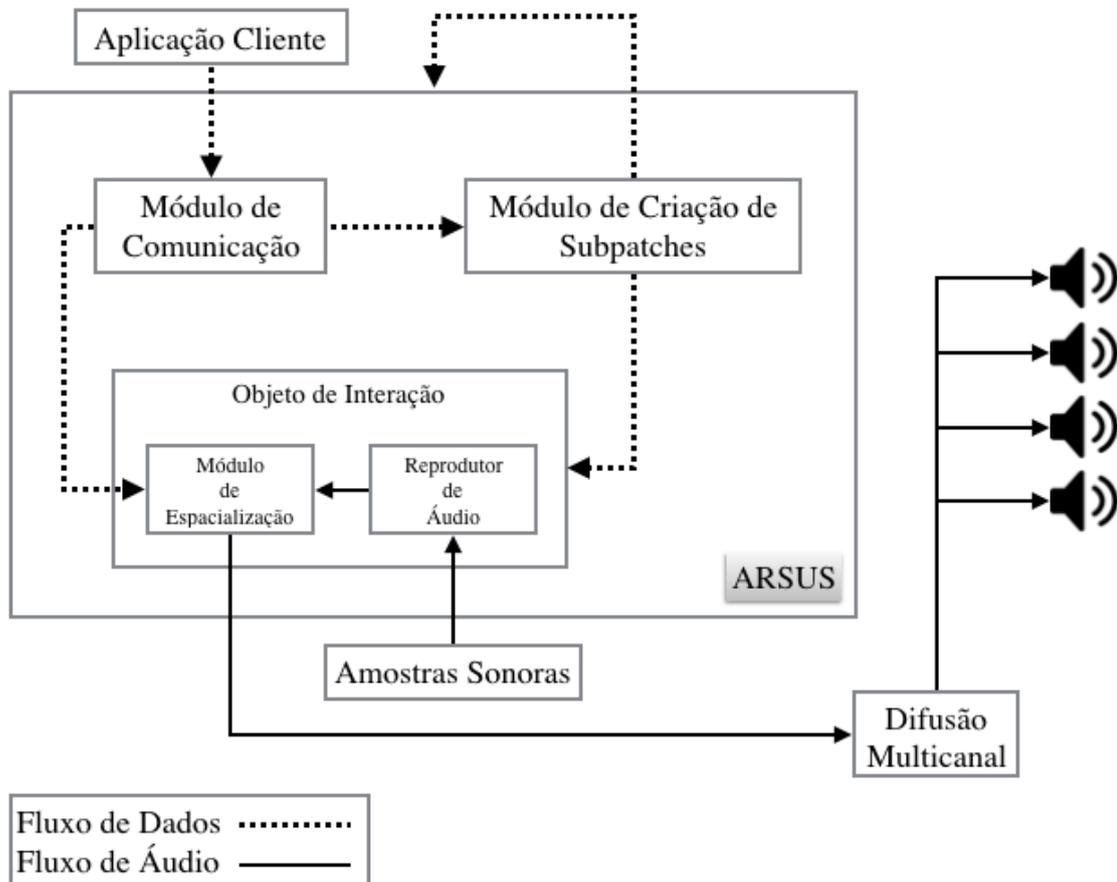


Figura 27 – Arquitetura proposta para o *Framework*.

O segundo tipo de entrada pelo módulo é o comando para reproduzir o áudio contido no objeto de interação. O formato da mensagem a ser recebida é: `“/playSound 2”`, em que `“playSound”` é o endereço de mensagem a ser enviada e `“2”` é a ID do objeto de interação.

O terceiro tipo de entrada esperada pelo módulo é um comando não obrigatório para iniciar a reprodução de um dado áudio em *loop* ou não. O formato da mensagem esperada é: `“/loopSound 0 2”`, onde `“loopSound”` é o endereço da mensagem OSC. Os argumentos esperados são `“0”` ou `“1”`, e são os comandos para ligar ou desligar a reprodução em *loop* do áudio, sendo `“0”` para desligar e `“1”` para ligar. E por, último o argumento `“2”` é o ID do objeto de interação que receberá essa mensagem.

O quarto tipo de entrada que esse modulo processa é o comando para terminar a execução do som, que é constituído somente do ID do objeto de interação, que deve ser do tipo inteiro, exemplo: `“/removeObjeto 2”`. Assim como nas entradas anteriores, `“removeObjeto”` é o tipo de mensagem a ser enviada para o *framework*, representando o destino da mensagem que é direcionada para o seu objeto correspondente e significa que o objeto de interação, criado anteriormente, agora será destruído.

O quinto tipo de entrada esperada pelo módulo são os dados de localização da fonte sonora em um plano cartesiano, que devem ser atualizados sempre que esta mudar.

O formato da mensagem a ser recebida pelo módulo é: “/posicaoObjeto 5 -3 2”, em que, novamente, “posicaoObjeto” representa o “endereço” final da mensagem a ser encaminhada para outro módulo diferente da anterior e, ao ser recebida, significa que a fonte sonora se moveu pelo plano; “5” representa a coordenada X, “-3” a coordenada Y e “2” seria sua ID.

Todas as mensagens recebidas do tipo “posicaoObjeto” são encaminhadas para os seus respectivos objetos de interação através de seu ID. Nesse caso, os números das coordenadas podem ser todos do tipo inteiro ou ponto flutuante para as coordenadas X e Y e inteiro para ID do objeto de interação, os dois tipos são suportados pelo *framework*.

4.2.2 Módulo de Criação de *Subpatches*

Este módulo de criação de *subpatches* é o que faz o ARSUS ser escalável, a técnica aplicada em sua construção chamada de *patching* dinâmico permite que *subpatches* com código personalizado sejam gerados. A arquitetura do ARSUS, Figura 27, ilustra seu funcionamento que se dá da seguinte maneira: ao receber a mensagem encaminhada pelo módulo de comunicação, o módulo de criação de *subpatches* inicia um processo de envio de várias mensagens. Essas mensagens são mensagens internas, e é o modo de comunicação entre os blocos de código, uma característica da própria linguagem.

Primeiramente, o módulo envia uma mensagem para o próprio framework com os comandos para criar um novo objeto de interação ou *subpatch* nomeado com ID recebido. Em seguida, envia uma sequência de mensagens ao novo objeto com comandos para criar um bloco de código com uma função para autodestruição do objeto e um reproduutor de áudio. Esse reproduutor tem sua saída de áudio conectada ao módulo de espacialização que, por sua vez, está conectado à saída de som multicanal, ou seja, todo objeto de interação criado possui dentro de si um módulo de espacialização de som, permitindo que cada som seja espacializado individualmente. Mais detalhes sobre o funcionamento do objeto de interação serão dados na subseção seguinte.

4.2.3 Objeto de Interação e Módulo de Espacialização

Com base nos conceitos de sonificação estudados em Projeto de Interação Sônica, o objeto de interação é a representação computacional de uma fonte sonora, isto é, assim como no mundo real, onde o som passa a existir durante o tempo em que é emitido por sua fonte sonora, se move ou não pelo campo sonoro e em seguida desaparece ao deixar de ser emitido, o objeto de interação foi modelado de forma que se torne o próprio som que lhe foi atribuído; existindo enquanto ele existir e desaparecendo quando ele desaparecer.

Ao ser criado, o objeto de interação recebe o reproduutor de áudio que contém, em seu código, o ID do som pré definido e o reproduz a partir do diretório com as amostras sonoras, que devem estar preferencialmente no formato *Audio Interchange File Format*

(AIFF) e renomeadas com números inteiros, que não necessariamente devem estar em ordem crescente, exemplo: o primeiro som 0.aiff, 1.aiff, 2.aiff ... x.aiff.

O formato padrão utilizado pelo ARSUS é o “AIFF”, por não possuir compressão e pela possibilidade de ser manipulado pelo protocolo OSC. Contudo, esse formato pode ser substituído a qualquer momento, bastando renomear o formato no módulo de criação de *subpatches*, procedimento sujeito à compatibilidade do Pure Data, podendo ser necessária a instalação de alguma biblioteca adicional.

Em seguida, o reprodutor de áudio embutido no objeto de interação envia o fluxo de áudio para o módulo de espacialização sonora, que é construído de acordo com a configuração de alto-falantes a ser utilizada. Este módulo faz uso das camadas três e quatro da biblioteca *OpenAudience*, que utiliza a técnica *Ambisonics* para a espacialização sonora e permite o uso de até oito alto-falantes para a criação de um ambiente espacializado de primeira ordem. O ARSUS permite a escolha da quantidade de alto-falantes a serem utilizados de acordo com as disposições suportadas pela biblioteca *OpenAudience*, Figura 28, e podem ser modificadas no módulo de criação de *subpatches*, as configurações disponíveis são: estéreo, quadrafônica, *surround*, hexagonal, octogonal e octogonal cúbica.

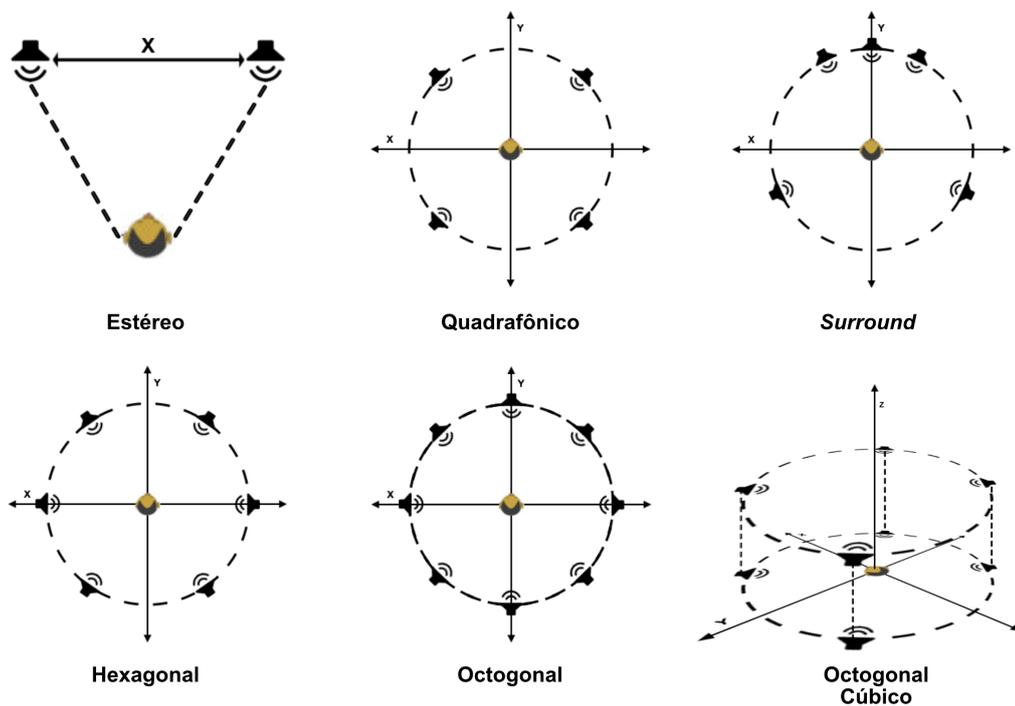


Figura 28 – Disposições de alto-falantes suportadas pelo ARSUS.

Por fim, a saída de fluxo de áudio é conectada à saída de áudio compatível com a configuração de espacialização, ou seja, se a espacialização é feita em quatro canais, o módulo de criação de *subpatches* cria uma saída de áudio com quatro canais e assim por

diante. O Pure Data é compatível com a maioria dos dispositivos de difusão multicanais, isto é, placas de som ou interfaces de áudio multicanais que devem ser utilizadas caso a forma de espacialização sonora seja diferente do estéreo.

4.3 Requisitos de Uso do ARSUS

Para a utilização do ARSUS como software para a manipulação do som em um ambiente espacializado, são necessários alguns requisitos mínimos de *hardware* e *software*, como:

- GPS, Bluetooth, infra-vermelho, etc;
- Processador Intel Core 2 duo ou mais recente;
- 2 GB de memória RAM;
- 100 MB de espaço em disco;
- Roteador WIFI ou rede LAN.

Os requisitos de *software* para a execução do ARSUS são:

O ARSUS framework foi desenvolvido no ambiente de programação Pure Data (PD) e somente pode ser executado dentro do ambiente, portanto, requer sua instalação para que seja utilizado. No entanto, o PD é um software multiplataforma podendo ser instalado nos seguintes sistemas operacionais: GNU/Linux e derivados, MacOS ou Windows;

Algumas bibliotecas específicas são necessárias e não estão presentes na instalação padrão do PD, como: **HCS**, **Iemguts**, **Net**, **OSC**, **Cyclone** e **Pan**. No entanto, essas bibliotecas podem ser instaladas através do repositório do software pela função “*Find Externals*” (busca embutida no PD) ou acessando seu site e fazendo o *download* e instalação manualmente.⁷ Somente a biblioteca *OpenAudience* não pertence ao repositório do PD, sendo necessário o *download* em sua página e instalação manual.⁸

A comunicação das aplicações cliente com ARSUS é feita através de rede *LAN* ou *Wireless* e devem utilizar o protocolo OSC.

As mensagens de *feedback* do ARSUS podem ser personalizadas da forma que for necessária para a melhor compreensão. Novos *feedbacks* podem ser adicionados ao código se assim o usuário desejar, bastando somente adicionar um bloco de impressão na saída de dados.

⁷ Como instalar novas bibliotecas e *externals* manualmente no Pure Data (material em ingles) <<https://puredata.info/docs/faq/how-do-i-install-externals-and-help-files>>

⁸ Disponível em: <<http://www.lsi.usp.br/neac/pt-br/openaudience>>

Dentre as limitações de uso, estão: a quantidade de sons diferentes que podem ser espacializados é sempre igual ao número de objetos que o ARSUS consegue espacializar. Ainda assim, o ARSUS consegue espacializar vários objetos com o mesmo som sem problemas. No entanto, o limite de objetos sendo espacializados em tempo real e ao mesmo tempo é delimitado pela capacidade computacional da máquina na qual o ARSUS está sendo executado. A quantidade máxima testada até o momento é de mil objetos, o teste prático foi realizado com uma aplicação cliente que visava testar o ponto de saturação do som. A discussão desse teste está detalhada no Capítulo 5.

4.4 Conclusão

Após o processo de implementação da melhor solução segundo o *Design Thinking*, percebeu-se a potencialidade desta solução para se tornar um *framework* para espacialização sonora. Sendo assim, a decisão de converter a aplicação servidora desenvolvida através do DT, descrito anteriormente, em *framework*, implicou na necessidade de projetar uma arquitetura generalista que desse suporte à espacialização sonora. Ainda neste capítulo, foram descritos os detalhes mais aprofundados acerca do funcionamento do ARSUS, requisitos para a espacialização sonora e requisitos de uso do *framework*.

Durante o processo de ajustes e refinamentos da melhor solução para se tornar um *framework*, foram modificadas e adicionadas novas funcionalidades que permitem novas possibilidades de uso. O módulo que mais recebeu modificações e adições desde do terceiro protótipo, descrito no Capítulo 3, foi o módulo de comunicação. Esse módulo recebeu mais duas opções de entrada em seus argumentos, adicionando a reprodução em *loop* de áudios distintos, e o comando para reproduzir o áudio contido nos objetos de interação remotamente, quando a opção de reprodução em *loop* não está ativada.

Sendo assim, essa nova implementação é um novo protótipo que necessitou de avaliação. Foram realizados testes com o ARSUS em pelo menos três cenários diferentes, a saber: o ponto de saturação do som, ou seja, quantos sons sendo reproduzidos ao mesmo tempo são necessários para saturar completamente o som; um segundo cenário é similar ao que o ARSUS foi já submetido, no entanto, decidiu-se mais uma vez testar seu desempenho no contexto de criação de música colaborativa espacializada, e por fim, o ARSUS foi testado na espacialização dos elementos sonoros do clone do jogo *Pac-Man*, fazendo com que o jogo recebesse um suporte ao áudio 3D. Todos esses testes são detalhados e podem ser conferidos no capítulo a seguir.

5 Avaliação do *Framework*

Neste capítulo é discutida a aplicação do ARSUS *framework* em três contextos diferentes. Durante os testes, o ARSUS não precisou de nenhuma mudança significativa em suas configurações para atender às três aplicações cliente. Como dito anteriormente, o ARSUS pode ter seu código personalizado para atender uma necessidade em específico, como um número maior de alto-falantes, ou mesmo a mudança de suas mensagens de *feedback* no console, reprodução de som em *loop* ou não, etc.

O ARSUS, enquanto aplicação servidora do sistema de composição colaborativa de música interativa espacializada, tinha em sua essência o foco na interação colaborativa para a criação de música eletroacústica e sendo aplicável somente a esse contexto. Ao evoluir para *framework*, o ARSUS deixa de fazer parte de um sistema colaborativo para dar suporte computacional a qualquer aplicação que necessite espacializar sons, sendo ela colaborativa ou não. A fim de verificar sua contribuição, neste capítulo é avaliada a aplicação do ARSUS nos três cenários diferentes detalhados a seguir.

5.1 ARSUS *Framework* na Avaliação do Ponto de Saturação do Som

A demanda por este estudo surgiu a partir de pesquisas realizadas por um dos professores da FAARTES, que necessitava verificar qual seria o ponto de saturação aproximado do som, isto é, quantos sons diferentes sendo reproduzidos, ao mesmo tempo, são necessários para que o som se torne somente um grande ruído, similar à estática das televisões antigas.

Para simular esse cenário, foi escrito um software na linguagem *Processing* para se comunicar com o ARSUS e enviar solicitações de reprodução de som para os vários objetos criados; Figura 29. *Processing* é uma linguagem de programação de código aberto e ambiente de desenvolvimento integrado (IDE), construído para as artes eletrônicas e comunidades de projetos visuais com o objetivo de ensinar noções básicas de programação de computador em um contexto visual e para servir como base para cadernos eletrônicos.¹

O software funciona da seguinte maneira: deve-se digitar a quantidade de instâncias que se deseja testar, como no momento só haviam vinte sons disponíveis para teste, foi necessário repeti-los várias vezes de forma aleatória.

¹ Disponível em: <<https://processing.org/>>

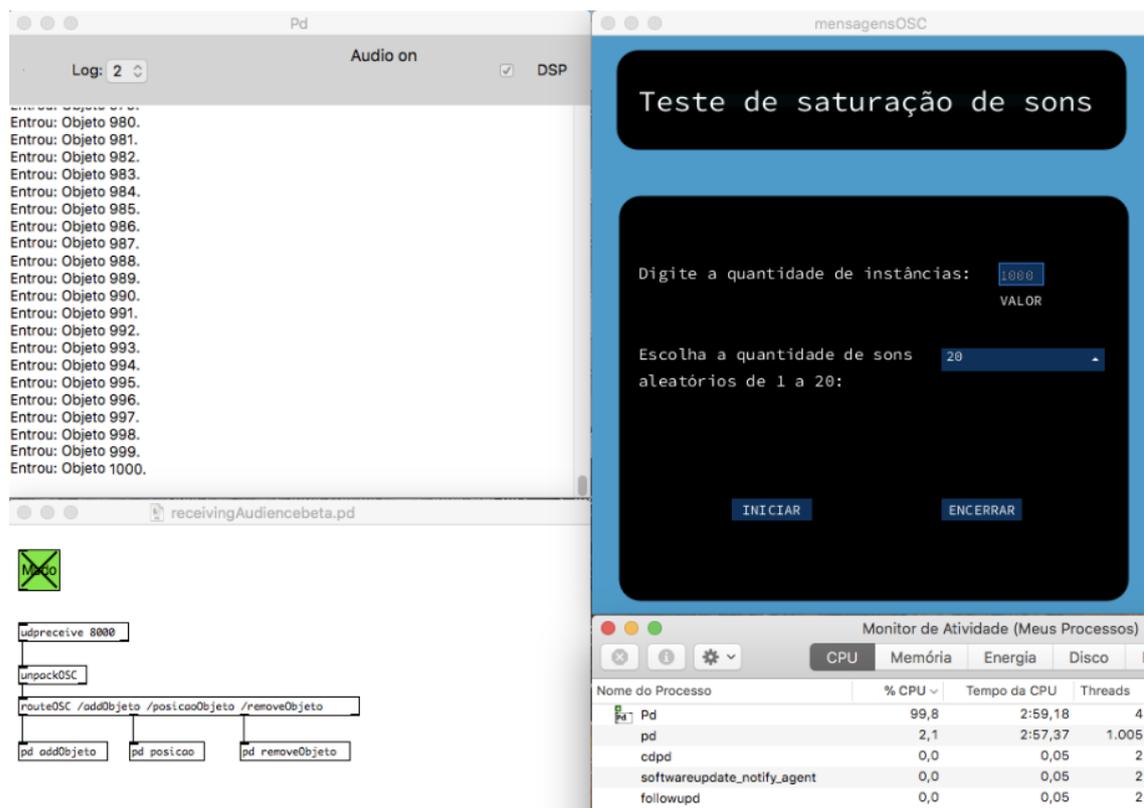


Figura 29 – Aplicação do ARSUS para testar o ponto de saturação do som.

5.1.1 Conceitos Sonoros

Para um melhor entendimento do assunto aqui discutido, é necessária a apresentação de alguns conceitos acerca do som como, por exemplo:

Ondas Sonoras - O som pode ser descrito através de uma sequência de ondas sonoras, que são ondas de deslocamento, densidade e pressão que se propagam pelos meios compreensíveis. Para que sejam representadas no meio digital, seu comportamento analógico (contínuo) tem que ser convertido numa série de valores discretos (descontínuos). Esses valores são números (dígitos) que representam amostras (*samples*, em inglês) instantâneas do som. Isso é realizado por meio de um conversor analógico/digital (CAD) (IAZZETTA, 2017).

Frequência - A frequência de uma onda sonora é uma grandeza física ondulatória que indica o número de ciclos (oscilações) durante um período de tempo (SOARES, 2017). Para determinar a frequência, basta calcular:

$$Frequência = \frac{númerodeoscilações}{tempo(intervalodetempo)}$$

Amplitude - A amplitude de uma onda sonora é a medida da extensão de uma perturbação durante um ciclo da onda. A amplitude de uma onda sonora pode permanecer constante, sendo classificada como uma onda contínua, ou pode variar de acordo com o tempo (SOARES, 2017) .

Como podemos ver na Figura 30, a distância Y é a amplitude da onda.

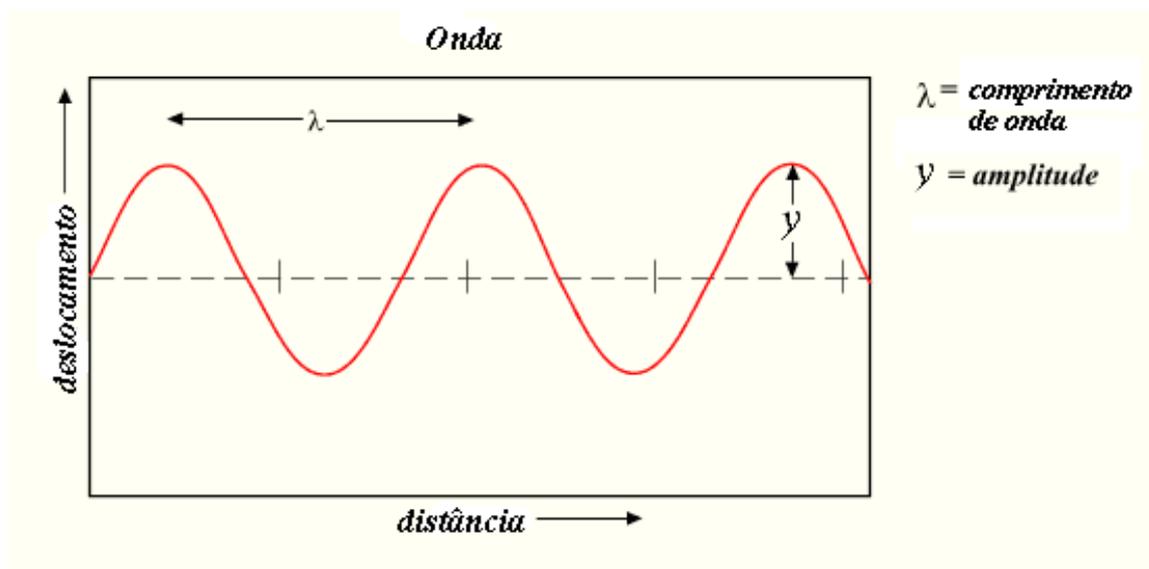


Figura 30 – Características de uma onda sonora.

O comprimento de uma onda sonora é a distância entre valores repetidos num modelo de onda. Essa medida é normalmente representada pela letra grega lambda (λ).

Clipping - Uma vez que a extensão dinâmica do áudio digital é determinada pelo número de bits utilizados, não é possível representar valores acima de um determinado limite. O valor mais alto que pode ser representado geralmente é expresso como sendo "0" decibéis. Se a amplitude da onda ultrapassa esse valor, ocorre um corte (*clipping*) da crista da onda, mudando sua forma original e ocasionando uma distorção no som Figura 31 (IAZZETTA, 2017).

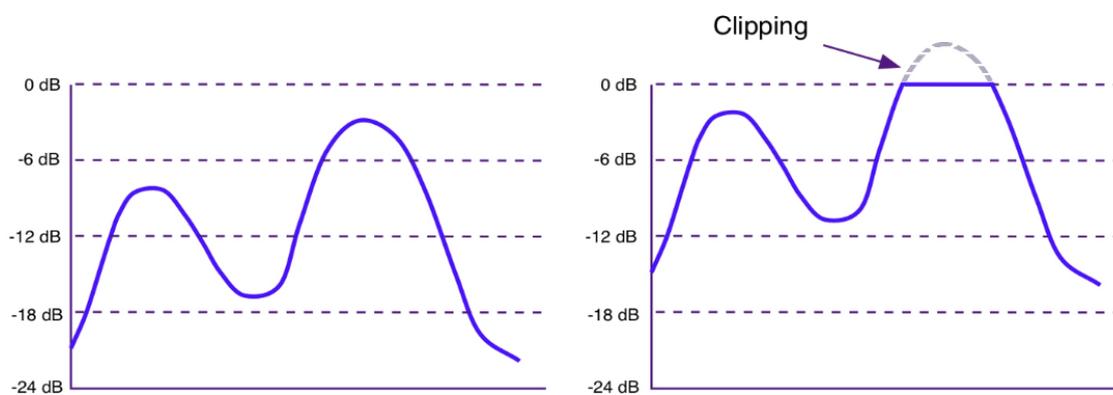


Figura 31 – Exemplos de ondas sonoras: à esquerda, dentro do limite de distorção; à direita, onda sonora ultrapassando o limite de distorção (IAZZETTA, 2017).

5.1.2 Execução

O estudo foi realizado em um notebook com um processador Intel i7 Quad-Core 2.4 GHz, com 8 GB de memória RAM, SSD de 240 GB e Sistema Operacional MacOS 10.13.1. Foram utilizadas 20 amostras de sons de pássaros da fauna amazônica para a realização deste estudo.

Em um primeiro momento, foram geradas 20 instâncias repetindo somente um som e, em seguida, cinquenta instâncias. Em uma nova rodada de testes, foi-se aumentando o número de instâncias utilizadas para números mais expressivos, como: cem, duzentas, trezentas, quinhentas, seiscentas e mil instâncias de sons aleatórios com duração de, no máximo, doze segundos.

5.1.3 Discussão

A execução deste estudo permitiu avaliar, de uma forma prática, grandes quantidades de sons aleatórios sendo reproduzidos ao mesmo tempo. Esse processo, se realizado manualmente, pode levar dias ou meses para chegar ao mesmo número de amostras que o ARSUS pode suportar.

A primeira rodada de teste não foi considerada, devido ao baixo número de instâncias, e uma análise preliminar as considerou insignificantes.

A segunda rodada de testes teve início com cem amostras e seguida de outro teste realizado com duzentas amostras, a Figura 32 ilustra o resultado obtido ao término destes dois testes e mostra, na parte de cima, um espectro ainda bem limpo, e ao ser reproduzida a amostra permitia ainda a percepção de diferentes sons sem muita dificuldade.

Ao dobrar o número de amostras, já é possível notar, no espectro abaixo, uma mudança de formato, tornando-se um espectro mais denso. Mesmo assim, ainda era possível notar com alguma facilidade detalhes e identificar alguns sons. É possível também notar uma certa deformação no formato geral do espectro. Nesse caso, analisando o som com base no estudo do som, já há uma distorção na amostra, mas não ao ponto de o som estar inaudível.

Na amostra com trezentos sons gerados, Figura 33, o espectro se mostra mais denso, e ainda se notam muitos picos, geralmente são alguns sons mais agudos que acabam se destacando. A sensação audível diminui, mas ainda é possível notar os sons que se destacam.

Na amostra com quinhentos sons, essa sensação de definição que permitia a identificação de sons diminui bastante e o espectro se mostra um pouco mais uniforme e denso. Nesses casos, há uma distorção ou *clipping* bastante elevada se comparada às amostras anteriores.

Analisando visualmente, a partir de quinhentos sons, o espectro se manteve parecido com a amostra de seiscentos, e pouca coisa mudou para com a amostra de mil

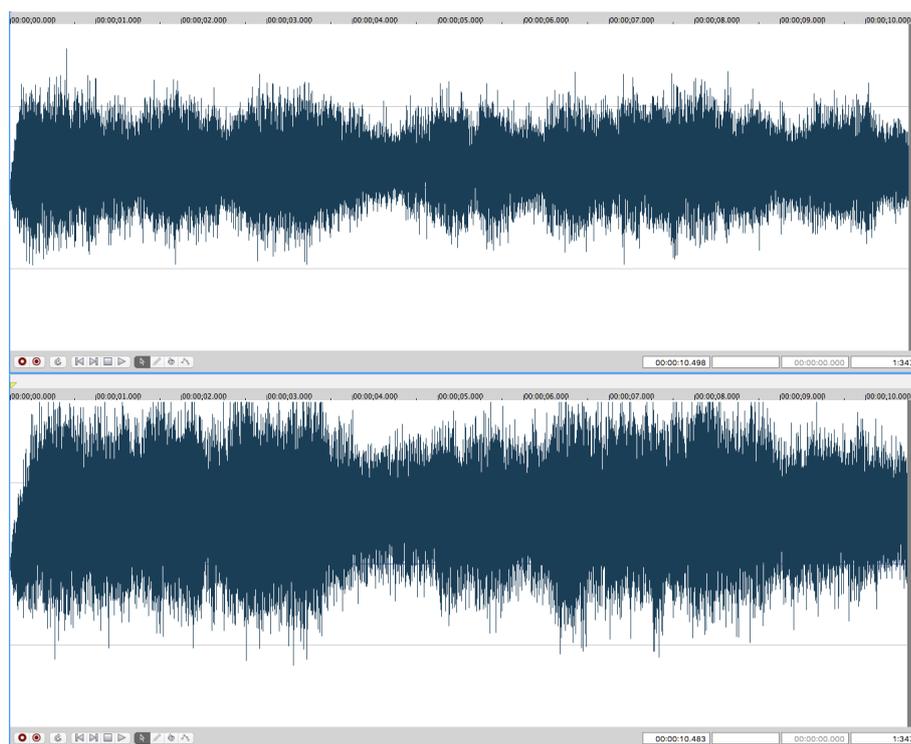


Figura 32 – Em cima, a amostra de som gerada com cem instâncias e, em baixo, com duzentas instâncias.

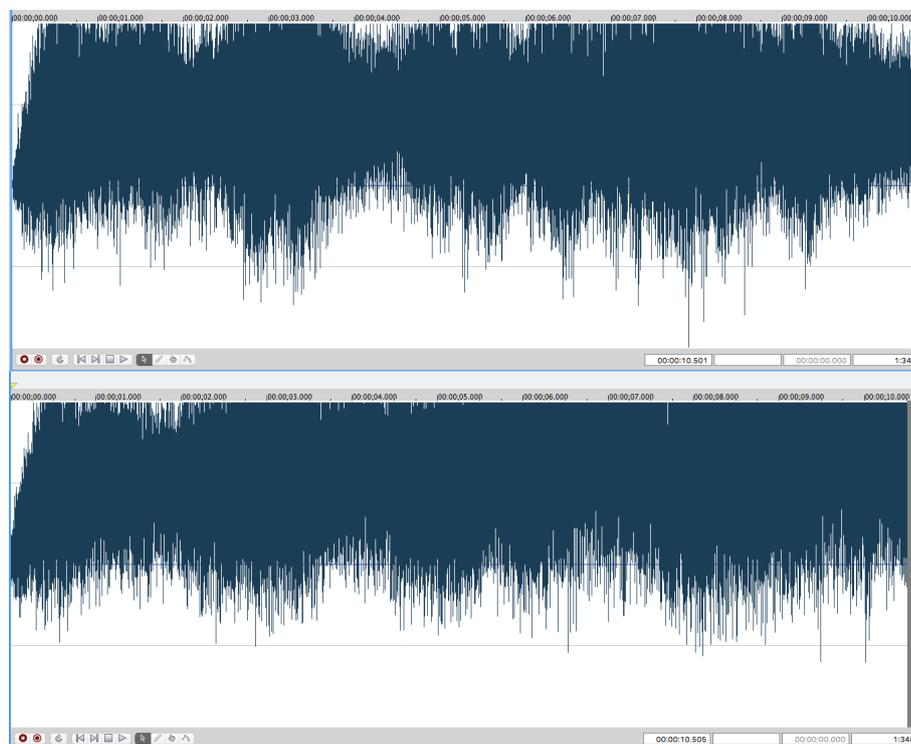


Figura 33 – Em cima, a amostra de som gerada com trezentas instâncias e, em baixo, com quinhentas instâncias.

sons.

Ao analisar o áudio, notou-se uma grande diferença entre as amostras com seiscentos sons da amostra com mil sons. A sensação audível, neste caso, já é bastante prejudicada, e já é bem difícil identificar algum som em específico. No entanto, o som do pássaro socó-boi possui um som bastante forte, o qual se destacou dentre todas as amostras anteriores, por possuir um canto que lembra o mugido de um boi, o que originou o seu nome. Na Figura 34, é possível verificar o espectro de áudio gerado após esse teste, sendo a primeira amostra de cima com seiscentos sons e a de baixo com mil sons sendo executados ao mesmo tempo.

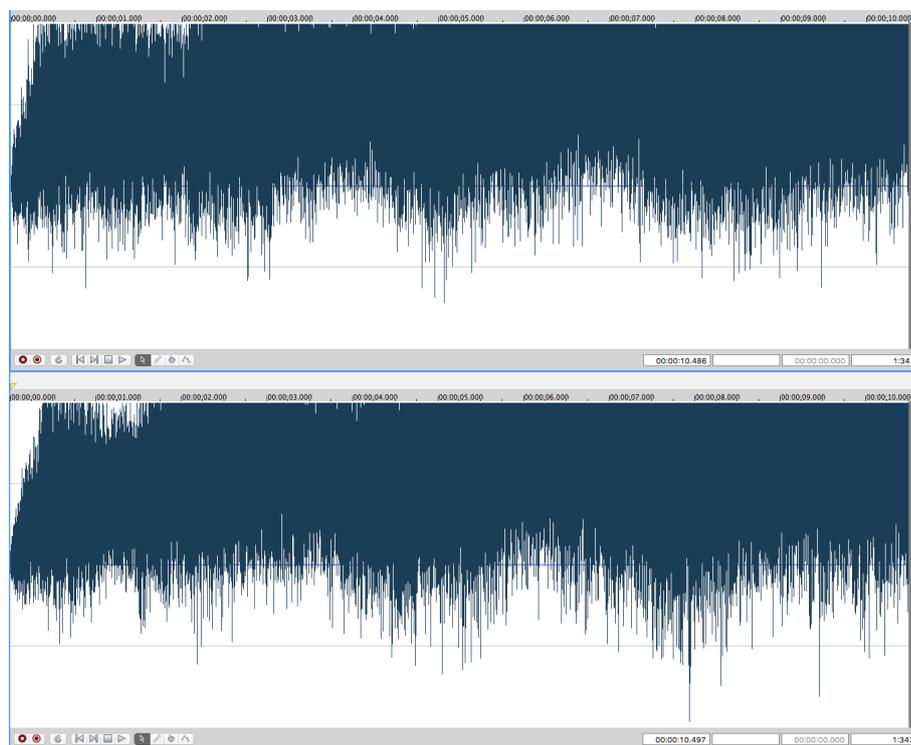


Figura 34 – Em cima, a amostra de som gerada com seiscentas instâncias e, em baixo com mil instâncias.

Na Figura 29, é possível verificar no monitor de atividade do sistema operacional, o consumo de recursos computacionais durante a execução do teste, em especial o segundo processo que se refere à quantidade de instâncias geradas, representada pelo número de *threads* requeridos para execução do estudo. Nesse caso, a máquina em que o estudo estava sendo realizado começou a apresentar lentidão na execução do teste, em específico ao gerar as mil instâncias de sons. Isto se deve ao ambiente de execução do teste, o software Pure Data, não possuir a técnica de programação concorrente, a qual utiliza mais de um núcleo do processador quando disponível. Ao utilizar somente um núcleo o ambiente, tende a consumir bastante recursos computacionais.

Apesar de os testes anteriores terem mostrado uma saturação no som, ainda não se tinha chegado ao resultado esperado. Mesmo com mil sons sendo executados ao mesmo

tempo, ainda foi possível identificar alguns sons, mesmo que com alguma dificuldade, e a saturação do som ao ponto de só existir estática ainda não tinha sido alcançada.

Devido às dificuldades para executar o teste com mais de mil sons no Pure Data, decidiu-se então nessa etapa utilizar um software de edição de áudio para duplicar o resultado do teste anterior para verificar se haveria diferenças notáveis.

A Figura 35 ilustra como ficou o espectro da amostra sonora, que sofreu uma grande “elevação” em relação ao eixo horizontal e o nível de “clipping” em relação ao espectro da amostra com mil sons é visivelmente maior.

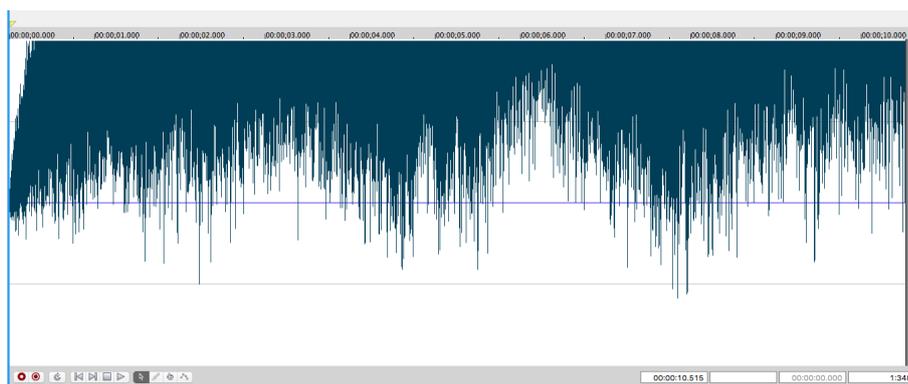


Figura 35 – Ponto de saturação aproximado em duas mil instâncias.

Nesse caso, a sensação audível é totalmente prejudicada e o áudio está com um nível de saturação bastante elevado, parecendo-se com o som de estática mencionado no começo desta seção.

Mesmo tendo um limite de mil sons sendo executados de uma só vez devido às limitações de desempenho da máquina na qual estava sendo executado, o ARSUS pode diminuir consideravelmente o tempo necessário em testes para se aproximar do ponto de saturação do som, como foi dito anteriormente, poderia levar meses para que se chegasse neste mesmo resultado.

5.2 ARSUS *Framework* na Composição Colaborativa de Música Eletroacústica Especializada

Apesar do ARSUS ter sido criado neste contexto, ao sofrer as modificações para que se tornasse um framework, foi necessário avaliar mais uma vez o seu comportamento com usuários reais, a fim de captar o *feedback* para avaliação desta nova versão.

5.2.1 Planejamento

Durante a fase de planejamento, as questões de pesquisa levantadas no estudo do terceiro protótipo foram mantidas e foram adicionadas mais duas. Tais questões eram

relacionadas ao seu contexto, à seleção dos temas e às tarefas a serem realizadas pelos participantes do estudo.

1. Questões:

- Os usuários perceberão sua interação no espaço de interação?
- Os usuários perceberão o *feedback* de áudio no alto-falante do ambiente?
- Os usuários perceberão a composição colaborativa em tempo real?
- Quão agradável pode ser a experiência do usuário ao interagir com o ambiente?
- Quão agradável pode ser para o usuário, colaborar para uma composição coletiva?

2. Contexto e Assuntos: este estudo foi realizado no Instituto de Ciências Exatas e Tecnologia - ICET, da Universidade Federal do Amazonas, com dezesseis alunos de cursos aleatórios ali presentes no momento da execução do estudo; Figura 36. Os voluntários não estavam familiarizados e conheciam os conceitos principais do projeto. Um ponto a ser destacado é que assim como os voluntários do estudo de avaliação do terceiro protótipo construído, os alunos participantes deste estudo não tinham nenhum conhecimento avançado sobre composição, produção ou teoria musical.

Considerando o espaço de interação para esse contexto, definiu-se um pequeno espaço na área de convivência do Instituto. Para limitar esse espaço, o intervalo do roteador que desempenha o papel de ponto central tinha de ser limitado. Foram posicionados quatro alto-falantes amplificados nos quatro cantos do espaço de interação para o *feedback*.

3. Definição das tarefas: Os sujeitos participantes desse estudo tiveram que realizar seis tarefas principais: (A) Instalar o Compomus em seus dispositivos móveis Android; (B) Preencher o formulário de inscrição e registrar a candidatura; (C) Selecionar um som para representá-los no espaço de interação; (D) Interagir livremente entrando e saindo do espaço de interação; (E) Alterar o som; (F) Completar um questionário após o estudo.

5.2.2 Execução

Após a montagem da estrutura no local, alguns alunos foram convidados para participar do estudo. Devido à falta de voluntários dispostos a participar do estudo, não foi possível reunir um grupo com um número de participantes maior do que oito por rodada do estudo.



Figura 36 – Avaliação do ARSUS em um contexto de composição musical interativa e espacializada em tempo real.

Sendo assim, os voluntários que se dispuseram foram divididos em dois grupos com oito participantes cada. Como os voluntários não estavam familiarizados com o projeto, foram explicados os conceitos do trabalho e propósito do estudo para os dois grupos em seguida foram dadas as instruções de uso do Compomus e a dinâmica de interação com o ambiente.

A rede sem fio a qual o ARSUS e Compomus faziam uso, foi liberada para conexão e foi solicitado aos usuários que conectassem seus dispositivos. Em seguida, foi instruído aos participantes que acessassem o diretório disponibilizado no servidor web do Compomus para o *download* e instalação do Compomus. Ao fim da instalação nos dispositivos e registro dos voluntários, foi dado início de fato aos testes. Foi aconselhado que eles colocassem seus dispositivos em modo silencioso, a fim de facilitar a percepção de seu próprio som nos alto-falantes do espaço de interação.

Nesse estudo, foram testadas as mesmas duas versões do Compomus discutidas na Subseção 3.3.4, a versão da melhor solução, descrita no Capítulo 3, a qual faz uso do GPS embutido nos smartphones para determinar a localização do usuário, e um ponto central para determinar o espaço de interação. A segunda versão utilizada foi a versão de depuração que utiliza um “*joystick*” para simular a movimentação do usuário pelo ambiente, e que dispensa o uso do roteador como ponto central que delimita o espaço de interação. Os usuários ficaram livres para testar o ambiente da forma que achassem interessante, durante um período de 15 minutos e, ao fim da participação dos grupos de voluntários, foi solicitado que respondessem um questionário com onze perguntas sobre a experiência de uso do ambiente (disponível no Apêndice B).

5.2.3 Discussão

Neste estudo também foi possível observar todas as dimensões da interação do modelo de colaboração 3C (FUKS et al., 2007). O que já se era esperado devido a este estudo ser semelhante ao que foi aplicado anteriormente e relatado no segundo protótipo na Subseção 3.3.2. No entanto, o que pode-se destacar neste estudo é o *feedback* positivo dos usuários que ficaram bastante empolgados e elogiaram a iniciativa.

As duas primeiras questões deste estudo que tem a ver com a percepção da interação do usuário com o som, e o *feedback* de áudio, puderam ser respondidas com base nos dados coletados através do questionário aplicado ao final do estudo onde 87,5% dos usuários relataram que em meio à mistura dos sons ainda era possível identificar o seu som sem problemas, e 93,8% dos voluntários afirmaram que conseguiram espacializar o seu som corretamente como mostra a Figura 37.

Você conseguiu direcionar o som escolhido nas caixas adequadamente?

16 respostas

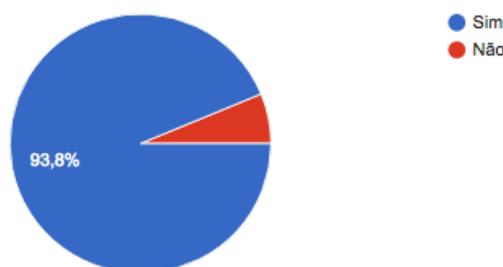


Figura 37 – Percentual de respostas dos usuários à pergunta realizada acerca da espacialização sonora. Apuração feita através das respostas do formulário aplicado após o estudo.

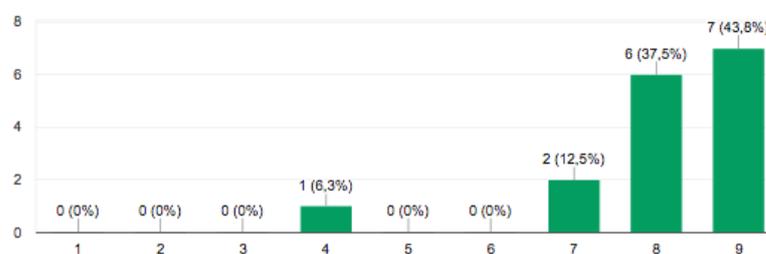
Em resposta ainda à questão da percepção da colaboração do usuário para a composição e às duas últimas questões foi possível verificar em uma escala de satisfação de 1 à 9, de acordo com a escala SAM (*Self Assessment Manikins*),² em que 12 de 16 alunos relataram um grau de satisfação acima de 7 ao se sentirem contribuindo para uma composição coletiva, vide Figura 38. Ainda na mesma imagem, é possível verificar o quão agradável foi a experiência dos usuários com este estudo.

Alguns alunos relataram, em comentários, que poder manipular o som ali na palma da sua mão foi uma experiência muito divertida. Tais afirmações podem ser exemplificadas nas citações: “me senti no controle do som, foi algo bem divertido”, “gostei deste experimento foi muito legal”. Outro comentário relatou o quão interessante foi a sensação do som estar lhe seguindo conforme citação: “É como se o som estivesse me seguindo”.

² O SAM utiliza uma abordagem cognitiva por meio de representações imagéticas (os Manikins), onde as emoções são encaradas como estímulos que respondem a eventos.

Na escala abaixo responda: Quão agradável foi a sua experiência neste estudo?

16 respostas



Você sentiu que o seu som contribuiu como parte de uma composição coletiva? Se sim, qual seria o grau da sua satisfação?

16 respostas

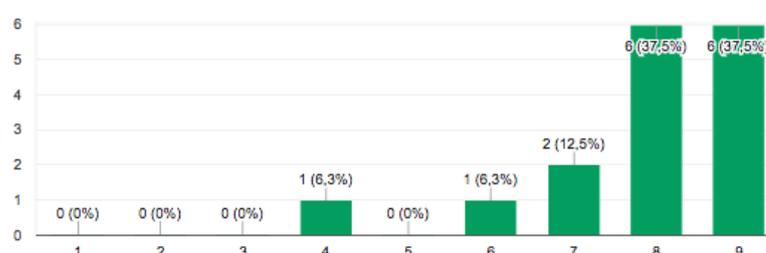


Figura 38 – Percentual de satisfação dos usuários acerca de sua experiência de uso do sistema e de sua colaboração para uma composição coletiva. O eixo Y representa a quantidade de alunos e o eixo X, o nível de satisfação/experiência relatados.

Outra aluna afirmou ser interessante o projeto e que não sabia que o som poderia ser manipulado desta forma, citação: “Nossa! bem loco(sic), não sabia que o som podia ser controlado desse jeito”.

Cerca de 43,8% dos usuários relataram problemas na instalação do Compomus em seus dispositivos via *download*, o App apresentava falhas quando instalado após ser baixado do servidor disponibilizado. O motivo deste problema ainda é desconhecido, mas foi contornado no momento do estudo ao ser realizada a instalação do Compomus via cabo de dados. No entanto, como mostra o resultado, a má experiência com relação aos problemas enfrentados não deixou de ser relatada pelos usuários, que fizeram afirmações como “foi instalado via WI-FI e apresentou erro”.

Outra dificuldade apontada por alguns dos usuários acerca da interface do Compomus, é a opção de cadastro de novo usuário estar pouco visível. No entanto, o Compomus segue o padrão de interfaces do Android. O procedimento foi explicado aos usuários que as opções não visíveis estavam armazenadas no menu de contexto do App, a porcentagem de usuários que relataram esta dificuldade é de 18,8% ou 3 usuários dos 16, conforme ilustra a Figura 39.

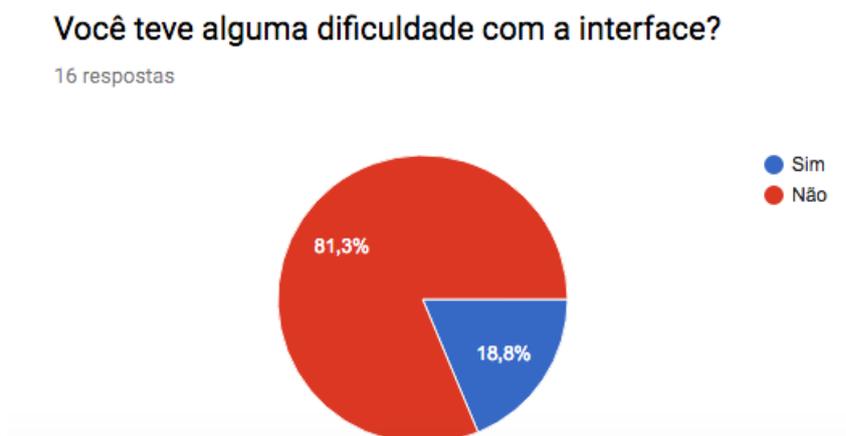


Figura 39 – Percentual de usuários que tiveram alguma dificuldade em utilizar a interface do Compomus.

5.3 ARSUS *Framework* no Contexto de Jogos

Para verificar a sua funcionalidade também no contextos dos jogos, buscou-se pela web um jogo de código aberto para lhe ser adicionado o atributo do som espacial. O jogo escolhido é um clone do clássico *Pac-Man*, o jogo é desenvolvido na linguagem PHP, CSS e Java Script, podendo ser executado diretamente pelo navegador.³

5.3.1 Execução

Este estudo foi realizado no modo estéreo em um notebook com um processador Intel i7 Quad-Core 2.4 GHZ, com 8 GB de memória RAM, SSD de 240 GB e Sistema Operacional MacOS 10.13.0, com fones de ouvido e um controle do tipo “*joystick*”.

Para promover uma sensação de um som envolvente para o jogo foi adicionado o suporte à comunicação com o ARSUS, fazendo uso de uma biblioteca OSC para JavaScript que funciona tanto no navegador quanto no Node.js.⁴ Foram adicionadas três funções para o envio de parâmetros como mensagens no código original do jogo. Uma função que adiciona um novo objeto de interação no ARSUS quando o jogo inicia, ou reinicia. Outra função para enviar os dados de localização com as coordenadas X e Y para o ARSUS. E por fim, uma função que exclui o objeto de interação quando o personagem morre; Figura 40.

Somente o personagem *Pac-Man* teve seu som espacializado, pois este é o único que reproduz som ao mover-se. A funcionalidade de reproduzir o som do personagem foi desabilitada no código original do jogo sendo reproduzida somente pelo ARSUS. Os outros elementos de som do jogo permaneceram inalterados.

A dinâmica geral da espacialização do jogo é bem simples. Ao iniciar uma nova

³ Disponível em: <<https://github.com/luciopanepinto/pacman>>

⁴ Disponível em: <<https://github.com/colinbdclark/osc.js/>>

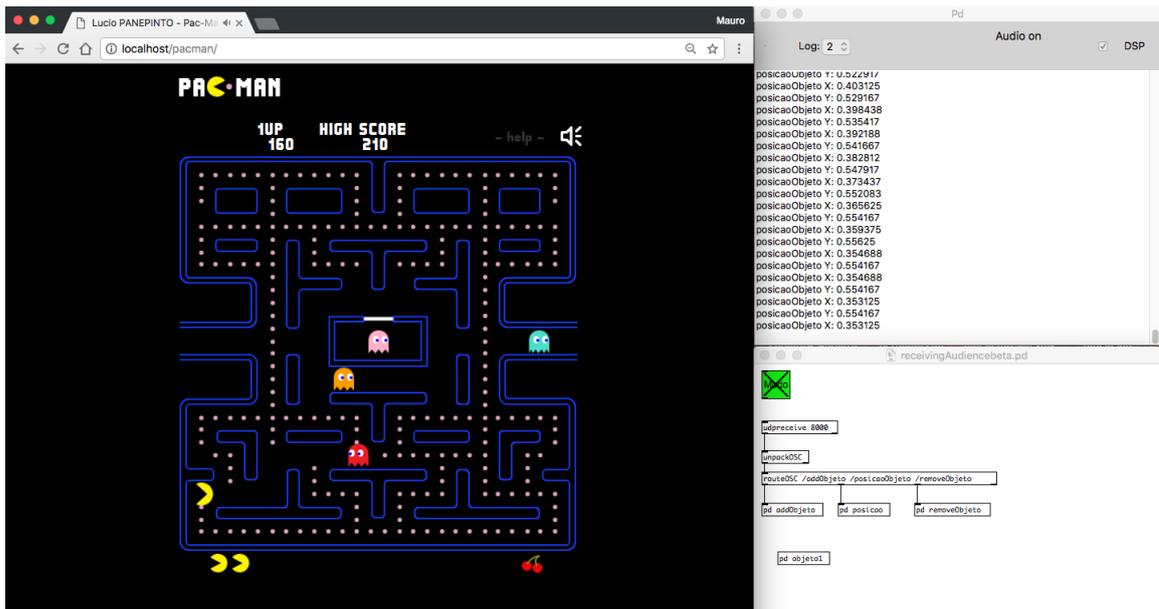


Figura 40 – Avaliação do ARSUS aplicado à espacialização do jogo *Pac-Man*.

partida, é enviada uma nova mensagem para o ARSUS, podendo ser executado no mesmo computador ou remotamente na mesma rede, e este adiciona um novo objeto com o som do personagem. Ao mover-se, uma sequência de mensagens são enviadas para o *framework* atualizando sua posição no plano cartesiano e este as processa e espacializa o jogo em tempo real. Quando o personagem morre, outra mensagem é enviada ao ARSUS ordenando a destruição do objeto.

5.3.2 Discussão

A execução deste estudo verificou a aplicabilidade do ARSUS como uma ferramenta que adiciona a espacialização sonora em jogos que não a possuíam em seus projetos originais. Apesar de ter sido testado no modo estéreo, o jogo ganhou suporte aos modos de espacialização: estéreo, quadrafônico, *surround*, hexagonal, octogonal, bastando somente ativar o modo de espacialização desejado no *framework*.

Um problema que foi notado na implementação durante o teste da espacialização sonora no jogo, é que quando o personagem está parado o som continua sendo reproduzido. No entanto, o comportamento padrão do personagem é emitir o som somente quando este “comer” um item. Este comportamento logo foi corrigido após o teste, mudando as funções adicionadas de lugar no código.

A Figura 41 ilustra a parte do código em que foi inserida as funções para o envio de mensagens para o ARSUS, neste caso o código foi inserido na função de contador da pontuação do personagem, quando esta é incrementada, são executadas as funções no lugar da função para reproduzir o som de “comer” do personagem.

A dinâmica da espacialização então passou a se dar da seguinte maneira: quando

```
score( SCORE_BUBBLE );
//playEatingSound();

oscPort.on("ready", function () {
  oscPort.send({
    address: "/addObjeto", //Endereço da mensagem a ser enviada
    args: [
      {
        type: "i", //Tipo de dado enviado "Inteiro"
        value: 1 //ID de som
      },
      {
        type: "i",
        value: 2 //ID de objeto
      }
    ]
  });
  oscPort.send({
    address: "/posicaoObjeto", //Endereço da mensagem a ser enviada
    args: [
      {
        type: "f", //Tipo de dado enviado "Float"
        value: PACMAN_POSITION_X //Posição X
      },
      {
        type: "f",
        value: PACMAN_POSITION_Y //Posição Y
      },
      {
        type: "i", //Tipo de dado enviado "Inteiro"
        value: 2 //ID de objeto
      }
    ]
  });
  oscPort.send({
    address: "/removeObjeto", //Endereço da mensagem a ser enviada
    args: [
      {
        type: "i",
        value: 2 //ID de objeto
      }
    ]
  });
});
});
```

Figura 41 – Funções inseridas no código original do jogo com as mensagens OSC a serem enviadas ao ARSUS.

o personagem “come” um item, é enviada uma mensagem ao ARSUS que adiciona um novo objeto e reproduz o som característico do personagem. Em seguida, em milésimos de segundo, é enviada uma mensagem atualizando a localização do personagem e então é enviada uma última mensagem ordenando o fim da reprodução do som, fazendo com que o procedimento todo que dure em média um segundo. Devido à velocidade de transporte de dados do protocolo não foi possível notar diferença no tempo de resposta e reprodução do som na versão original e a versão especializada do jogo.

Os resultados dos testes foram bastante animadores, pois foi possível adicionar a espacialização sonora a um jogo que não possuía suporte para tal utilizando o *framework*, necessitando somente um conhecimento básico de programação. A sensação de imersão no jogo teve um aumento significativo somente na espacialização sonora do personagem principal, espera-se obter uma maior imersão ao adicionar a espacialização a todos os sons dos personagens do jogo em uma versão futura.

5.4 Conclusão

Neste capítulo, são mostrados os estudos de verificação do ponto de saturação do som; ARSUS na composição colaborativa de música eletroacústica espacializada e por fim, ARSUS no contexto de jogos. Esses estudos visavam avaliar o uso do *framework* desenvolvido em pelo menos três cenários distintos, com o intuito de verificar sua aplicabilidade a múltiplos contextos.

O primeiro estudo investigou o ponto de saturação do som, em que se desejava saber quantos sons sendo reproduzidos ao mesmo tempo seriam necessários para que o som resultante fosse parecido com o som emitido por televisões mais antigas quando estas não estavam sintonizadas em nenhum canal. O ARSUS conseguiu promover uma pesquisa mais rápida neste cenário, pois os métodos disponíveis para a realização deste estudo são manuais, necessitando-se sobrepor manualmente um som sobre o outro. O software escrito em *processing*, possibilitou a automatização desse processo ao solicitar que o ARSUS reproduzisse múltiplos sons ao mesmo tempo. Sendo assim, ao final do estudo, o arquivo de áudio gerado continha todas as instâncias de som em apenas uma amostra.

A falta de suporte ao processamento multi-núcleo do ambiente Pure Data acabou por limitar a quantidade de sons sendo reproduzidos ao mesmo tempo, em mil instâncias. A partir dessa limitação, fez necessária a utilização de um software de edição de áudio para a duplicação da amostra, pois em mil instâncias ainda não se tinha chegado ao resultado esperado. Mesmo com a limitação imposta pela linguagem e ambiente, o ARSUS contribuiu de forma significativa em conjunto com a aplicação desenvolvida em *processing* para se chegar ao resultado esperado de maneira mais rápida, e eficiente do que as metodologias disponíveis para se chegar ao mesmo resultado.

O segundo estudo foi conduzido de forma similar ao estudo relatado na Subseção 3.3.4 do Capítulo 3, com o propósito de avaliar as mudanças feitas no código do ARSUS e avaliar seu comportamento após estas modificações. Nesse estudo foi possível notar um aumento da estabilidade e desempenho do ARSUS em comparação à última versão testada nesse contexto. Os participantes do estudo relataram uma boa experiência de uso tanto do ambiente quanto do Compomus, ao utilizar a versão padrão do sistema com GPS e WIFI, e também a versão de depuração onde é possível controlar a localização do som manualmente.

No terceiro estudo o ARSUS foi utilizado em conjunto com um clone do jogo *Pac-Man*. O jogo foi escrito em PHP e *JavaScript* e teve adicionado em seu código o suporte à espacialização sonora por meio do protocolo de comunicação OSC. O jogo utiliza os sons monofônicos do título original, e mesmo sendo reproduzido em alto-falantes ou fones de ouvido estéreo, não proporciona nenhuma sensação de som espacializado. Ao ser utilizado em conjunto com o ARSUS, o jogo ganhou o suporte à espacialização sonora trazendo ao

jogador uma imersão maior ao jogo. Para tal, foi necessário somente que o jogo enviasse, via OSC, as localizações dos personagens e objetos dentro do tabuleiro para o ARSUS e este processasse e espacializasse o som em tempo real.

6 Conclusão

A espacialização sonora tem sido estudada durante décadas e a evolução das tecnologias tem permitido com que a espacialização sonora seja bastante explorada na atualidade, não somente na música ou no cinema, mas também na arte, jogos e navegação de pessoas com deficiência visual. Novas tecnologias como a realidade virtual e jogos eletrônicos têm utilizado a espacialização sonora para uma experiência nova para seus utilizadores. Artistas performáticos e músicos também têm explorado a espacialização sonora como fonte de originalidade. No entanto, em todos esses cenários há a figura do projetista de som ou engenheiro de áudio para definir o comportamento do som durante uma apresentação, filme ou jogo.

Neste trabalho, a espacialização sonora foi associada à colaboração para proporcionar a composição de música eletroacústica de forma descentralizada em que não existe um artista principal ou projetista de som.

A ideia de proporcionar uma tecnologia em que o público também pudesse ser o compositor, ao colaborar para a composição sendo realizada em tempo real, deu origem ao problema considerado neste trabalho:

Como dar suporte computacional à espacialização sonora realizada em tempo real em um espaço definido, considerando diferentes fontes sonoras sendo espacializadas por atores distintos de forma colaborativa, espontânea e aleatória, sem a necessidade da presença de uma figura centralizadora para orquestração.

Posto isto, foram levantadas as seguintes questões de pesquisa:

- Como espacializar diferentes sons a serem manipulados por diversos usuários em tempo real?
- Como proporcionar uma interação dos usuários com o som?
- Os usuários serão capazes de se sentir contribuindo em um ambiente espacializado?

Com relação à questão de pesquisa, a qual investigava como se espacializar diferentes sons manipulados por diversos usuários, foi resolvida construindo-se um *framework* o qual permite que sejam espacializados diversos sons de forma independente. Sua espacialização pode ser controlada remotamente por um ou vários dispositivos, bastando somente enviar mensagens no formato do protocolo OSC com as entradas requeridas como argumentos. O *framework* desenvolvido permite que sejam alteradas suas configurações e código para que se adeque ao contexto a ser utilizado, suportando vários métodos de espacialização como: Estéreo, Quadrafônico, *Surround*, Hexagonal, Octogonal e Octogonal Cúbico.

Este *framework* foi avaliado em pelo menos três contextos diferentes detalhados no Capítulo 5, em que duas aplicações cliente fizeram uso de suas funcionalidades para espacialização sonora, uma voltada para o cunho artístico, que desejava promover composições de música eletroacústica colaborativamente e a outra foi uma integração de um clone do jogo *Pac-Man* desenvolvido em PHP com o *framework*, a qual foi possível adicionar a espacialização sonora trazendo uma sensação de imersão para o jogo e, por último, uma aplicação desenvolvida em *processing* colocou a escalabilidade do *framework* à prova ao testar o ponto de saturação aproximado do som.

Em resposta à questão de pesquisa, como proporcionar uma interação dos usuários com o som. Durante essa pesquisa, foram estudadas várias soluções que visavam proporcionar ao usuário uma interação com o som de maneira que este se sentisse realmente interagindo com o som. Essa pesquisa adotou a abordagem do *Design Thinking* para desenvolver estas soluções e que podem ser conferidas no Capítulo 3, onde a melhor solução é baseada no uso de smartphones para detecção da localização do usuário em um espaço de interação. Foi desenvolvida uma aplicação na plataforma Android em que o usuário poderia escolher um som disponível e movimentar-se em um espaço destinado à interação. Ao movimentar-se pelo espaço, o som escolhido por esse usuário era espacializado de acordo com a sua localização, dando a impressão do som estar seguindo-lhe.

Durante os estudos relatados no Capítulo 5, foi possível responder à última questão de pesquisa em que se desejava verificar se os usuários seriam capazes de se sentir contribuindo em um ambiente de interação. Nesse caso, o *feedback* de áudio proporcionou aos usuários participantes do estudo realizado para avaliar o *framework* uma sensação de estarem envoltos pelos sons. Ao identificar seu som dentre os outros, os usuários verificaram que os sons sendo reproduzidos durante o estudo não eram somente um punhado de sons aleatórios e, ao movimentar-se, puderam confirmar que sua interação era respondida virtualmente ao confirmar que o som era projetado para outro lugar no espaço de interação.

Como contribuições deste trabalho, podemos enumerar:

1. Criação de uma ferramenta computacional que fornece suporte à espacialização sonora em tempo real para aplicações cliente, seja no âmbito artístico, de pesquisa, educacional ou musical, proporcionando uma utilização fácil, sem requerer um conhecimento de programação ou fundamentos teóricos da música aprofundados;
2. Esta ferramenta utiliza como base as camadas três e quatro da arquitetura AUDI-ENCE para proporcionar a espacialização sonora. Entretanto, alguns componentes da biblioteca existentes nas camadas um e dois foram classificados como proprietários, não sendo disponibilizados em sua versão de código aberto. A implementação deste *framework* fecha a lacuna deixada pelos autores, ao proporcionar uma ferramenta para espacialização sonora completa com componentes alternativos aos

- inexistentes na versão de código aberto da biblioteca AUDIENCE;
3. Estudos com a espacialização musical possibilitando a composição colaborativa de música eletroacústica fazendo uso de tecnologias emergentes;
 4. Novas tendências para a criação e reprodução de áudio espacializado. Através deste *framework*, o campo de áudio espacial tem suas divisas ampliadas;
 5. Desenvolvimento de um software que utilizado em conjunto com o *framework* proposto, pode-se verificar o ponto de saturação do som.

O *framework* proposto se comportou bem e de maneira estável sem lentidão ou atraso na reprodução. O foco desse *framework* durante o seu desenvolvimento não foi a precisão da espacialização e nem a qualidade sonora dos sons espacializados. As falhas de reprodução e imprecisões da localização relatadas se deram por, perda de pacotes durante a transmissão de informações pela rede e limitações hardware dos dispositivos utilizados pelos usuários durante o teste.

Os resultados dos testes do terceiro protótipo descritos na Subseção 3.3.4, demonstraram que a espacialização sonora se mostrou satisfatória ao utilizar um “joystick” para simular o movimento virtual de uma fonte sonora. Através deste teste, concluiu-se que o *framework* proposto é promissor para a utilização em aplicações musicais e que pode ser uma ferramenta interessante e de baixo custo para compositores, produtores musicais e artistas que pretendem fazer experimentos utilizando a espacialização sonora. A facilidade do uso do *framework* corrobora com esta conclusão.

A partir do desfecho do trabalho elaborado, como sugestões de trabalhos futuros e possíveis extensões do sistema, são descritos a seguir:

1. Melhoramento da verificação de redundância de informação recebida;
2. Melhoramentos no projeto da interface gráfica para uma melhor utilização;
3. Desenvolvimento de novos módulos para o sistema de espacialização sonora, a fim de proporcionar uma maior precisão na localização da fonte sonora;
4. Estudos com configurações tridimensionais de alto-falantes, como o modelo Octogonal Cúbico, já suportado pelo *framework* módulo de espacialização;
5. Transformar os módulos e componentes do ARSUS em *externals* (funções) para o ambiente do Pure Data;
6. Este *Framework* poderia ser portado para uma linguagem estruturada, como a Super Collider,¹ para se transformar em um sistema *standalone*, sem a necessidade de rodar

¹ Disponível em: <<http://supercollider.github.io/>>

sobre o Pure Data, e poder ser embutido em softwares específicos, jogos *indies*, aplicações de realidade virtual, etc.

Referências

- ASKA, A.; RITTER, M. Approaches to Real Time Ambisonic Spatialization and Sound Diffusion using Motion Capture. *International Computer Music Conference*, p. 327–332, 2016. Citado na página 37.
- BLAINE, T.; FELLS, S. Contexts of collaborative musical experiences. *Proceedings of the 2003 International Conference for New Instruments for Musical Expression (NIME-2003)*, p. 129–134, 2003. ISSN 0929-8215. Disponível em: <<https://www.etc.cmu.edu/projects/jamodrum/spring04/Nime03BF.pdf>>. Citado na página 14.
- BLAUERT, J. *Spatial Hearing ? The Psychophysics of Human Sound Localization*. 1996. MIT Press, 1996. Citado na página 19.
- BONINI, L. A.; ENDO, G. D. B. Design thinking : uma nova abordagem para inovação. *Biblioteca TerraForum Consultores*, p. 1–7, 1995. Disponível em: <<http://biblioteca.terraforum.com.br/Paginas/designthinking.aspx>>. Citado na página 17.
- BROECKMANN, A. Imagem, processo, performance, máquina: aspectos de estética do maquínico. *Arte, ciência e tecnologia: passado, presente e desafios*. Fundação Editora da Unesp, São Paulo, 2009. Citado na página 14.
- CARPENTIER, T. Récents développements du Spatialisateur. *Actes des Journées d'Informatique Musicale*, p. 4–11, 2015. Citado na página 37.
- CAVALCANTI, C. C.; FILATRO, A. *Design Thinking. Na Educação Presencial, a Distância e Corporativa*. [S.l.]: Editora Saraiva, 2017. Citado na página 42.
- COOK, P. Principles for designing computer music controllers. *n Proceedings of the 2001 conference on New interfaces for musical expression*, 2001. Citado na página 34.
- COOPER, R.; JUNGINGER, S.; LOCKWOOD, T. Design thinking and design management: A research and practice perspective. *Design Management Review*, Wiley Online Library, v. 20, n. 2, p. 46–55, 2009. Citado na página 42.
- COSTA, N. J. B. *O Surround e a Espacialidade Sonora no Cinema*. Dissertação (Mestrado) — Escola de Belas Artes - UFMG, 2013. Citado na página 15.
- FARIA, R.; ALVES, R. *Auralização em ambientes audiovisuais imersivos*. Tese (Doutorado) — Universidade de São Paulo, 2005. Citado 6 vezes nas páginas 11, 22, 23, 25, 27 e 30.
- FARIA, R. R. A. AUDIENCE for Pd, a scene-oriented library for spatial audio. *Proceedings of the Pure Data Convention, Berlin*, 2011. Citado na página 17.
- FARIA, R. R. A. et al. Audience-audio immersion experiences in the caverna digital. In: *Proceedings of the 10th Brazilian Symposium on Computer Music, Belo Horizonte*. [S.l.: s.n.], 2005. p. 106–117. Citado 3 vezes nas páginas 13, 26 e 30.

- FRANINOVIĆ, K.; SERAFIN, S. *Sonic interaction design*. [S.l.]: Mit Press, 2013. Citado na página 34.
- FRITSCH, E. F. *Música Eletrônica-Uma Introdução Ilustrada*. [S.l.]: Eloy Fernando Fritsch, 2008. Citado na página 20.
- FUKS, H. et al. *The 3C Collaboration Model*. Ned Kock (org): The Encyclopedia of E-Collaboration, 2007. 637–644 p. Citado 2 vezes nas páginas 53 e 78.
- FUSHIMI, H. et al. *atmoSphere: Designing Cross-Modal Music Experiences Using Spatial Audio with Haptic Feedback*. p. 2016–2017, 2017. Citado 3 vezes nas páginas 11, 35 e 36.
- GARCIA, D. H. L. Modelos perceptivos na música eletroacústica. *São Paulo, PUC [Doutorado em Comunicação e Semiótica]*, 1998. Citado na página 21.
- GARCIA, J.; FAVORY, X.; BRESSON, J. Trajectoires : a Mobile Application for Controlling Sound Spatialization. *CHI '16 Extended Abstracts on Human Factors in Computing Systems*, p. 4 pages, 2016. Citado 2 vezes nas páginas 11 e 37.
- GERZON, M. *Surround-sound psychoacoustics*. 1974. *Wireless World*, p. 483-485, dez. 1974b. Citado 2 vezes nas páginas 19 e 25.
- GIOVANELLA, T. Levantamento de Requisitos. *Slideshare.Net*, p. 1–33, 2013. Disponível em: <<http://www.slideshare.net/tgiovanela1/aula-1-levantamento-de-requisitos>>. Citado na página 61.
- HERMANN, T.; HUNT, A.; NEUHOFF, J. G. *The sonification handbook*. [S.l.]: Logos Verlag Berlin, 2011. Citado na página 35.
- IAZZETTA, F. *Áudio Digital*. 2017. <http://www2.eca.usp.br/prof/iazzetta/tutor/audio/a_digital/a_digital.html>. [Online; acessado em 29-Março-2017]. Citado 3 vezes nas páginas 12, 70 e 71.
- IDEO. *Design Kit: The Human-Centered Design Toolkit*. 2017. Disponível em: <<https://www.ideo.com/post/design-kit>>. Citado na página 42.
- IGN US. *F.E.A.R Review*. 2006. <<http://www.ign.com/articles/2006/10/25/fear-review-2?page=1>>. [Online; acessado em 29-Março-2017]. Citado na página 15.
- JENSENIUS, A. R.; VOLDSUND, A. The Music Ball Project : Concept , Design , Development , Performance. *Proceedings of the International Conference on New Interfaces for Musical Expression*, n. Figure 2, p. 300–303, 2012. Citado 3 vezes nas páginas 11, 14 e 38.
- LECOMTE, P.; GAUTHIER, P.-A. Real-Time 3D Ambisonics using Faust, Processing, Pure Data, And OSC. *15th International Conference on Digital Audio Effects (DAFx-15), Trondheim*, 2015. Citado na página 56.
- MJV. *Design Thinking*. 2017. [Http://blog.mjv.com.br/ideias/3-fases-design-thinking-imersao-ideacao-e-prototipagem](http://blog.mjv.com.br/ideias/3-fases-design-thinking-imersao-ideacao-e-prototipagem). Citado na página 42.
- MOORE, B. *An Introduction to the Psychology of Hearing*. 5 ed. 2003. Academic Press. Citado 2 vezes nas páginas 15 e 19.

- NAS, E.; LOPES, J. Design De Interfaces Para Experiências Lúdicas Com Novas Tecnologias: O Caso Do Software-Livre Reactivision Em Interações Sonoras. *12º Congresso Brasileiro de Pesquisa e Desenvolvimento em Design*, 2016. Citado 3 vezes nas páginas 11, 14 e 36.
- OSC. *Open Sound Control*. 2016. <<http://opensoundcontrol.org>>. [Online; acessado 10-Julho-2016]. Citado na página 37.
- OSC. *Open Sound Control Specifications*. 2017. <http://opensoundcontrol.org/spec-1_0>. [Online; acessado 10-Outubro-2017]. Citado na página 63.
- PLATTNER, H. *Bootcamp Bootleg*. 2017. Disponível em: <[http://www.ieo.nctu.edu.tw/~ieofuture/106/big_meeting/BootcampBootleg2010v2SLIM\(1\).pdf](http://www.ieo.nctu.edu.tw/~ieofuture/106/big_meeting/BootcampBootleg2010v2SLIM(1).pdf)>. Citado na página 42.
- PLATTNER, H.; MEINEL, C.; LEIFER, L. *Design thinking: understand–improve–apply*. [S.l.]: Springer Science & Business Media, 2010. Citado na página 42.
- PUCKETTE, M. *PureData*. 2016. <<http://www.puredata.info>>. [Online; acessado 01-Julho-2016]. Citado 2 vezes nas páginas 17 e 29.
- RECOMMENDATION, I. 775-1, multichannel stereophonic sound system with and without accompanying picture. *International Telecommunication Union, Geneva, Switzerland*, 1994. Citado na página 25.
- Road to VR. *Infinadeck is a 1000 Pound Omnidirectional Treadmill Designed and Built By One Man*. 2012. <<http://www.roadtovr.com/infinadeck-omnidirectional-vr-treadmill/>>. [Online; acessado em 29-Março-2017]. Citado na página 15.
- ROCCHESSE, D. et al. Sonic interaction design: Sound, information and experience. In: *CHI '08 Extended Abstracts on Human Factors in Computing Systems*. New York, NY, USA: ACM, 2008. (CHI EA '08), p. 3969–3972. ISBN 978-1-60558-012-8. Disponível em: <<http://doi.acm.org/10.1145/1358628.1358969>>. Citado na página 34.
- ROCHA, F. Questões de performance em obras eletrônicas mistas. *XX Congresso da ANPPOM*, Goiânia: ANPPOM,, 2010. Citado na página 14.
- RUMSEY, F.; MCCORMICK, T. *Sound and Recording: an introduction*. 6a edição. ed. [S.l.]: Taylor and Francis, 2012. ISBN 9781136121739. Citado na página 24.
- SANTOS, F. *Auralização binaural com HRTF e decodificação de Ambisonics*. Dissertação (Mestrado) — Universidade de Aveiro, 2012. Citado na página 26.
- SAVIOJA, L. et al. Introduction to the Issue on Spatial Audio. *IEEE Journal of Selected Topics in Signal Processing*, v. 9, n. 5, p. 767–769, 2015. ISSN 1932-4553. Citado na página 23.
- SERRA, X.; WIDMER, G.; LEMAN, M. A roadmap for sound and music computing. *The S2S Consortium*, 2007. Citado na página 34.
- SHULL, J. C. F.; TRAVASSOS., G. An empirical methodology for introducing software processes. *CM SIGSOFT Software Engineering Notes*, 2000. Citado na página 51.

SOARES, A. *ÁSom e características do som: Frequência, Amplitude e Timbre*. 2017. <<https://anasoares1.wordpress.com/2011/01/31/som-e-caracteristicas-do-som-frequencia-amplitude-e-timbre/>>. [Online; acessado em 29-Março-2017]. Citado na página 70.

SuperCollider. *SuperCollider*. 2017. Disponível em: <<http://supercollider.github.io/>>. Citado na página 31.

THOMAZ, L. F. *Aplicação à música de um sistema de espacialização sonora baseado em Ambisonics*. Dissertação (Mestrado) — Universidade de São Paulo, 2007. Citado 5 vezes nas páginas 20, 21, 22, 27 e 39.

TUIO. *TUIO*. 2016. <<http://www.tuio.org>>. [Online; acessado 10-Setembro-2016]. Citado na página 46.

Vue VR. *VUE VR – Entering Into Virtual Reality World*. 2017. <<http://www.vuetechnologie.com/product/vue-vr-treadmill/>>. [Online; acessado em 29-Março-2017]. Citado na página 15.

WIKIPÉDIA. *Quadrifônico*. 2017. [Online; acessado 03-Novembro-2017]. Disponível em: <<https://pt.wikipedia.org/wiki/Quadrifônico>>. Citado na página 24.

ZAWACKI, L. S. *ChucK*. 2017. Disponível em: <<https://blog.lfzawacki.com/chuck-introducao-hello-world/>>. Citado na página 32.

Apêndices

APÊNDICE A – Questionário de experiência de uso do Compomus



UNIVERSIDADE FEDERAL DO AMAZONAS

Instituto de Computação

Programa de Pós-Graduação em Informática

Questionário de Experiência do Usuário Compomus

O projeto de interação sônica é o estudo e a exploração do som como um dos principais canais que transmitem informação, significado e qualidades estéticas/emocionais em contextos interativos.

Este campo de estudos desenvolve objetos sonoros com os quais as pessoas interagem, e tais interações são facilitadas por meios computacionais, no projeto de interação sônica, o som está mediando a interação como exibição de processos ou como meio de entrada.

1. Com base na leitura acima, você sente houve uma interação com o som?, e que isso foi proporcionado pela computação?

Não

Sim

Gostaria de fazer algum comentário? _____

2. Em sua opinião, há colaboração neste projeto?

Não

Sim

Gostaria de fazer algum comentário? _____

3. Você achou a aplicação fácil de usar?

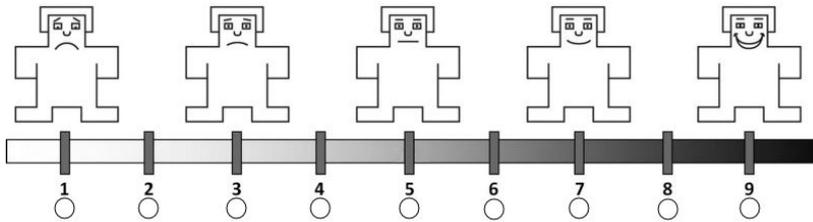
Sim

Não

Gostaria de fazer algum comentário? _____

4. Descreva sua opinião sobre a aplicação

5. Na escala abaixo, qual o grau de usabilidade do Compomus?



6. Do ponto de vista da arte, você se sentiu colaborando para uma performance sonora?

Sim

Não

Gostaria de fazer algum comentário? _____

7. Ficou claro a proposta apresentada pelo projeto?

Sim

Não

Gostaria de fazer algum comentário? _____

8. O feedback de status dado pela aplicação foi suficiente para você saber se estava dentro do alcance?

Sim

Não

Gostaria de fazer algum comentário? _____

APÊNDICE B – Questionário de experiência de uso do Compomus e ARSUS Framework



UNIVERSIDADE FEDERAL DO AMAZONAS

Instituto de Computação

Programa de Pós-Graduação em Informática

Questionário de Experiência do Usuário ARSUS/Compomus

1. Você teve algum problema ao usar o app?

Não

Sim

Qual? _____

2. Você teve alguma dificuldade com a interface?

Não

Sim

Qual? _____

3. Você conseguiu direcionar o som escolhido nas caixas adequadamente?

Sim

Não

Gostaria de fazer algum
comentário? _____

4. Percebeu uma mistura de sons? Se percebeu, ainda era possível identificar o seu?

Sim

Não

Gostaria de fazer algum
comentário? _____

5. Você conseguiu identificar a origem de outros sons?

Sim

Não

Gostaria de fazer algum
comentário? _____

6. A mistura dos sons fez você interagir com outros sons?

Sim

Não

Gostaria de fazer algum comentário? _____

7. A mistura de sons afetou a sua percepção de sons, transformando a maneira como você poderia interagir?

Sim

Não

Gostaria de fazer algum comentário? _____

8. Você conseguiu identificar a movimentação dos outros sons?

Sim

Não

Gostaria de fazer algum comentário? _____

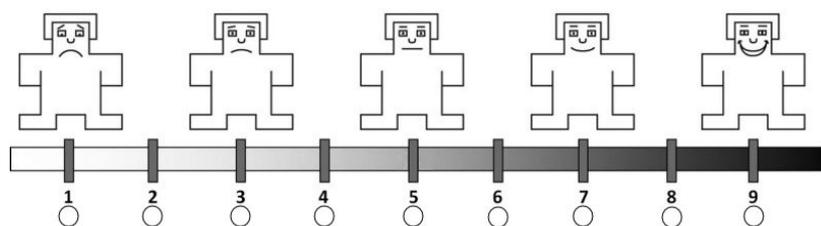
9. Você se sentiu no controle do som escolhido?

Sim

Não

Gostaria de fazer algum comentário? _____

10. Na escala abaixo responda: Quão agradável foi a sua experiência neste estudo?



11. Você sentiu que o seu som contribuiu como parte de uma composição coletiva? Se sim, qual sentimento melhor explicaria essa sensação?

