Simulação acústica no ambiente AcMus

Mário Henrique Cruz Tôrres

DISSERTAÇÃO APRESENTADA
AO
INSTITUTO DE MATEMÁTICA E ESTATÍSTICA
DA
UNIVERSIDADE DE SÃO PAULO
PARA
OBTENÇÃO DO TÍTULO DE MESTRE
EM
CIÊNCIAS

Área de Concentração: Ciência da Computação
Orientador: Prof. Dr. Fabio Kon

São Paulo, 08 de Outubro de 2008

Simulação acústica no ambiente AcMus

Este exemplar corresponde à redação
final da dissertação devidamente corrigida
e defendida por Mário Henrique Cruz Torres
e aprovada pela Comissão Julgadora.

São Paulo, 08 de Outubro de 2008

Banca Examinadora:

- Prof. Dr. Fabio Kon (orientador) - IME-USP.
- Prof. Dr. Marcelo Gomes de Queiroz - IME-USP
- Prof. Dr. Márcio Henrique de Avelar Gomes CEFET - SC

A Daniele e às pequenas
Isabela e Cecília, por existirem. A minha mãe, Maria Emília, por sempre me apoiar. Ao Fabio Pinto,
meu infinito amigo. Aos meus irmãos Dalmo e Daniel, por fazerem parte de mim.

Agradecimentos

Para poder conduzir esse trabalho, apoiei-me no trabalho de muitas gentes, muitas pessoas que quase anônimas, contribuíram comigo, mesmo que muitas vezes sem ter plena consciência disso, para que eu pudesse sentar por longas horas na frente do computador e fazer meu trabalho.

Obrigado aos motoristas da LinhaUSP, Marçola, Romildo, aos funcionários da Cantina do Sandrão, às coopeiras sempre dispostas a nos servir deliciosos cafezinhos, até as 15:30.

Eu sempre tive o apoio de minha família, que me incentivou a continuar caminhando. Amanda, Maria Luiza, Dalmo, Erika, Daniel, Sheila, Pai, Mãe, Dona Ismênia, Seu Roberto, Clóvis. Obrigado por todos momentos que passamos juntos.

Aos meu companheiros de jornada, Vinícius, Pablo, Wellington, Jessé pelas inúmeras horas de bate papo nas filas do bandejão.

Agradeço especialmente ao Márcio Henrique de Avelar Gomes, que sempre se mostrou disposto a me ajudar, mesmo com as dificuldades impostas pela distância. Agradeço também ao professor Fernando Iazzetta e ao pessoal do projeto AcMus, pelo ambiente, estrutura e conhecimento a mim oferecidos.

Por fim, quero agradecer aos professores com os quais tive contato no IME, Imre Simon, Cristina Gomes Fernandes, Carlos Humes Jr., Orlando Stanley Jurians, Marcelo Gomes de Queiroz e em especial ao Fabio Kon, meu orientador, que me mostrou o quanto eu não sabia, o quanto eu poderia saber e que dedicou parte de sua vida a mim, nas infinitas horas de revisão de meus trabalhos.

Agradeço a Deus, pela minha vida.

Resumo

O estudo da acústica de salas data do início do século XX e, até então, esta atividade era vista quase como uma arte e não como ciência. Foram desenvolvidas teorias, como a acústica geométrica de salas, que tornou o estudo do fenômeno acústico mais facilmente inteligível. Com o advento dos computadores digitais, os pesquisadores da área de acústica, como Asbjørn Krokstad e Manfred Robert Schroeder desenvolveram, em torno de 1960, os primeiros algoritmos para simular o comportamento do campo sonoro dentro de salas. Desde então, vários métodos de simulação acústica foram criados [QIK+08].

Nesta dissertação descrevemos uma implementação funcional do algoritmo de Traçado de Raios na linguagem Java. Explicamos em nosso trabalho os vários métodos de simulação acústica existentes nos dias de hoje e descrevemos os passos necessários para a correta implementação de um algoritmo de traçado de raios.

Nossa implementação foi realizada como um módulo do sistema AcMus, uma ferramenta de software livre que reúne, em um único ambiente integrado, medição, análise e, agora, simulação acústica de salas, tornando-se o único software livre existente a integrar essas três funcionalidades em um único ambiente.

O módulo de simulação foi implementado como um arcabouço orientado a objetos de forma que outros pesquisadores possam estendê-lo incluindo, com pouco esforço, novos algoritmos de simulação e novas funcionalidades.

Palavras-chave: acústica, simulação, acmus, computação musical.

Abstract

Research on Room Acoustics dates back to the beginning of the 20th century; at that time this activity was not seen as a science but almost as an art. Since then, researchers in the field have developed theories, such as Geometric Room Acoustics, that were a simplification of the acoustical phenomena.

With the advent of digital computers, researchers such as Asbjørn Krokstad and Manfred Robert Schroeder have developed, in the 1960s, the first algorithms to simulate the sound field behavior inside rooms. Since then, various acoustic simulation methods have been created.

In this thesis, we describe our working implementation, using the Java language, of the Ray Tracing algorithm. We also explain the various acoustic simulation algorithms used nowadays and describe the necessary steps to a correct ray tracing algorithm implementation.

Our implementation was built as a module of AcMus, a free software system, which combines acoustic measurement, analysis and, now, simulation in one integrated computational environment. The simulation module was built as an object-oriented framework so that other researchers and developers can extend it, including new simulation algorithms and new features with little effort.
Keywords: acoustics, simulation, acmus, computer music.

Sumário

1 Introdução **1**

1.1 Motivação 2

1.2 Objetivos 3

1.3 Organização da Dissertação 3

2 Simulação Acústica **5**

2.1 Modelos em Escala 7

2.2 Acústica Geométrica de Salas 8

2.2.1 Método das Fontes Virtuais 10

2.2.2 Traçado de Raios 12

2.3 Métodos Numéricos Baseados na Equação de Onda 12

2.4 Conclusão 13

3 Simulação Acústica usando a Técnica de Traçado de Raios **15**

3.1 Modelo da Sala Simulada 15

3.2 Modelagem da Fonte Sonora 16

3.2.1 Fonte Sonora Determinística 17

3.2.2 Fonte Sonora Pseudo-Aleatória 18

3.3 Reflexão dos Raios 20

3.3.1 Reflexão Especular 20

3.3.2 Reflexão Difusa 25

3.4 Receptor Sonoro 26

3.4.1 Detecção do Raio pelo Receptor Esférico 27

3.4.2 Cálculo da Energia do Raio Interceptado pelo Receptor Esférico 29

3.5 Resposta Impulsiva 29

3.6 Algoritmo de Traçado de Raios 31

4 Ambiente AcMus **33**
4.1 Arquitetura 34
4.1.1 Módulo de Medição Acústica 35
4.1.2 Módulo de Utilidades 35
4.1.3 Módulo de Simulação Acústica 36
4.2 Arquitetura do Módulo de Simulação Acústica 36
4.2.1 Fonte Sonora 38
4.2.2 Receptor Sonoro 39
4.2.3 Resposta Impulsiva 39
4.2.4 Desempenho 41
4.3 Uso do Módulo de Simulação através da linha de comando 41
4.4 Integração do Módulo de Simulação ao Ambiente AcMus 42
4.5 Resumo 43

5 Avaliação do Módulo de Simulação Acústica **47**
5.1 Ambiente de Simulação 48
5.1.1 Sala Simulada 48
5.1.2 Emissor e Receptor Sonoros 51
5.2 Resultados da Simulação 51
5.3 Resumo 58

6 Conclusão **59**
6.1 Trabalhos Futuros 60

Lista de Figuras

2.1 Projeto do Oden reconstruído [RGN06]. 6
2.2 Fonte onidirecional capaz de reproduzir sons entre 500 Hz a 80 kHz [Inc03] 8
2.3 Modelo em Escala da Sydney Opera House [Rin02]. 9
2.4 Fontes virtuais de primeira ordem em uma sala cubóide. 11
2.5 Visão esquemática de um raio refletido. 12

3.1 Ilustração do auditório da FEC-UNICAMP, com dimensões em metros [GBD07] 16
3.2 Ilustração da subdivisão de uma das faces do icosaedro regular 17
3.3 Subdivisão geodésica do icosaedro regular circunscrito por uma circunferência 18
3.4 Pontos uniformemente distribuídos num cubo unitário. Os pontos do interior da esfera unitária têm cor mais clara, já os pontos rejeitados estão marcados com uma cruz e cor mais escura 19
3.5 1000 pontos distribuídos uniformemente sobre a superfície de uma esfera unitária 19
3.6 Esquema de verificação do plano de incidência do raio, sendo ̣ o ângulo agudo entre $\vec{v}$ e $\vec{n_6}$, nesta figura o raio v não incide no plano 6 21
3.7 Teste de ponto no polígono [Kul85] 22
3.8 Esquema mostrando os vetores usados no cálculo da distância entre p_s e o i-ésimo plano 22
3.9 Reflexão de um raio. Ps e Prs são a fonte sonora e sua imagem; Pi é um ponto qualquer no plano de incidência do raio sonoro; n_i é o vetor normal ao plano de incidência. 24
3.10 Ilustração da distribuição dos raios numa reflexão difusa [TC06] 26
3.11 Esquema para ilustrar se um raio atinge o receptor esférico 28
3.12 Respostas Impulsivas medida e simulada através do método de traçado de raios [GBD07] 30
3.13 Algoritmo de traçado de raios 32

4.1 Arquitetura do ambiente AcMus 34
4.2 Cálculo de Parâmetros Acústicos [UFIK05] 36
4.3 Diagrama de Classes do módulo de simulação acústica 37
4.4 Gráfico com resposta impulsiva, gerada e exibida no módulo de simulação. 40

4.5 Arquivo WAV com uma resposta impulsiva simulada. 40
4.6 Tempo de processamento para simulação em sala cubóide. 42
4.7 Menu de ferramentas do sistema AcMus com a opção para *Ray Tracing Simulation* em destaque 43
4.8 Dinâmica de uma simulação acústica. 44
4.9 Tela do AcMus para simulação acústica de salas cubóides. 45

5.1 Vistas posterior, lateral direita e superior do estúdio de música do PTB [Bun02]. 49
5.2 EDT obtido pelo AcMus marcado com pontos pretos, média e respectivo desvio padrão dos resultados obtidos pelos participantes do RR 3 marcado com a barra vertical. 52
5.3 T30 obtidos pelo AcMus, marcado com pontos pretos, média e respectivo erro dos outros participantes. 53
5.4 Parâmetro D50 calculado pelo AcMus e pelos participantes do RR 3. 54
5.5 EDT calculado pelo AcMus e dos participantes do RR3. 55
5.6 T30 calculado pelo AcMus e pelos participantes do RR 3. 56
5.7 D50 calculado pelo AcMus e pelos participantes do RR 3. 57

Lista de Tabelas

5.1 Descrição dos vértices usados na sala de referência do RR3. 49
5.2 Descrição das paredes da sala de referência, em relação aos vértices dados na Tabela 5.1 50
5.3 Parâmetros acústicos calculados. 50
5.4 S1 e S2 são os emissores sonoros, R01 e R02 são os receptores sonoros. 51
5.5 EDT: fonte S1 e receptor R1. Valores em segundos. 52
5.6 T30: fonte S1 e receptor R1. Valores em segundos. 53
5.7 D50: fonte S1 e receptor R1 54
5.8 EDT: fonte S2 e receptor R2. Valores em segundos 55
5.9 T30: fonte S2 e receptor R2 56
5.10 D50: fonte S2 e receptor R2 57

Capítulo 1

Introdução

Cotidianamente julgamos a qualidade acústica de um ambiente baseando-nos em nossas experiências auditivas passadas. Na tentativa de explicar objetivamente o comportamento acústico de ambientes, foi desenvolvido o estudo da acústica de salas. Assim, criaram-se modelos matemáticos para descrever o comportamento do campo sonoro em ambientes fechados, bem como criaram-se vários parâmetros para descrever o comportamento acústico de salas.

Uma das ferramentas desenvolvidas pelos pesquisadores da área foi a simulação acústica, que é usada para simular o comportamento acústico de uma sala e obter, assim, mais informações sobre tal comportamento, o que é feito calculando diversos parâmetros acústicos da sala simulada. Uma ferramenta que se mostrou muito útil para o projeto acústico de grandes salas foram os modelos físicos em escala. Porém, com o advento dos computadores, seu uso foi superado pela simulação digital. Introduzida por M.R. Schroeder em 1958, a simulação acústica de salas sofreu um grande avanço nos últimos anos, pois o tempo e o custo necessários para fazer uma simulação foram reduzidos drasticamente e, devido a esse avanço, atualmente é possível prever, por exemplo, o comportamento sonoro de salas ainda não construídas, simular a acústica de construções antigas não mais existentes e ainda melhorar a acústica de salas já existentes.

O estudo da acústica de salas data do início do século XX, quando os modelos matemáticos usados para descrever o fenômeno acústico ainda eram poucos e a construção de grandes salas de teatro ou para apresentação musical era tratada quase como uma arte. Nesta dissertação, trataremos da simulação da acústica de salas, que podem ser salas de concerto ainda não construídas, salas de concerto já existentes, ou simplesmente novas salas para prática musical que só existirão num ambiente computacional para platéias sentadas em frente a seus computadores em qualquer parte do mundo.

São muitas as possibilidades trazidas pela simulação acústica e com o uso dessa técnica é possível construir salas com características inimagináveis em salas feitas de tijolo e concreto. Explorar essas novas possibilidades pode ser um trabalho para músicos usuários de nosso programa e para engenheiros acústicos interessados em explorar os limites do que a acústica pode oferecer para nossos cérebros.

Mostraremos as principais técnicas de simulação acústica existentes atualmente e mostraremos também como criamos uma plataforma de software ampliável que permite a incorporação de novas técnicas de simulação ao ambiente AcMus.

Há muitos anos, pesquisadores desenvolvem programas para simulação acústica de salas, mas somente nos últimos anos foi possível perceber um avanço significativo na área. Isto porque o acesso aos computadores pessoais e a capacidade de processamento destes aumentou muito, permitindo que qualquer usuário possa fazer experimentos de simulação acústica [Kut00].

As técnicas de simulação acústica podem ser divididas em dois grandes grupos: os métodos baseados na resolução numérica da equação de onda e os métodos baseados na acústica geométrica de salas. Os métodos baseados na resolução numérica da equação de onda tentam solucionar a equação de onda de forma numérica, discretizando o espaço da sala e resolvendo a equação de onda em cada ponto resultante dessa discretização. Já os métodos baseados na acústica geométrica tratam a onda sonora como uma coleção de raios sonoros; o comportamento desses diversos raios é monitorado ajudando assim na compreensão de como a onda se propaga na certa sala.

Ao realizar uma simulação acústica é possível obter a Resposta Impulsiva (RI) da sala, que é uma resposta da sala a um certo impulso de entrada. Essa Resposta Impulsiva pode ser representada por uma sequência de valores de energia em função do tempo ou pressão sonora em função do tempo, dependendo do algoritmo de simulação usado. De posse da RI é possível calcular diversos parâmetros acústicos, como clareza, tempo inicial de decaimento, dentre outros.

Neste texto, aprofundaremo-nos no método de Traçado de Raios que é baseado na acústica geométrica de salas e mostraremos em detalhes os conceitos matemáticos necessários para desenvolver um algoritmo de traçado de raios, apresentando aspectos da implementação deste algoritmo usando a linguagem Java.

Por fim, mostraremos a arquitetura do nosso programa de simulação acústica de salas, como é feita a integração do módulo de simulação ao ambiente AcMus e como é possível estender esse ambiente, de forma que ele inclua novos algoritmos de simulação acústica.

1.1 Motivação

O principal objetivo ao simular a acústica de um ambiente é predizer como esse ambiente se comportará acusticamente. De posse dos resultados dessa simulação é possível tomar decisões para melhorar a acústica dos lugares. Este trabalho foi motivado pelo interesse em prover ferramentas computacionais que possam auxiliar na construção de salas com melhores características acústicas, pois acreditamos que esses ambientes, quer sejam teatros, salas para prática musical ou auditórios para ministrar aulas, podem ser projetados com uma maior qualidade acústica e, assim, melhorar a experiência das pessoas que usam tais espaços.

A segunda motivação é a inexistência de ferramentas de software livre que façam a simulação acús-

tica de salas. As ferramentas para simulação acústica de salas têm um alto custo, o que impede a adoção de tais programas por um maior número de pessoas e dificulta a realização de pesquisas científicas.

1.2 Objetivos

Nosso objetivo é estudar as técnicas de simulação acústica de salas e implementar um software livre com uma delas. Este software foi integrado ao ambiente AcMus, tornando-o, de acordo com nosso conhecimento, o único ambiente integrado existente capaz de realizar medições e simulações acústicas.

Criamos também um programa com arquitetura simples para que outros pesquisadores possam implementar novos algoritmos de simulação acústica no ambiente AcMus.

1.3 Organização da Dissertação

Trataremos, na parte teórica desta dissertação, isto é, nos Capítulos 1 e 2, da simulação acústica de salas, usando o computador como meio para realizar tais simulações. Mostraremos as principais técnicas computacionais de simulação acústica de salas, sendo elas, métodos baseados na equação de onda, *(Wave-based methods)* e métodos baseados na acústica geométrica de salas, *(Geometrical acoustic methods)*. Aprofundaremo-nos num método derivado da acústica geométrica de salas chamado Traçado de Raios, apresentando os conceitos matemáticos necessários para desenvolver tal método.

Já a segunda parte do trabalho, Capítulos 3 e 4, diz respeito à implementação do módulo de simulação no ambiente AcMus. Mostraremos a arquitetura do módulo de simulação, como é feita a integração das ferramentas de simulação dentro do ambiente AcMus e também como outros métodos de simulação poderão ser implementados nesse ambiente. Mostraremos também os resultados obtidos pelo algoritmo de simulação implementado no AcMus em relação a outros programas para simulação acústica.

Por fim, no Capítulo 5, apresentaremos nossas conclusões e possíveis trabalhos futuros.

Capítulo 2

Simulação Acústica

Computadores são ótimas ferramentas para simular processos do mundo em que vivemos; segundo Law e Kelton [LK97], para estudar cientificamente um processo é necessário fazer certas considerações e simplificações a respeito do funcionamento de tais processos. Essas considerações são feitas na forma de relações lógicas e matemáticas e constituem o *modelo* do processo. Um exemplo de modelo é a acústica geométrica de salas, na qual o conceito de uma onda sonora é substituído pelo conceito de raio sonoro.

Dependendo da simplicidade do modelo, podemos usar ferramentas matemáticas como cálculo, álgebra ou probabilidade para obter uma resposta exata às questões feitas ao sistema, como, por exemplo, o tempo de reverberação do som numa sala. Essa resposta exata é chamada solução analítica do modelo. Porém, é muito raro encontrar sistemas no mundo real que contem com modelos simples o suficiente para serem resolvidos de forma analítica.

Para processos mais complexos, a simulação é uma poderosa ferramenta que pode ajudar a obter mais informações a respeito de tais processos, como ocorre em simulação acústica. Neste tipo de simulação, é possível prever a acústica de salas extremamente complexas, como conduzido no âmbito do projeto ERATO *(identification Evaluation and Revival of the Acoustical heritage of ancient Theatres and Odea)*, em que foi feita a reconstrução da acústica de uma sala de concerto romana *(Odeon)*, usando dados de sítios arqueológicos e estudos sobre a civilização romana [RGN06]. Um modelo da sala usada nessa simulação pode ser visto na Figura 2.1.

Existem modelos relativamente simples como as equações de Sabine e de Eyring [Ber06], as quais nos dão o tempo de decaimento do som numa certa sala. Para a obtenção de outros parâmetros acústicos como, por exemplo, clareza e definição, a simulação computacional se mostra necessária.

Há várias técnicas para a modelagem e simulação acústica, como o traçado de raios, método das fontes virtuais, método de elementos finitos para solucionar a equação de onda, porém nenhuma delas consegue resolver todos problemas existentes ao tentar prever o comportamento acústico de uma sala [Elo05]. Por isso, é importante estudarmos os limites de cada técnica para saber quando podemos usá-la ou não. A seguir, faremos uma breve explicação sobre algumas técnicas de simulação acústica.

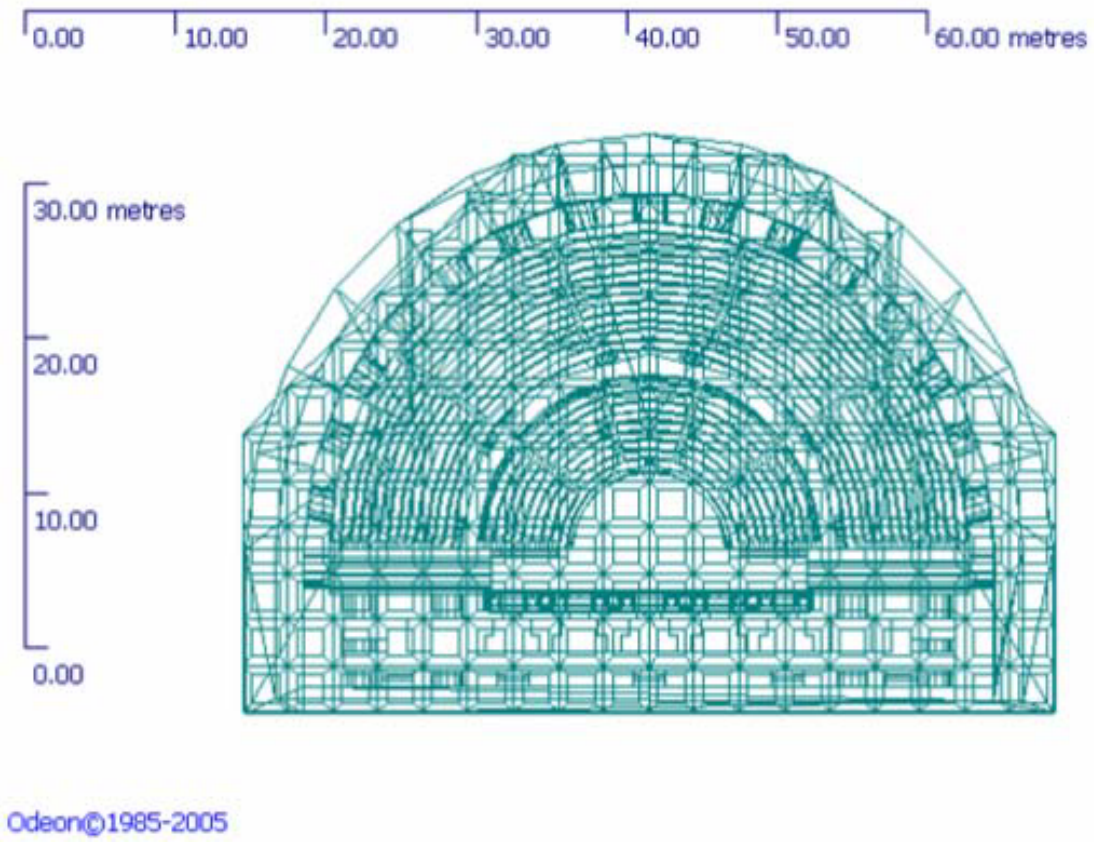

Figura 2.1: Projeto do Oden reconstruído [RGN06].

2.1 Modelos em Escala

Segundo Kuttruff [Kut00], um método muito bem testado e conhecido, usado para estudar o comportamento acústico de grandes salas e auxiliar em seu projeto, consiste em construir um modelo em escala de uma determinada sala, tentando reproduzir sua geometria e os materiais com os quais ela será construída. É possível, inclusive, experimentar vários materiais para as paredes e até mesmo mudar o formato da sala. Para o autor, esse método tem a vantagem de possibilitar, com pouco trabalho, a construção de um grande número de variações em relação ao projeto da sala. Segundo Nagata [Inc03], outra grande vantagem dos modelos em escala está na reprodução fidedigna dos efeitos acústicos devido à difração da onda sonora.

De acordo com Kuttruff [Kut00], já que várias propriedades de propagação são comuns a vários tipos de ondas, não é absolutamente necessário usar ondas sonoras para fazer medições no modelo em escala. Este fato foi muito importante para o início do uso desses modelos, quando as técnicas de medição acústica não tinham atingido o avançado estágio no qual estão hoje. Assim, eram usadas ondas de água, à vezes aplicadas em "tanques d'água", de forma que era fácil observar essas ondas. Porém, o uso em tais tanques se restringia ao estudo das seções planas das salas.

Outra alternativa era o uso de luz para substituir o som. Neste caso, as paredes absorventes eram pintadas de preto ou cobertas com papel preto, enquanto as áreas refletoras eram feitas de metal polido. Áreas que causariam reflexões difusas também podiam ser simuladas, usando-se um pedaço de papel branco. Com isso, a detecção da distribuição da energia podia ser feita através de células fotoelétricas ou fotografias. No entanto, havia problemas relativos à alta velocidade da luz e, além disso, os fenômenos devido à difração eram negligenciados, pois o comprimento de onda da luz é extremamente pequeno quando comparado ao tamanho dos objetos que poderiam causar a difração das ondas sonoras no modelo em escala.

Atualmente, as técnicas de transdutores eletroacústicos alcançaram um nível de excelência que permitem o uso de ondas sonoras para a investigação da propagação do som nos modelos em escala. Um exemplo de tal avanço são os equipamentos usados por Nagata que usa uma fonte sonora feita na forma de um dodecaedro - de apenas 50 mm de diâmetro - com um *tweeter* em cada um de seus vértices. Tal equipamento, mostrado na Figura 2.2, consegue reproduzir frequências que vão de 500 Hz a 80 kHz, com características quase onidirecionais para frequências menores que 10 kHz [Inc03].

Para se obter a resposta impulsiva nos modelos em escala é preciso primeiro posicionar os microfones e depois realizar diversas medições de um sinal sonoro. Tendo em mente o bom desempenho dos equipamentos de Nagata [Inc03], podemos dizer que a simulação acústica usando modelos em escala é uma técnica interessante devido a seus resultados, porém financeiramente custosa, para a simulação acústica de salas. A Figura 2.3 ilustra o modelo em escala da sala principal usado na construção da *Sydney Opera House*.

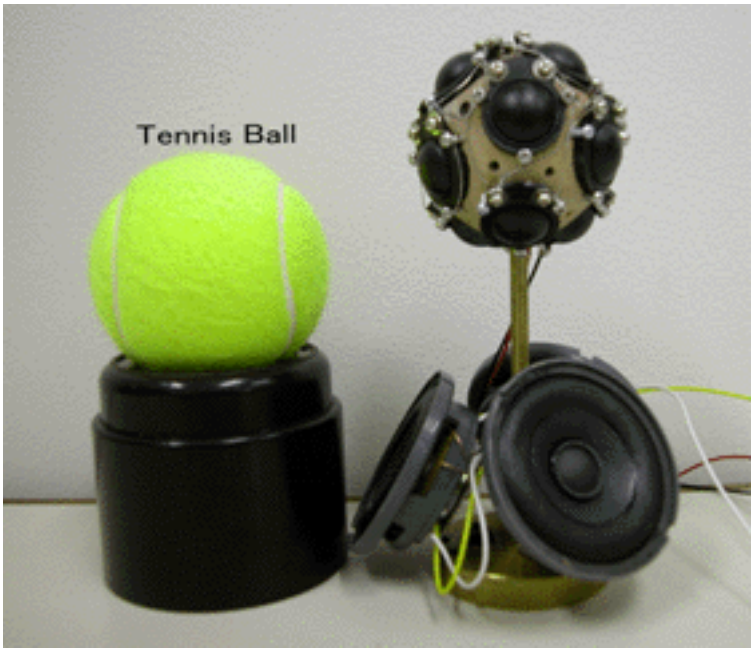

Figura 2.2: Fonte onidirecional capaz de reproduzir sons entre 500 Hz a 80 kHz [Inc03]

2.2 Acústica Geométrica de Salas

De acordo com Kuttruff, ao se estudar a acústica de salas usando técnicas geométricas, o conceito de onda sonora é substituído pelo conceito de raio sonoro. Tratar, pois, a onda sonora de uma maneira geométrica é muito útil, devido à facilidade em se trabalhar com o conceito de raio sonoro. Em outras palavras, os raios sonoros seguem as regras da ótica geométrica, o que é uma grande simplificação do fenômeno acústico. O uso de tal simplificação é possível no caso limite em que o comprimento de onda é relativamente pequeno, o que ocorre para altas frequências. Para se ter uma idéia, o comprimento de onda de uma onda sonora com frequência média de 1000 Hz é de aproximadamente 34 centímetros, ou seja, é algumas ordens de grandeza menor do que o comprimento de uma parede ou a distância percorrida por uma onda sonora numa sala de teatro.

Em acústica geométrica, o raio sonoro tem origem em um certo ponto e é uma pequena porção de uma frente de onda esférica. Ele possui uma direção bem definida e sua propagação é regida pelas mesmas leis de propagação de um raio de luz, porém com a velocidade do som.

Um ponto importante que o autor ressalta é que um conjunto divergente de raios segue a lei do inverso do quadrado, $1/r^2$, como em toda onda esférica, onde r é a distância em relação à origem, percorrida pelo raio. Esse efeito é obtido pela construção da fonte sonora modelada e assim não precisa ser explicitamente modelado no algoritmo de simulação. Outro ponto relevante em simulação acústica é a lei da reflexão. Ao contrário da ótica geométrica, em acústica de salas não é considerada a mudança de

Figura 2.3: Modelo em Escala da Sydney Opera House [Rin02].

meio nem a refração decorrente de tal mudança. A velocidade de propagação do som é responsável por muitos efeitos importantes como reverberação, ecos etc. [Kut00]. Ainda segundo o autor, o fenômeno da difração é negligenciado, pois os raios sonoros são propagados em linhas retas. A interferência também não é diretamente considerada, isto é, se os componentes de vários campos sonoros são sobrepostos, suas relações de fase simplesmente não são modeladas explicitamente, ao invés disso, somente suas densidades de energia e intensidades são adicionadas.

Os dois principais métodos de simulação acústica derivados da teoria da acústica geométrica de salas são:

- Fontes Virtuais *(Image Source)*
- Traçado de Raios *(Ray-Tracing)*

Esses métodos são explicados nas próximas seções de nosso trabalho.

2.2.1 Método das Fontes Virtuais

Na sua forma original, o método das fontes virtuais modelava as salas como recintos retangulares e os objetivos iniciais de seus autores, Allen e Berkley [AB79], eram: estudar a acústica de escritórios que geralmente têm geometria retangular; facilitar a implementação em um programa de computador e convergir rapidamente o resultado de uma simulação para a solução da equação de onda, caso as paredes de uma determinada sala fossem consideradas rígidas. O resultado de uma simulação usando esse método é uma resposta impulsiva, ou seja, uma resposta da sala a um impulso inicial.

O Método das Fontes Virtuais é baseado no princípio de que uma reflexão especular pode ser construída geometricamente ao espelhar a fonte sonora no plano da superfície de reflexão [Rin00]. Ao se obter esta imagem refletida da fonte sonora, o raio original, que sofreria uma reflexão, é substituído por um raio direto que parte dessa imagem refletida da fonte para o receptor [Sav99]. Assim, o princípio básico do modelo das fontes virtuais é que o caminho seguido por um raio em suas várias reflexões, pode ser substituído por uma linha reta que liga o receptor a uma certa fonte virtual.

Às primeiras imagens refletidas dá-se o nome de fontes virtuais de primeira ordem. Já às imagens obtidas pela reflexão de fontes virtuais de primeira ordem dá-se o nome de fontes virtuais de segunda ordem e assim sucessivamente. A Figura 2.4 ilustra a representação de duas fontes virtuais de primeira ordem.

Uma das grandes vantagens do método das fontes virtuais é a possibilidade de garantir que todas as imagens virtuais, a uma certa distância, que possam contribuir com alguma energia para o receptor sonoro, são encontradas [Bor84]. Tal característica permite construir auralizações fidedignas da sala simulada, pois todas reflexões iniciais são encontradas e computadas.

Se a sala simulada não tiver uma geometria simples, uma sala cubóide, por exemplo, tem geometria simples, a quantidade de fontes virtuais pode crescer muito, tornando a simulação acústica computacio-

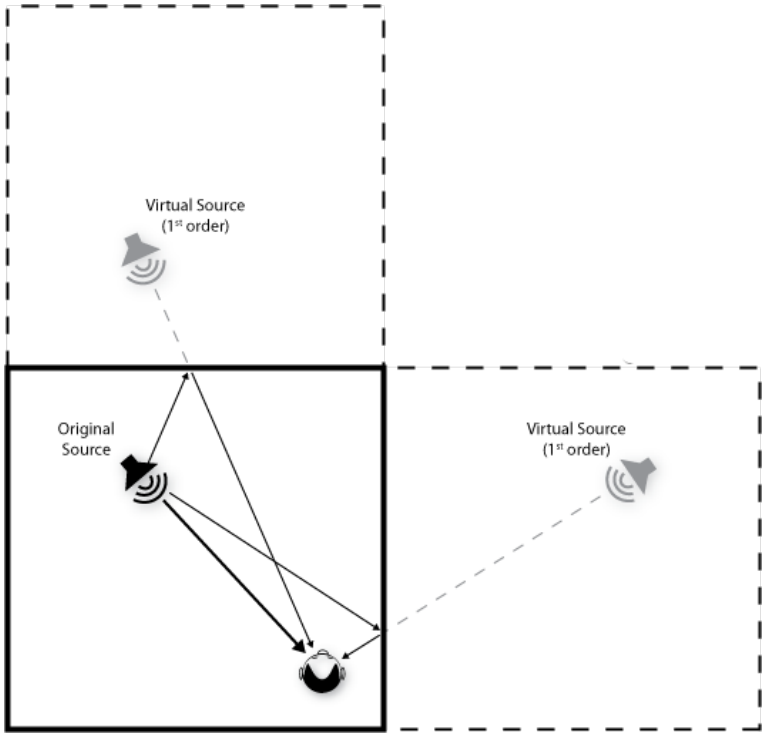

Figura 2.4: Fontes virtuais de primeira ordem em uma sala cubóide.

nalmente custosa. Numa sala com *n* paredes, poderão existir *n* fontes virtuais de primeira ordem e cada uma delas poderá dar origem a *n-1* fontes virtuais de segunda ordem, o que leva a

$$
\begin{aligned}
N_{fontes} &= \sum_{j=1}^{i} n\,(n-1)^{j-1} \\
&= n \sum_{j=1}^{i} (n-1)^{j-1} \\
&= \frac{n}{(n-2)} \left[(n-1)^{i} - 1 \right] \\
&\approx (n-1)^{i}
\end{aligned}
\tag{2.1}
$$

onde N_{fontes} é o número de possíveis fontes e *i* é a ordem de reflexão de uma fonte. É fácil ver, a partir da Equação (2.1) que o número de fontes virtuais cresce exponencialmente com a ordem das fontes, podendo tornar a simulação intratável, para um *i* grande [Rin00].

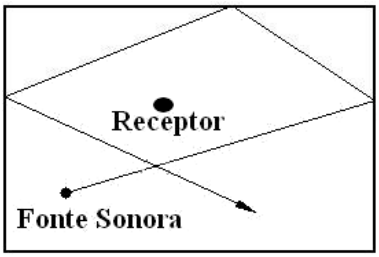

Figura 2.5: Visão esquemática de um raio refletido.

Borish [Bor84] criou uma extensão para o método das fontes virtuais funcionar para salas com geometria complexa e a única limitação é o tempo computacional necessário para realizar a simulação. O autor aponta também que o número de lados da sala simulada não é o único fator a influenciar o tempo de computação. Mudar a forma da sala, mesmo que mantendo o número de superfícies, pode afetar a quantidade de processamento de uma maneira imprevisível.

2.2.2 Traçado de Raios

A técnica de Traçado de Raios é usada no ambiente AcMus para fazer a simulação acústica de salas. Nesta técnica, descrita pela primeira vez por Krokstrad e Strom, em 1967 [Rin00], vários raios são criados numa fonte sonora e, ao longo do caminho percorrido, sofrem reflexões ao encontrar obstáculos. Assim, os raios são rastreados enquanto não atingem um receptor ou enquanto não satisfazem a um critério de parada estabelecido como, por exemplo, o número de reflexões sofridas por ele.

Os raios sonoros têm uma direção e energia e ao longo da simulação, estes raios são refletidos nas paredes da sala, como exemplificado na Figura 2.5. A cada reflexão é verificado se eles interceptam o receptor sonoro e, quando ocorre tal colisão, a energia e o tempo decorridos para o raio chegar até o receptor são anotados, a fim de podermos calcular, em seguida, a Resposta Impulsiva da sala [Kul85].

Explicaremos os vários passos necessários para construir um algoritmo para simulação acústica usando da técnica de traçado de raios no Capítulo 3.

2.3 Métodos Numéricos Baseados na Equação de Onda

Outra abordagem usada para simular a acústica de salas são os métodos numéricos baseados na resolução numérica da equação de onda. Ao estudar fisicamente o fenômeno acústico, é possível conectar variáveis como pressão e temperatura do ar e chegar a um modelo matemático que reflete muito bem o comportamento ondulatório do som. Tal modelo é expresso através da equação diferencial:

$$c^2 \Delta p = \frac{\partial^2 p}{\partial t^2} \tag{2.2}$$

onde

$$c^2 = \kappa \frac{p_o}{\rho_0}$$

p é a pressão sonora, t é o tempo, ρ_0 é o valor estático da densidade do ar e κ é o expoente de pressão adiabática - para o ar o valor de κ é $1,4$. A propagação das ondas sonoras em um meio viscoso sem perdas de energia é regida pela equação (2.2), que é denominada equação de onda [Kut00].

Tendo em mente a afirmação de Raghuvanshi [RGL08] de que os métodos numéricos baseados na equação de onda conseguem capturar todos os efeitos decorrentes da natureza ondulatória do som, acreditamos que esse é o principal benefício de tais métodos. Os principais métodos baseados na resolução numérica da equação de onda são:

- Método de Elementos Finitos *(Finite Element Method - FEM)*
- Método dos Elementos de Fronteira *(Boundary Element Method - BEM)*
- Método da Malha Digital *(Digital Waveguide Mesh - DWM)*
- Método das Diferenças Finitas no Domínio do Tempo *(Finite Difference Time Domain Method - FDTD)*

Segundo Savioja, os métodos numéricos podem ser divididos em duas grandes categorias: métodos que trabalham no domínio da frequência, como o Método de Elementos Finitos (FEM) e o Método dos Elementos de Fronteira (BEM) e métodos que trabalham no domínio do tempo, como o Método da Malha Digital (DWM) e o Método das Diferenças Finitas no Domínio do Tempo [SL01].

Segundo Raghuvanshi, alguns desses métodos são usados principalmente para encontrar o estado estável da onda sonora no ambiente, como o FEM e o BEM. Já os métodos que trabalham no domínio do tempo apresentam uma resposta da sala no tempo. A fim de simular de maneira eficiente a propagação da onda no domínio do tempo, o DWM foi proposto por Duyne, em 1993, e desde então se tornou uma área de pesquisa muito ativa.

Os métodos baseados na resolução numérica da equação de onda encontram vários desafios computacionais para simular frequências médias, acima de 8 Kilohertz, ou para simular a acústica de ambientes grandes. Pois o tempo de processamento cresce super-linearmente, tornando o problema intratável [RGL08], devido à maior granularidade necessária na construção das malhas usadas por estes métodos.

2.4 Conclusão

Há várias formas de modelar a acústica de uma sala. Os métodos baseados na teoria da acústica geométrica têm se mostrado apropriados para predizer os parâmetros acústicos de grandes salas, mesmo

não se tratando de uma teoria na qual o campo sonoro seja exatamente descrito [Vor00].

Já os métodos baseados na resolução numérica da equação de onda trabalham com a representação exata da equação de onda, podendo aproximar-se mais ou menos dela, de acordo com a granularidade da malha usada. O principal problema dos métodos baseados na equação de onda é o tempo computacional necessário para fazer tais simulações.

Capítulo 3

Simulação Acústica usando a Técnica de Traçado de Raios

Em simulação acústica de salas, a técnica de traçado de raios data de meados da década de 1970 [Rin02] e vem sendo aprimorada desde então. Além da aplicação em simulação acústica, essa técnica também é usada para a síntese de imagens, isto é, para a criação de imagens bidimensionais do mundo tridimensional [Gla89]. É importante ressaltar, entretanto, que, a partir de agora, ao nos referirmos ao traçado de raios, estaremos tratando especificamente da técnica de traçado de raios para simulação acústica, escopo de nosso trabalho.

Segundo Vorländer [Vor00], o traçado de raios advém da teoria de acústica geométrica de salas, a mais bem sucedida técnica para a previsão dos parâmetros acústicos do campo sonoro em grandes salas. Investigações sobre a trajetória de raios sonoros viajando pelas salas são a base para a teoria de reverberação estatística e das equações de Sabine e Eyring [Ber06]. É conhecido o fato dessas equações descreverem de forma analítica alguns aspectos do comportamento acústico de salas.

Com o traçado de raios é possível observar as primeiras reflexões dos raios sonoros e, a partir delas, obter parâmetros da acústica do ambiente, tais como tempo inicial de decaimento *(early decay time)* e clareza *(clarity)* [LNY00]. Para desenvolver tal técnica é preciso, inicialmente, definir o modelo da sala a ser simulada.

3.1 Modelo da Sala Simulada

Ao realizar uma simulação acústica estamos interessados, sobretudo, em estudar a acústica de uma certa sala. Para podermos realizar a simulação é preciso, inicialmente, definir como é a representação da sala. A essa representação damos o nome de Modelo da Sala.

Em nosso modelo, representamos a sala através de vários polígonos regulares com quatro lados, quadriláteros regulares. Representamos cada quadrilátero por um vetor normal ao plano ao qual o qua-

drilátero pertence, mais quatro pontos referentes aos seus vértices. Proposta por Kulowski, essa forma de modelar a sala satisfaz às necessidades do algoritmo de traçado de raios [Kul85].

A sala representada por quadriláteros regulares pode apresentar até mesmo paredes curvilíneas, paredes estas modeladas pela junção de vários quadriláteros. Na Figura 3.1 é possível ver a representação em quadriláteros de um auditório.

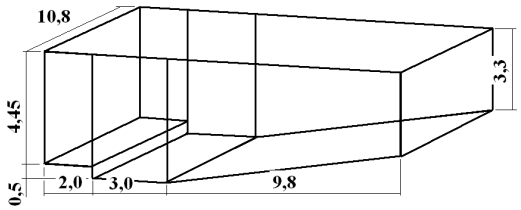

Figura 3.1: Ilustração do auditório da FEC-UNICAMP, com dimensões em metros [GBD07]

De posse de um modelo da sala a ser simulada, é preciso, primeiro, criar os raios sonoros que são emitidos pela fonte sonora, cuja modelagem é descrita na seção a seguir.

3.2 Modelagem da Fonte Sonora

Na técnica de traçado de raios precisamos criar um modelo matemático que represente uma fonte sonora, uma vez que é a partir desta fonte que serão emitidos os raios sonoros usados ao longo da simulação. Esses vários raios distribuídos sobre a superfície da fonte sonora simulada têm uma energia inicial definida como uma fração da energia total da fonte sonora, como nos mostra Gomes [GBD07]:

$$I_{raio} = \frac{I_0}{N_{raios}}$$

onde, I_{raio} é a energia inicial de um raio, I_0 é a energia total inicial da fonte sonora e N_{raios} é o número de raios a serem gerados. Podemos adotar a energia de cada raio como sendo unitária, já que estamos interessados na energia relativa de cada raio e não em seu valor absoluto [GBD07].

Há dois objetivos possíveis ao modelarmos fontes sonoras. Em primeiro lugar, quando o principal objetivo é estudar o comportamento acústico de uma determinada sala e depois comparar seus resultados com os de outras simulações, os raios são gerados de forma a simularem uma fonte sonora onidirecional. No entanto, criar uma fonte sonora simulada na técnica de traçado de raios que seja verdadeiramente onidirecional é impossível [Lew93], devido à natureza contínua da fonte sonora real e à natureza discreta da fonte sonora simulada. Na tentativa de minimizar esse problema, Lewers [Lew93] propõe um modelo determinístico de fonte, o qual, acredita ele, aproxima-se melhor de tal fonte onidirecional.

Em segundo lugar, quando objetivamos estudar os efeitos de uma determinada fonte, ou conjunto de fontes, no ambiente, podemos modelar fontes sonoras direcionais [LNY00]. Este estudo pode, por exemplo, auxiliar-nos a entender o efeito combinado da existência de uma fonte sonora numa certa

sala, simulando um orador e vários alto-falantes, cada um com suas características de direcionalidade, amplificação e atrasos [Kut00].

Há vários métodos usados para criar fontes sonoras onidirecionais, dentre os quais destacamos o método determinístico da subdivisão geodésica do icosaedro regular, proposto por Lewers, e o método da fonte sonora pseudo-aleatória descrito por Kulowski [Kul85].

3.2.1 Fonte Sonora Determinística

Para Lewers [Lew93], os raios emitidos por uma fonte sonora devem ter ângulos iguais entre si. Para conseguir construir tal fonte, o princípio da subdivisão geodésica de uma esfera foi incorporado ao método descrito por ele. Neste método, as vinte faces do icosaedro regular são divididas em triângulos menores que são projetados na superfície de uma esfera que circunscreve o icosaedro. A direção de um raio a ser seguido é, então, determinada pela junção entre o centro da esfera e o centro da área dos triângulos esféricos resultantes.

Uma forma de construir tal fonte é partir dos vértices do icosaedro regular, cujas coordenadas num espaço tridimensional são dadas por:

$$\begin{gathered}(0, \pm 1, \pm \varphi)\,, \\ (\pm 1, \pm \varphi, 0)\; e \\ (\pm \varphi, 0, \pm 1)\,,\end{gathered}$$

onde $\varphi = \left(1 + \sqrt{5}\right)/2$.

Para se obter os vários pontos da superfície da esfera que circunscreve o icosaedro é preciso subdividir cada aresta em *n* partes, ilustrado na Figura 3.2, de onde podemos ver que $n = \sqrt{(N_r - 2)/10}$.

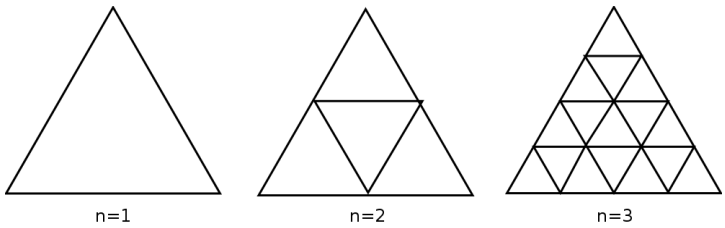

Figura 3.2: Ilustração da subdivisão de uma das faces do icosaedro regular

A Figura 3.3 ilustra parte do processo e como são os pontos obtidos. A fonte sonora obtida com o modelo acima é uma fonte onidirecional e os pontos da superfície da esfera gerada definirão, juntamente com o centro da esfera, a direção dos raios a serem seguidos na simulação [GBD07].

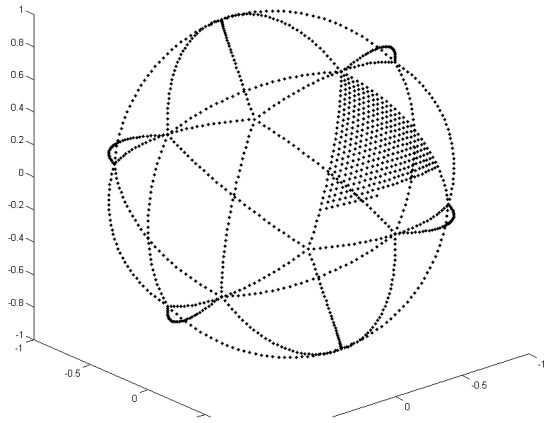

Figura 3.3: Subdivisão geodésica do icosaedro regular circunscrito por uma circunferência

3.2.2 Fonte Sonora Pseudo-Aleatória

O modelo da fonte sonora pseudo-aleatória, descrito por Kulowski [Kul85], tem como princípio a distribuição aleatória uniforme, segundo a qual a probabilidade de ocorrência de qualquer evento é sempre a mesma. Em outras palavras, a probabilidade de se ter um ponto numa determinada área da superfície da esfera deve ser a mesma para se ter outro ponto em qualquer outra região da superfície da esfera que tenha a mesma área. Nesse modelo, os pontos da superfície da esfera precisam ter densidade uniforme e tais pontos em conjunto com o centro da esfera definem a direção dos raios a serem seguidos na simulação.

É possível gerar pontos uniformemente distribuídos sobre a superfície de uma esfera a partir da distribuição aleatória normal [BM58], no entanto, outra técnica extremamente rápida e simples de se gerar tais pontos é a chamada técnica de Monte Carlo. Esta técnica consiste em gerar aleatoriamente três números no intervalo [-1,1] que são usados como as coordenadas (x, y, z) de um ponto no espaço euclidiano [Wei08], de forma que os pontos sejam uniformemente distribuídos num cubo unitário. Depois de gerado cada ponto, basta conferir se tal ponto satisfaz à inequação abaixo.

$$x^2 + y^2 + z^2 <= 1 \tag{3.1}$$

onde os pontos *x, y, z* que satisfizerem à inequação pertencem à esfera unitária. Essa é uma técnica de rejeição, pois os pontos que não satisfazem a inequação (3.1) acima são rejeitados, enquanto os pontos restantes são normalizados e podem ser usados para definir os raios usados na simulação.

Abaixo é possível ver 1000 pontos aleatórios gerados no interior do cubo unitário.

Os pontos que satisfazem o teste da inequação (3.1) são normalizados e os raios são gerados a partir deles e do centro da esfera. Na Figura 3.5 é possível ver 1000 pontos aleatórios que satisfazem tal inequação.

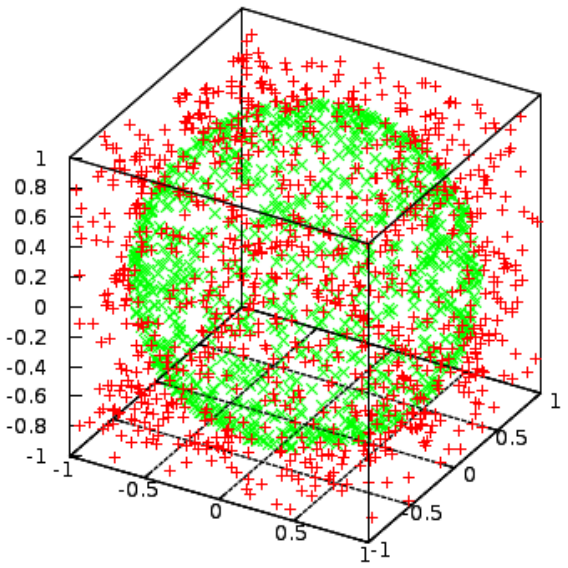

Figura 3.4: Pontos uniformemente distribuídos num cubo unitário. Os pontos do interior da esfera unitária têm cor mais clara, já os pontos rejeitados estão marcados com uma cruz e cor mais escura

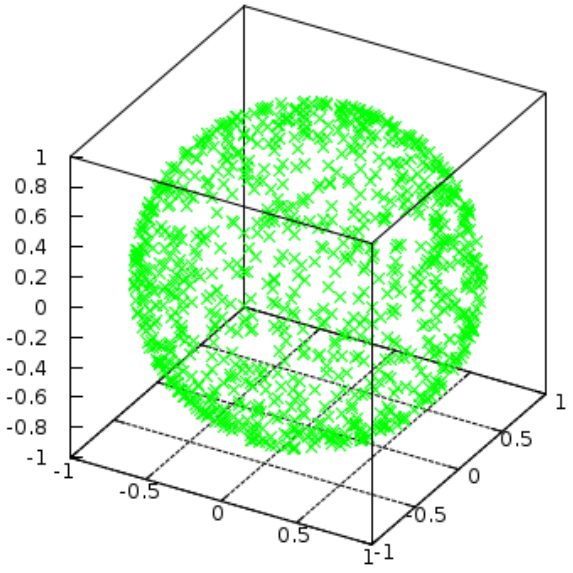

Figura 3.5: 1000 pontos distribuídos uniformemente sobre a superfície de uma esfera unitária

3.3 Reflexão dos Raios

Como dito anteriormente, o traçado de raios segue a teoria matemática da acústica geométrica de salas e, de acordo com essa teoria, os raios podem ser refletidos especularmente ou difusamente. A nova direção de um raio refletido especularmente é dada pelas leis da reflexão geométrica, isto é, o raio é refletido com um ângulo igual ao ângulo de incidência, esses ângulos em relação à normal do plano de incidência [Kut00].

Segundo Kuttruff, sempre que ocorre uma reflexão do raio, além de calcular a nova direção a ser seguida por ele, é preciso considerar o aspecto da energia do raio refletido, pois a cada reflexão o raio perde uma certa quantidade de energia para o obstáculo causador da reflexão, devido à absorção de energia sonora causada por este. Tal perda de energia é calculada a partir do coeficiente de absorção sonoro do obstáculo, representado em nosso modelo por um plano. É preciso considerar, ainda, a quantidade de energia que o raio perde para o ar. Essa informação sobre a energia é fundamental para montarmos a Resposta Impulsiva da sala e, assim, conhecermos seu comportamento acústico. A fim de verificar se um raio irá refletir num certo plano, é preciso seguir os passos descritos a seguir:

3.3.1 Reflexão Especular

A reflexão do raio pode ser ilustrada por três partes principais, sendo elas:

- determinação do plano de incidência do raio
- cálculo da direção do raio refletido
- cálculo da energia do raio refletido

Determinação do Plano de Incidência do Raio

Para determinarmos o plano de incidência do raio, devemos detectar, primeiramente, quais são os possíveis planos de incidência, o que é feito avaliando o produto cartesiano entre os vetores normais de cada plano e o vetor direção do raio, como ilustrado na Figura 3.6.

Assim, para obtermos a lista dos possíveis planos de incidência de um raio a inequação (3.2) deve ser avaliada para todos os planos da sala:

$$\vec{n_i} \cdot \vec{v} < 0 \tag{3.2}$$

onde $\vec{n_i}$ é o vetor normal do i-ésimo plano e $\vec{v}$ é o vetor que representa a direção do raio. Devemos notar que os vetores são normalizados. Caso algum plano satisfaça essa inequação, inicialmente é preciso realizar o teste de ponto no polígono. Tal teste é usado para avaliar se o raio intercepta a parede representada

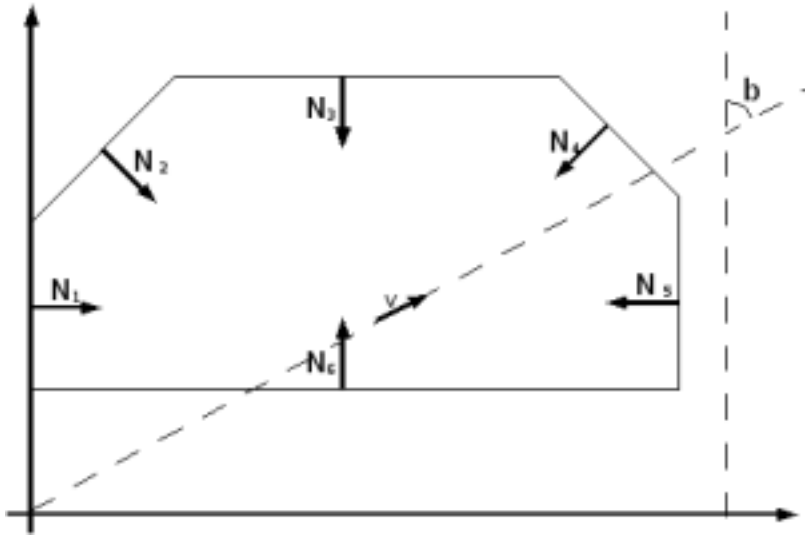

Figura 3.6: Esquema de verificação do plano de incidência do raio, sendo ̣ o ângulo agudo entre $\vec{v}$ e $\vec{n_6}$, nesta figura o raio v não incide no plano 6

pelo plano em questão. Cada plano, por sua vez, é composto por um vetor normal e quatro pontos que determinam um polígono que, em nosso modelo, é um quadrilátero convexo.

Para testarmos se um ponto pertence a um polígono, basta seguirmos os passos propostos por Kulowski. Considerando que queremos testar se um certo ponto Q pertence ou não ao quadrilátero, em primeiro lugar criamos vetores auxiliares, Sj e Q,Rj, j = 1, ..., 4. Como mostrado na Figura 3.7.

Os vetores Sj são direcionados para dentro do polígono e são normais aos lados de tal quadrilátero, enquanto os vetores QRj são formados ao conectarmos o ponto Q à sua projeção Rj nos lados do quadrilátero.

Em segundo lugar, depois de criados os vetores auxiliares, basta fazermos o produto escalar Sj.QRj. Se o resultado for 1 significa que os vetores são compativelmente paralelos, e se o resultado for -1 significa que são anti-paralelos. Assim, se qualquer um dos produtos escalares for 1 significa que o ponto Q não pertence ao polígono em questão.

Após o teste do ponto no polígono, caso o ponto pertença ao polígono, para isso é necessário calcular a distância entre a fonte sonora e o plano atingido pelo raio, o que é ilustrado na Figura 3.8. Tendo essa distância, podemos calcular a distância percorrida pelo raio sonoro, Para realizar o cálculo da distância percorrida pelo raio sonoro entre seu ponto de emissão e o ponto de incidência usamos a equação abaixo.

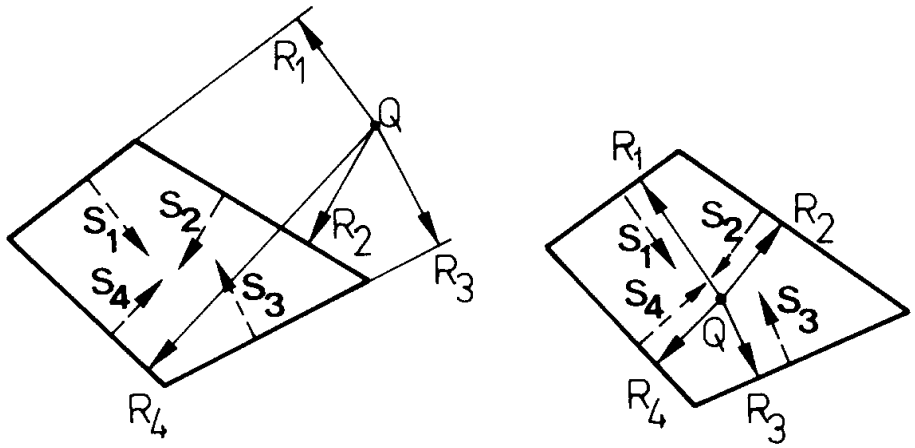

Figura 3.7: Teste de ponto no polígono [Kul85]

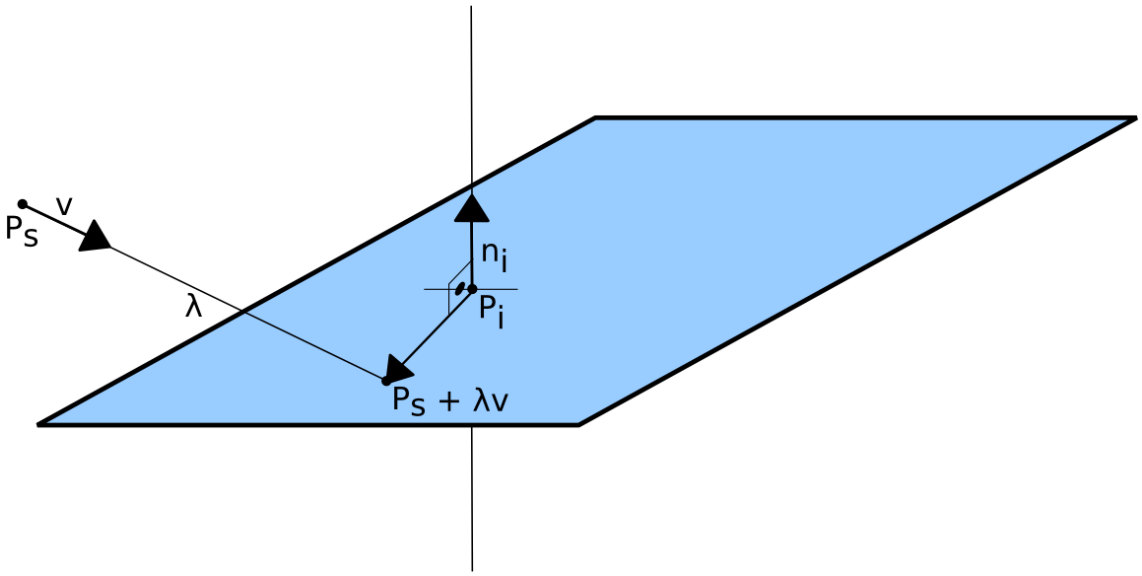

Figura 3.8: Esquema mostrando os vetores usados no cálculo da distância entre p_s e o i-ésimo plano

$$
\begin{aligned}
n_i'.(P_s + \lambda v - P_i) &= 0 \\
n_i' P_s + n_i' \lambda v - n_i' P_i &= 0 \\
\lambda n_i' v &= -n_i' P_s + n_i' P_i
\end{aligned}
$$

$$\lambda = -\frac{n_i'(P_s - Pi)}{n_i' v} \tag{3.3}$$

Onde λ é a distância percorrida pelo raio entre P_s e o ponto Q do i-ésimo plano, já que o vetor v é normalizado. O raio incidirá no plano que satisfizer o teste de ponto no polígono e que tiver a menor distância λ do ponto de emissão do raio, P_s.

Com o valor λ e a partir do vetor do raio, é possível sabermos o ponto onde o raio incidirá no plano, ponto este que, de acordo com a Figura 3.8, é dado por:

$$Q = P_s + \lambda v \tag{3.4}$$

onde Q é o ponto de incidência no plano, λ é a distância e v é o vetor direção do raio em questão.

Cálculo da Direção do Raio Refletido

Nas reflexões especulares, discutidas nessa seção, a direção do raio refletido é determinada a partir do ângulo de incidência ao plano de intersecção. O ângulo de incidência é calculado em relação ao vetor normal deste plano e o ângulo de reflexão é numericamente igual ao ângulo de incidência, como ilustrado na Figura 3.9.

Para calcular a direção do raio de reflexão, usamos a definição de produto interno. Desta maneira, é possível obter as relações abaixo:

$$
\begin{cases}
P_s - P_0 = \delta . n_i \\
n_i'.(P_i - P_0) = 0
\end{cases}
$$

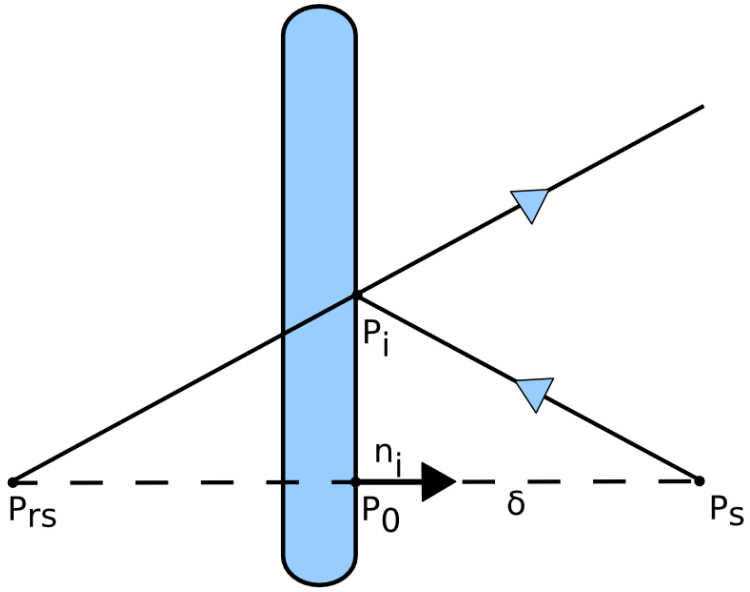

Figura 3.9: Reflexão de um raio. Ps e Prs são a fonte sonora e sua imagem; Pi é um ponto qualquer no plano de incidência do raio sonoro; n_i é o vetor normal ao plano de incidência.

$$
\begin{aligned}
n_i'.P_s - n_i'.P_i &= n_i'\delta n_i \\
\delta &= \frac{n_i'.(P_s - P_i)}{\|n_i\|^2} \\
\delta &= \frac{n_i'.(P_s - Q)}{\|n_i\|^2} \\
\delta &= n_i'.(P_s - Q)
\end{aligned}
$$

Usando o valor de δ, podemos obter o ponto P_{rs}

$$P_{rs} = P_s - n_i \cdot 2 \cdot \delta \tag{3.5}$$

que, junto com o ponto de incidência Q, dado pela equação (3.4), determina a direção do raio refletido. Depois de obtermos essa direção é necessário normalizarmos o vetor obtido, a fim de continuar o processo iterativo. Abaixo mostraremos como é calculada a energia do raio refletido.

Cálculo da Energia do Raio Refletido

O raio sonoro perde energia ao sofrer reflexões, pois parte de sua energia é transferida para as paredes onde são feitas estas reflexões. A atenuação da energia causada pela parede é descrita pelo parâmetro α que é o coeficiente de absorção sonora. É possível usar o coeficiente de absorção sonora de duas maneiras. O primeiro modo é reduzir a energia do raio por um fator *1*-α sempre que ocorrer uma reflexão

(onde α é dado de acordo com cada parede). O segundo modo é interpretar α como um "coeficiente de probabilidade", isto é, um coeficiente usado para gerar aleatoriamente um outro número, o qual, por sua vez, pode ser usado para determinar se o raio prosseguirá em sua nova trajetória, ou se será absorvido [Kut00].

Quando uma onda sonora é propagada numa sala, ela também sofre uma atenuação causada pelo meio viscoso no qual ela se propaga, no nosso caso, o ar. Tal atenuação se caracteriza pela perda de energia e consequente perda de intensidade das ondas sonoras.

Ainda segundo Kuttruff, cada elemento de volume, numa onda sonora, torna-se periodicamente mais comprido ou curto em direção à propagação do som. O meio viscoso no qual a onda viaja oferece uma reação elástica que é proporcional à compressão exercida pela onda sonora. Esta reação é controlada por forças devidas ao meio viscoso e também proporcionais à velocidade dos elementos da onda. Em consequência disso, a energia sonora é convertida de forma irreversível para energia térmica. Portanto, em um meio com perdas, a energia sonora não se mantém a mesma. Uma forma de modelar essas perdas é através da constante de atenuação do ar. Em nosso modelo, a perda de energia é dada pela equação

$$E_{nova} = E_{velha} e^{-m.l} \tag{3.6}$$

onde *m* é a constante de atenuação calculada a partir da temperatura ambiente, da umidade relativa do ar e da frequência do som, o cálculo desse coeficiente é ditado pela norma ISO 9613-1; *l* é a distância percorrida pelo raio; α é o coeficiente de absorção sonora do material usado na parede onde o raio é refletido.

A cada reflexão do raio sabemos a sua energia inicial e calculamos a energia do raio refletido baseando-nos na Equação (3.7), o que faz com que a energia do raio decresça exponencialmente. Calculamos também a energia perdida para a parede, perda esta dada pelo coeficiente α, resultando a equação abaixo:

$$E_{nova} = (1 - \alpha_i) E_{velha} e^{-m.l} \tag{3.7}$$

que é a equação usada em nosso modelo de simulação acústica.

3.3.2 Reflexão Difusa

Quando uma onda sonora é refletida numa parede, parte de sua energia é espalhada por vários ângulos, criando novas ondas de menor intensidade em várias novas direções. Uma forma de modelar esse efeito físico é através da reflexão difusa. Existem, atualmente, duas normas que padronizam como calcular os coeficientes que medem tal efeito. Um deles é o coeficiente de espalhamento, definido pela norma ISO 17497-1 (publicada em Maio de 2004), que determina a parcela de energia que é espalhada por direções distintas da direção de reflexão especular [GVG04]. O segundo coeficiente, feito para expressar

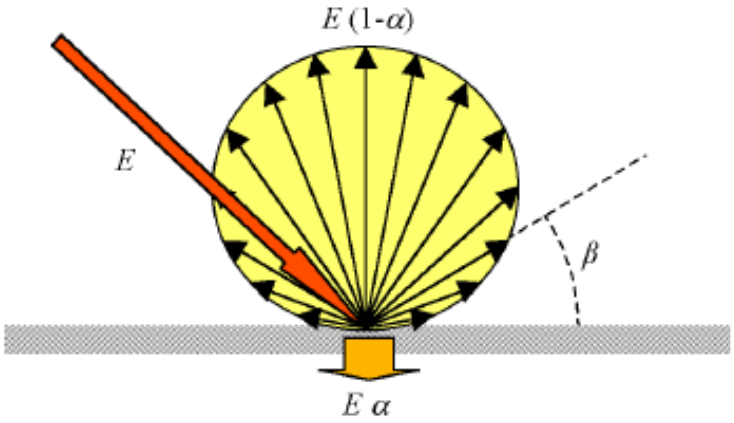

Figura 3.10: Ilustração da distribuição dos raios numa reflexão difusa [TC06]

o grau de uniformidade da distribuição angular da energia sonora espalhada, é chamado de coeficiente de difusão e é definido pela norma AES-4id-2001 (2001) [TC06].

A intensidade do som espalhado na direção caracterizada pelo ângulo υ, medida a uma distância *r* do elemento de superfície *dS*, é dada por:

$$I\left(r\right) = I_0 dS \frac{\cos\upsilon \cos\upsilon_0}{\pi r^2} = B_0 dS \frac{\cos\upsilon}{\pi r^2}$$

onde B_0 é chamado de "reta de irradiação", isto é, a energia por segundo que incide numa unidade de área da parede. Caso a parede também absorva energia, é preciso levar em consideração o coeficiente de sua absorção, fazendo com que a equação acima seja multiplicada por um fator $(1 - \alpha\left(\upsilon\right))$. Podemos ver na Figura 3.10, uma ilustração de tal equação.

3.4 Receptor Sonoro

Os resultados de uma simulação acústica são coletados por meio de receptores sonoros simulados que nada mais são do que "contadores", isto é, áreas ou volumes que guardam informações sobre os raios sonoros da sala. Sempre que um raio sonoro cruza um desses receptores, sua energia e tempo no qual ocorre este evento são armazenados. É possível armazenar também o raio incidente, dado necessário para realizar a auralização da sala [Kut00].

Para Zeng [XKJ03], uma boa modelagem do receptor sonoro é importante para o correto funcionamento dos algoritmos de traçado de raios, porém a maioria dos modelos de traçado de raios se preocupa com a emissão dos raios e suas reflexões nas paredes da sala, negligenciando o modelo do receptor sonoro. Nesses casos, os receptores sonoros mal modelados podem levar a inúmeros erros de detecção que podem afetar a precisão dos resultados da simulação.

Uma maneira de construir o receptor sonoro é usar o modelo de um receptor esférico onidirecional.

Apesar do receptor ser indispensável ao traçado de raios, determinar o melhor modelo de receptor ainda é um problema de pesquisa [JQ07]. Há três tipos de receptores esféricos, sendo que, no primeiro tipo, o raio do receptor é constante para diferentes posições e salas. Para Kuttruff [Kut00], é possível usar um valor típico, geralmente é de 0,5 m ou 1,0 m, para o raio do receptor. Já Zeng [XKJ03], mostra um modelo no qual o raio do receptor é calculado pela relação entre o volume da sala e o número de raios gerados,

$$r = \left(\frac{15V}{2\pi.N}\right)^{\frac{1}{3}}$$

onde *r* é o raio do receptor, *V* é o volume da sala e *N* é o número de raios gerados.

O segundo tipo de receptor tem o raio fixo para uma determinada sala, no entanto, este raio pode ser diferente para salas diferentes, já que esse modelo leva em consideração alguns aspectos da geometria da sala, além de seu volume. Um exemplo desse tipo receptor, proposto por Zhongjin [JQ07], é definido pela Equação (3.8)

$$r = \sqrt{\frac{N\left(\frac{S}{2} + 4V\left(\frac{1}{\sqrt{w^2+l^2}} + \frac{1}{\sqrt{w^2+h^2}} + \frac{1}{\sqrt{l^2+h^2}}\right)\right)}{9\pi N log_{1-\alpha} C_{th}}} \tag{3.8}$$

onde *N* é o número de raios, *S* é a área da superfície, *V* é o volume, *w* é a largura, *l* é o comprimento, *h* é a altura da sala, α é o coeficiente de absorção das paredes e C_{th} é um coeficiente de limiar dos raios.

No terceiro tipo de receptor, o raio do receptor é variável para uma determinada sala e também para diferentes raios nela traçados. Um exemplo é o modelo proposto por Hilmar [Leh92], no qual o raio é dado por

$$r = ct\sqrt{\frac{2\pi}{N}}$$

onde *c* é a velocidade do som, *t* é o tempo no qual o raio atinge o receptor e *N* é o número de raios.

Para quaisquer dos receptores esféricos modelados é preciso verificar se um certo raio atinge tal receptor. Na seção que segue, explicaremos como verificar se um raio atinge o receptor esférico.

3.4.1 Detecção do Raio pelo Receptor Esférico

Para calcular se um raio incide no receptor esférico, usamos técnicas simples de geometria analítica. Observando a Figura 3.11, é possível ver que a projeção de *c* em $p_s + \alpha v$ é perpendicular a v.

Sabendo que a a projeção de *c* em $p_s + \alpha v$ é perpendicular a v, temos que

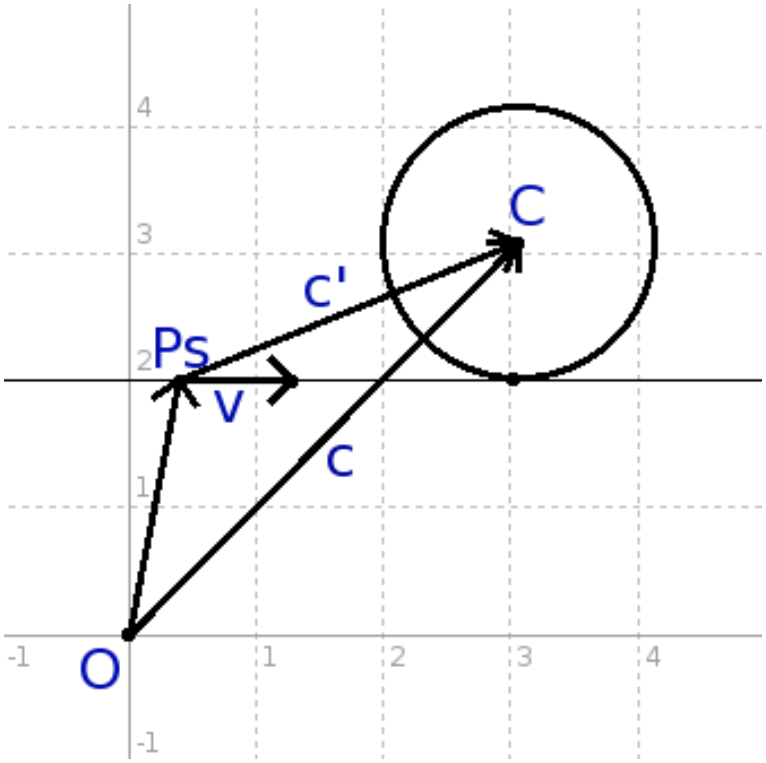

Figura 3.11: Esquema para ilustrar se um raio atinge o receptor esférico

$$
\begin{aligned}
cv' - p_s v' - \alpha \|v\|^2 = 0 \\
\alpha = \frac{v'\,(c - p_s)}{\|v\|^2} \\
fazemos \\
\|c - (p_s + \alpha v)\,\| \leq r
\end{aligned}
$$

$$
\|c - \left(p_s + \frac{v'\,(c - p_s)\,v}{\|v\|^2}\right)\| \leq r \tag{3.9}
$$

caso o raio sonoro satisfaça a inequação 3.9 e a inequação $\alpha < \lambda$, onde λ é definido pela Equação (3.3), então o raio sonoro intercepta o receptor esférico.

3.4.2 Cálculo da Energia do Raio Interceptado pelo Receptor Esférico

Segundo Gomes [GBD07], é preciso usar um fator de correção da energia do raio antes de armazenar seu valor e tempo no qual o raio intercepta o receptor esférico. A energia do raio incidente no receptor é dada por

$$
E_R = \frac{L}{2r} E
$$

onde *E* é a energia do raio ao atingir o receptor esférico, *L* é o comprimento do raio dentro da esfera e *r* é o raio da esfera. Essa correção na energia implica que um raio que apenas "raspe" no receptor esférico contribuirá somente com uma fração de sua energia para o receptor. Já um raio que atinge o centro do receptor esférico terá toda a sua energia armazenada por este receptor.

A informação sobre energia e tempo dos raios que atingem o receptor esférico serão usadas após o rastreamento da trajetória de todos os raios da simulação, de modo a construir a resposta impulsiva da sala.

3.5 Resposta Impulsiva

Ao fim da simulação, obtemos a Resposta Impulsiva (RI) a partir dos dados guardados no receptor. A RI é a resposta da sala no domínio do tempo a um impulso de entrada idealmente curto (Delta de Dirac), dado pela fonte sonora [Kut00]. A Resposta Impulsiva pode ser entendida também como a distribuição temporal das várias reflexões sonoras [Elo05].

Cada raio sonoro contribui com informação para a construção da Resposta Impulsiva, que pode ser entendida, ainda, como uma sumarização da trajetória de cada raio numa determinada sala. Então, todas as perdas de energia para o ar, para as paredes da sala e todas as reflexões sofridas pelo raio contribuem para a determinação da Resposta Impulsiva.

É a partir da Resposta Impulsiva que podemos calcular vários parâmetros acústicos como a definição *D*, o tempo central t_s e o parâmetro de energia lateral *LEF*, dentre outros [Kut00], parâmetros estes que nos auxiliam a entender como é o comportamento acústico de uma sala. Na Figura 3.5 é possível ver a Resposta Impulsiva de uma medição, bem como a RI de uma simulação.

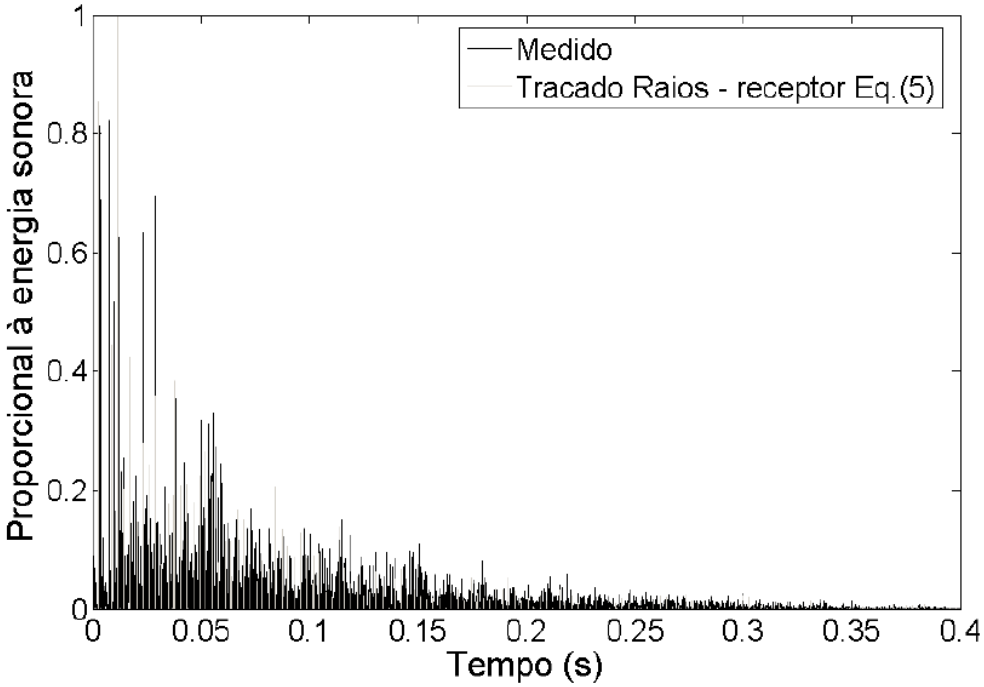

Figura 3.12: Respostas Impulsivas medida e simulada através do método de traçado de raios [GBD07]

A Resposta Impulsiva é parte fundamental do processo de auralização, que consiste em tornar audíveis os resultados de uma simulação acústica. Para se fazer a auralização é preciso guardar a Resposta Impulsiva na forma de uma matriz que guarda a informação sobre a energia e tempo de cada raio interceptado pelo receptor esférico e não somente a soma da energia de tais raios [GBD07].

A resolução temporal da Resposta Impulsiva tem uma relação direta com o raio do receptor sonoro [GBD07]. Essa resolução temporal é crítica, pois se o intervalo de amostragem for muito longo, a Resposta Impulsiva será apenas uma aproximação da verdadeira Resposta Impulsiva da sala, já que detalhes importantes são perdidos devido às médias. Por outro lado, intervalos de amostragem muito curtos interferem na Resposta Impulsiva, devido ao aparecimento de flutuações aleatórias de energia em tais intervalos [Kut00].

Kuttruff aponta que uma regra prática é a resolução temporal ser entre 5 e 10ms, porque este intervalo corresponde aproximadamente à resolução temporal alcançada por nosso sistema auditivo. Uma forma de se chegar a esse intervalo é representar a energia dos raios a cada 2r/c segundos, onde r é o raio original do receptor esférico e c é a velocidade do som [GBD07].

3.6 Algoritmo de Traçado de Raios

Uma vez que já descrevemos as etapas necessárias para a técnica de traçado de raios, mostraremos na Figura 3.13 como o algoritmo é construído. Como precisamos rastrear a trajetória de cada raio sonoro gerado, a simulação acústica por traçado de raios é uma técnica iterativa: simula as seguidas reflexões que um raio sonoro sofre ao refletir nas paredes de uma sala. Inicialmente, é preciso gerar os raios (linha 2 da Figura 3.13, explicado na Seção 3.2) que serão rastreados. Cada raio recebe uma energia inicial e é verificado qual é o plano de incidência do raio (linha 8, explicado Seção 3.3.1). Em seguida (linha 11), verificamos se o raio atinge o receptor esférico, como visto na Seção 3.4 e, caso o raio intercepte tal receptor, seu tempo e energia serão armazenados, o que é feito na linha 12. Depois disso, rastreia-se o próximo raio gerado pela fonte sonora.

Caso o raio não atinja o receptor esférico, nas linhas 16 e 17 o algoritmo calcula sua nova energia e nova direção de propagação. Na Seção 3.3.1, explicamos como o raio perde energia, por causa da colisão com a parede e da atenuação causada pelo ar.

É interessante notar que a simulação é feita para cada banda de frequência, pois os materiais com os quais são feitas as paredes das salas apresentam um certa taxa de absorção sonora, que varia entre as bandas de frequência. Assim, o resultado de uma simulação é válido para uma certa banda, como também o cálculo de parâmetros acústicos [Kut00].

De posse da informação armazenada no receptor, o algoritmo constrói a Resposta Impulsiva (RI) da sala, linha 23, a qual corresponde a uma resposta energética da sala em relação a um impulso de entrada dado pela fonte sonora, explicada na Seção 3.5.

A precisão dos resultados obtidos depende principalmente do número de raios contados pelo receptor esférico. Consequentemente, o número de raios gerados deve ser suficientemente grande, em torno de 10^5 a 10^6 raios [Kut00].

```
1  Traçado de Raios()
2  gerar raios a partir da fonte sonora
3  enquanto existir raio não rastreado
4  {
5    definir energia inicial do raio
6    enquanto raio tem energia suficiente
7    {
8      percorrer planos da sala e fazer teste de ponto no polígono
9      se plano é atingido
10     {
11       se o raio atingir o receptor esférico
12       {
13         guardar energia e tempo no qual o raio atingiu o receptor esférico
14         rastrear próximo raio
15       } senão {
16         diminuir energia do raio devido à colisão com parede e à atenuação pelo ar
17         definir nova direção do raio
18       }
19     }
20   }
21 }
22
23 montar Resposta Impulsiva a partir dos dados guardados no receptor esférico
```

Figura 3.13: Algoritmo de traçado de raios

Capítulo 4

Ambiente AcMus

O projeto AcMus foi criado em 2002, como um esforço conjunto de pesquisadores do Departamento de Música e de Ciência da Computação da Universidade de São Paulo que tinham por objetivo consolidar uma comunidade de pesquisa devotada a estudar assuntos relacionados a acústica de salas para prática musical. Um dos objetivos do projeto era construir um software que tornasse mais simples a tarefa de fazer medições, análises e simulações da acústica dessas salas [IKS01].

O sistema AcMus foi desenvolvido com o intuito de integrar em um único ambiente, ferramentas para realizar medições acústicas, que são tratadas pelo Módulo de Medição e realizar simulações da acústica de sala, que são feitas pelo Módulo de Simulação, que provê ferramentas para simular o comportamento acústico de salas cubóides e salas com geometria mais complexa, poliédrica. Por ser um ambiente integrado e prover todas as funcionalidades necessárias para analisar a acústica de um único ambiente, no AcMus o usuário não precisa manter arquivos de vários sistemas distintos, como é comum em outros ambientes.

O software deveria atender da melhor maneira possível a seu público alvo, que seria diversificado como músicos, engenheiros de áudio, engenheiros acústicos e mesmo usuários domésticos. Assim, decidiu-se desenvolver o sistema numa linguagem multiplataforma e orientada a objetos, Java, e tendo em vista a usabilidade do sistema optou-se por uma interface gráfica amigável, que foi construída usando a tecnologia SWT/JFace *(Standard Widget Toolkit)*. O SWT/JFace é um sistema que provê um rico conjunto de elementos de tela nativos do sistema operacional usado pelo usuário e, além disso, o comportamento de janelas sempre segue o comportamento padrão definido pelo Sistema Operacional onde o sistema é executado [Guo05].

Outro objetivo era criar uma comunidade de desenvolvedores para o sistema, assim, o software foi criado como um software livre e licenciado sob a licença GPL *(General Public Licence)*, licença que garante o que Richard Stallman chama de quatro liberdades fundamentais de usuário de software: (0) liberdade para usar o programa para qualquer fim, (1) liberdade para estudar como um certo programa funciona e modificá-lo da maneira que achar melhor, (2) liberdade para poder redistribuir cópias do

programa e (3) liberdade para melhorar o programa e redistribuir melhorias [Sta08]. Seguindo essa filosofia, o código fonte do programa está disponível junto com os pacotes binários do sistema e ficam disponíveis na Web através do sítio (http://gsd.ime.usp.br/acmus), para que qualquer usuário que desejar fazer quaisquer modificações, possa fazê-lo.

Por fim, ainda pensando na comunidade de desenvolvimento e no futuro desenvolvimento do sistema, foi escolhida a plataforma Eclipse. Ela oferece um poderoso ambiente, baseado em componentes, para construção de aplicações com interfaces gráficas complexas e que permite a ampliação de sistemas construídos sobre ela, através de um mecanismo de *Plug-ins* [Fou07].

É importante salientar que os módulos de medição e análise foram extensivamente testados através de várias medições da acústica das principais salas de teatro de São Paulo. Os resultados obtidos destas medições foram comparados com os resultados de outros programas já consagrados como o plugin Aurora do software Adobe Auditorion, do engenheiro Angelo Farina, sendo esses resultados completamente satisfatórios [Fig05].

Nas seções seguintes apresentaremos os quatro principais módulos do AcMus, junto a uma explicação das ferramentas disponíveis em cada um deles.

4.1 Arquitetura

Nesta seção, descrevemos a arquitetura do sistema AcMus que é composta por três módulos principais que se intercomunicam para satisfazer às requisições do usuário. Os módulos são: Medição, responsável por gerar o sinal sonoro e pela sua gravação (vide Seção 4.1.1), Utilidades que contém várias ferramentas de auxílio ao trabalho com arquivos de áudio (vide Seção 4.1.2) e Simulação, brevemente explicado na Seção 4.1.3 e minuciosamente explicado na Seção 4.2.

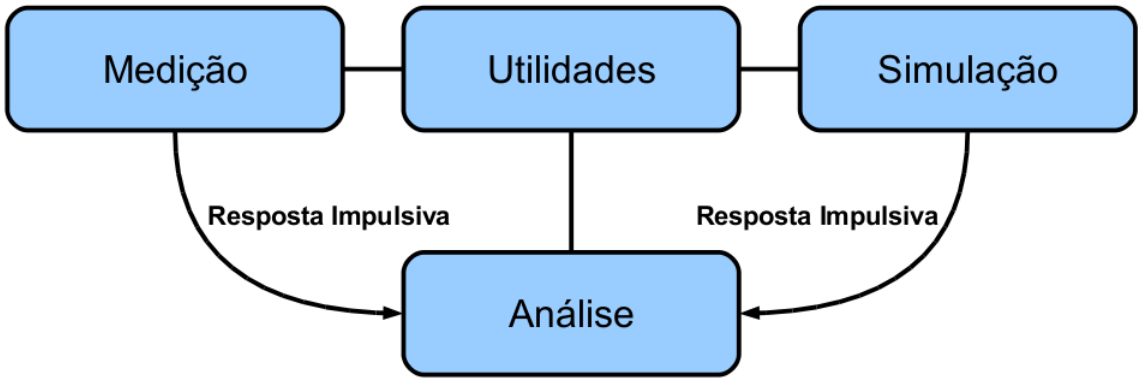

Figura 4.1: Arquitetura do ambiente AcMus

4.1.1 Módulo de Medição Acústica

Este módulo é usado para ajudar o usuário a fazer as várias medições da acústica de uma sala. O processo de medição consiste em reproduzir, dentro da sala estudada, um sinal sonoro e gravar a resposta impulsiva da sala [UFIK05]. O próprio sistema cuida da reprodução e gravação do som para evitar possíveis erros do usuário. Todas as gravações são armazenadas para poderem ser analisadas depois das medições. É possível escolher dentre dois sinais sonoros e, então, definir a duração do sinal escolhido, a qual depende do tamanho e tempo de reverberação estimado da sala. Os sinais implementados são o sinal de Sequência de Tamanho Máximo *(Maximum Length Sequence - MLS)* e o sinal de Varredura Logarítmica *(Log Sweep FFT - LSF)*.

Após realizar as medições acústicas, é possível usar o subsistema de Análise que é composto por vários algoritmos de análise e cálculo de parâmetros acústicos. Segundo Iazzetta et al. [IFM04], tais parâmetros formam uma base científica para a análise acústica das salas de música, pois podem ser calculados a partir de medidas objetivas. Para calcular os parâmetros acústicos, o módulo de análise precisa de uma resposta impulsiva da sala estudada. Essa resposta pode ser medida usando o próprio AcMus ou através de outro software, para depois ser importada para o ambiente.

A partir da resposta impulsiva, o AcMus pode calcular o RT60 e o EDT, que indicam o intervalo de tempo de decaimento do som, e calcular também os parâmetros C80, D50, RDR e LFC que são razões de energia tomadas ao longo do tempo. O AcMus pode calcular, ainda, o ITDG que é calculado a partir do instante de captação do primeiro som a chegar no receptor - instante representado pelo primeiro valor de energia da resposta impulsiva - e do instante de chegada da primeira reflexão sonora. Para um estudo detalhado sobre os vários parâmetros acústicos calculados pelo AcMus, indicamos o trabalho de Bruno Masiero [Mas04] e, para uma análise sobre a interpretação dos diversos parâmetros acústicos calculados pelo ambiente, indicamos a dissertação de mestrado de Fabio Leão [Fig05], pois ambos estudaram esses assuntos em profundidade. Na Figura 4.2 é possível ver uma tabela com os vários parâmetros acústicos calculados pelo AcMus.

4.1.2 Módulo de Utilidades

No sistema AcMus há várias funcionalidades usadas pelos módulos de Medição, Simulação e Análise, como a manipulação de arquivos de áudio *(em formato WAV)*, exibição de uma representação gráfica desses arquivos, importação e exportação de arquivos, além de algoritmos que ajudam a descrever a acústica de salas.

Devido a esta necessidade de oferecer serviços comuns a vários módulos, foi criado o módulo de Utilidades que oferece ferramentas úteis para o projeto de salas, medições acústicas e também para processamento de áudio.

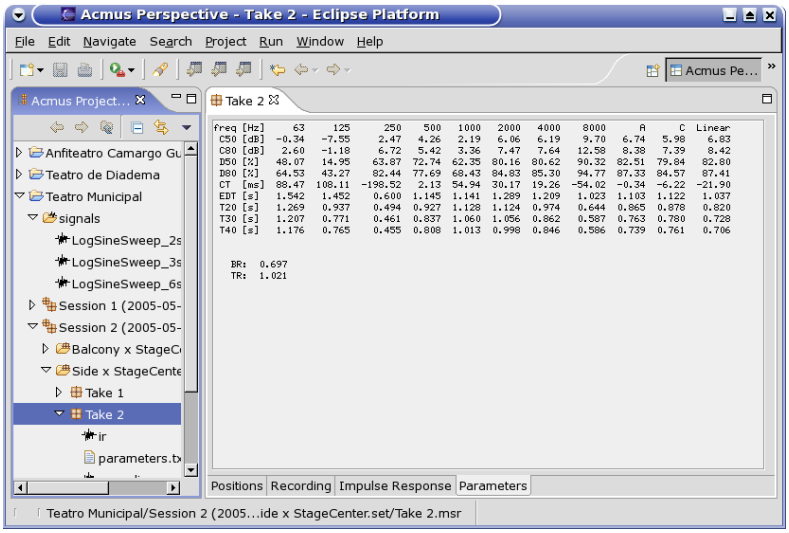

Figura 4.2: Cálculo de Parâmetros Acústicos [UFIK05]

4.1.3 Módulo de Simulação Acústica

O módulo de Simulação Acústica, foco de nossa dissertação, é responsável por realizar a simulação acústica de salas. Optamos por desenvolver um algoritmo de simulação acústica baseado na acústica geométrica de salas, o algoritmo de Traçado de Raios, e além de implementá-lo no ambiente AcMus, criamos uma estrutura de classes facilmente reusável, a fim de que outros desenvolvedores também possam implementar outros algoritmos de simulação acústica no ambiente.

Implementamos duas versões do algoritmo de traçado de raios, uma voltada para eficiência, outra voltada a ser facilmente ampliável e mais facilmente usada para experimentações, por exemplo. O Módulo de Simulação acústica se comunica com o módulo de Análise, para que este faça os cálculos dos parâmetros acústicos a partir da resposta impulsiva gerada pela simulação.

4.2 Arquitetura do Módulo de Simulação Acústica

A arquitetura do módulo de simulação acústica é voltada para dois aspectos dos algoritmos de simulação acústica: ampliação com novos algoritmos e desempenho. Porém, é impraticável conseguir atingir simultaneamente esses dois aspectos num mesmo algoritmo. Tendo tal dificuldade em mente, criamos várias interfaces que podem ser usadas pelos algoritmos de simulação, para que cada desenvolvedor tenha a liberdade de otimizar ou não seu código, já que não fica obrigado a implementar nenhuma herança.

Usamos extensivamente o padrão de projeto Strategy [GHJV95] para evitar o acoplamento da aplicação a uma determinada implementação e, assim, facilitar a troca de algoritmos de simulação. É possível

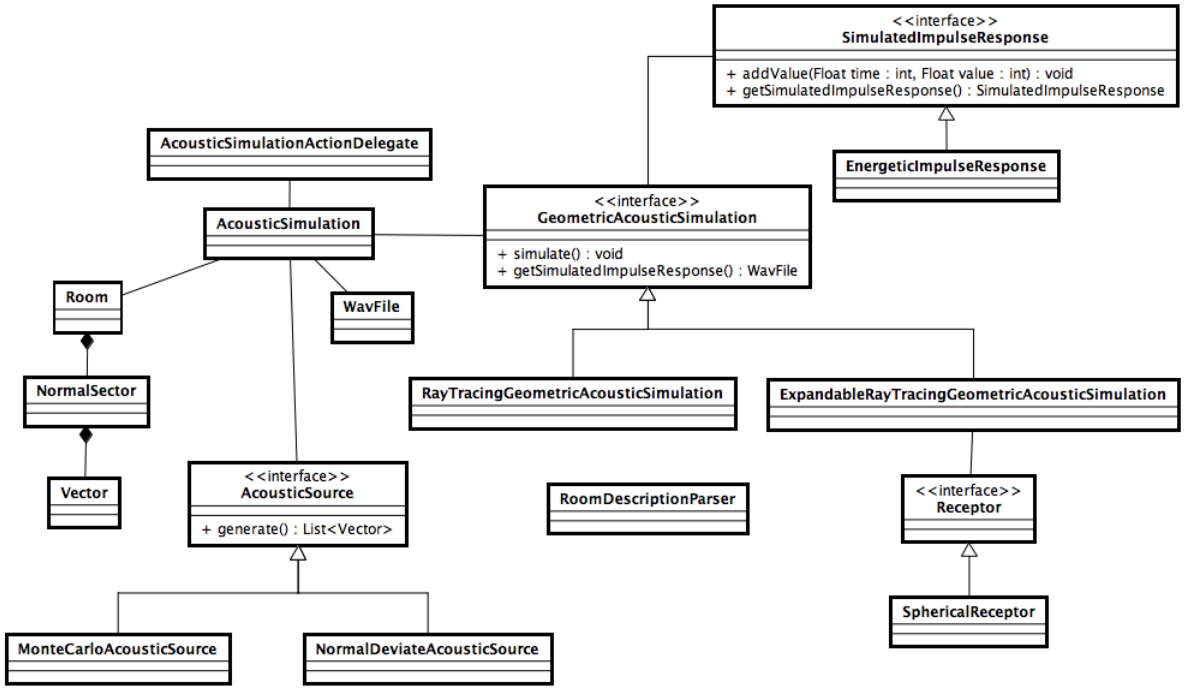

Figura 4.3: Diagrama de Classes do módulo de simulação acústica

ver na Figura 4.3 as principais classes e interfaces do módulo de simulação acústica.

A implementação do método de Traçado de Raios para salas cubóides é feita na classe `RayTracingGeometricAcousticSimulation` que implementa a interface `GeometricAcousticSimulation`. Esse algoritmo de traçado de raios foi implementado de acordo com a descrição de Kulowski [Kul85], usando planos delimitadores para representar as paredes da sala. Os planos são representados pela classe `NormalSector` que armazena um ponto e um vetor normal pertencentes ao plano.

Todos pontos e raios, em nossa arquitetura, são representados pela classe `Vector` que provê objetos imutáveis. Optamos por fazer isso pensando em seu uso numa aplicação paralela, já que simular salas com geometria complexa exige um grande poder computacional e consequente tempo de processamento. Uma alternativa para diminuir o tempo de processamento dos algoritmos de simulação acústica é implementá-los de forma paralela. As funções básicas para manipulação de vetores, disponíveis na classe `Vector`, são:

- `normalize()` - cria vetor normalizado a partir do vetor corrente.
- `sub(Vector otherVector)` - subtrai o valor de `otherVector` do vetor corrente.

- `add(Vector otherVector)` - adiciona o vetor `otherVector` ao vetor corrente.
- `crossProduct(Vector w)` - calcula o produto vetorial do vetor corrente com o vetor w.
- `dotProdutct(Vector w)` - calcula o produto escalar do vetor corrente com o vetor w.
- `length()` - devolve a norma ou módulo do vetor corrente.
- `times(float esc)` - multiplica o vetor corrente por um valor escalar.
- `toString()` - retorna uma `String` representando o vetor corrente, no formato (x,y,z).
- `equals(Object obj)` - avalia se o vetor corrente é igual ao vetor passado como parâmetro. Para verificar essa igualdade, é feita uma avaliação bit a bit dos valores das coordenadas x, y, z dos vetores avaliados.

Os objetos da classe `Vector` armazenam os valores das coordenadas do ponto representado através das coordenadas cartesianas x, y, z. Essas coordenadas são armazenadas em variáveis de ponto flutuante de precisão simples, do tipo `float`.

Todos os algoritmos de simulação acústica implementados no ambiente AcMus, baseados na acústica geométrica, precisam implementar a interface `GeometricAcousticSimulation`, que é parte da implementação do padrão de projeto `Strategy` mencionado acima. Outra interface que precisa ser usada por estes algoritmos é a `SimulatedImpulseResponse`, que descreve a forma como devem ser inseridos dados em uma resposta impulsiva e como o sistema de simulação fará para obter tais dados, através do método `getEnergeticImpulseResponse`.

Essa interface é usada para transferir os resultados obtidos ao realizar a simulação, para o sistema de simulação, independentemente do algoritmo de simulação usado, ficando a cargo do desenvolvedor definir como armazenar os dados de seu algoritmo de simulação. Assim, é possível que cada pesquisador desenvolva uma estratégia para armazenar a resposta impulsiva. Fizemos a implementação da classe `EnergeticImpulseResponse`, que cria um histograma da resposta impulsiva com o intervalo entre os vários pontos do histograma, definido na hora de sua criação. Dessa maneira, é possível que o algoritmo de simulação acústica use a nossa implementação ou crie outra representação para a resposta impulsiva que for mais conveniente ou eficiente para seu algoritmo.

4.2.1 Fonte Sonora

Como explicado no Capítulo 3, os algoritmos baseados na acústica geométrica de salas precisam de um modelo de fonte sonora. A interface `AcousticSource` representa a fonte sonora em nossa arquitetura. Essa fonte sonora de raios pode ter qualquer formato que o desenvolvedor quiser, pois, para o sistema somente interessa que a implementação respeite o contrato de devolver uma lista de `Vectors` com o número de vetores solicitados, o que é definido através do método `generate()`.

Para poder implementar o algortimo de traçado de raios, criamos uma implementação da interface `AcousticSource`, a classe `MonteCarloAcousticSource`, que usa o método de Monte Carlo para gerar pontos que ficam uniformemente distribuídos sobre a superfície de uma esfera unitária. Mesmo descartando 47,64% dos pontos gerados, essa técnica mostrou-se extremamente rápida, uma vez que os cálculos necessários para criar cada ponto são muito simples, como explicado na Seção 3.2.2.

É interessante notar que com essa interface é possível também representar fontes direcionais, como alto-falantes. Na seção de trabalhos futuros, falaremos sobre a possível importação de arquivos CLF *(Common Loud-speaker Format)*, que servem para descrever as características de distribuição de som de alto-falantes.

4.2.2 Receptor Sonoro

Como apontamos no Capítulo 3, uma parte fundamental de um algoritmo de traçado de raios é o receptor sonoro. A interface `Receptor` define a forma para os vários receptores sonoros que podem ser usados numa simulação acústica. Já a classe `SphericalReceptor` implementa essa interface e modela um receptor sonoro esférico de raio fixo.

O receptor implementado na classe `SphericalReceptor` cria uma instância da classe `EnergeticImpulseResponse` que implementa a interface `SimulatedImpulseResponse`, para armazenar o tempo e energia dos raios que o interceptam. O receptor define ainda o intervalo de amostragem dessa resposta impulsiva simulada seguindo a fórmula $2r/c$ definida na Seção 3.5.

4.2.3 Resposta Impulsiva

Como explicado na Seção 3.5, ao realizarmos a simulação acústica de uma sala, obtemos uma resposta dessa sala ao sinal de entrada e a esse resultado dá-se o nome de Resposta Impulsiva (RI). Quando trata-se de uma simulação acústica usando a técnica de traçado de raios, a RI obtida é uma resposta energética da sala, o que significa que a RI mostra a relação da energia inicial emitida pela fonte sonora e a energia recebida pelo receptor sonoro. Como ocorrem perdas de energia, a cada reflexão dos raios e devido ao ar, a RI apresenta uma curva decrescente de valores.

Uma maneira preliminar para se avaliar a resposta impulsiva, é observar sua representação gráfica. Para facilitar tal avaliação, o nosso módulo de simulação permite que qualquer algoritmo de simulação acústica, que retorne uma instância na forma `SimulatedImpulseResponse`, implementado no ambiente AcMus, possa exibir um gráfico da RI obtida. A saída gráfica foi feita usando a biblioteca JFreeChart, que provê um conjunto de classes para desenhar gráficos, usando a plataforma Java [Gil05]. É possível ver na Figura 4.4 uma resposta impulsiva de uma sala para testes.

A Figura 4.4 foi feita usando o algoritmo de traçado de raios para salas cubóides; usamos a implementação `EnergeticImpulseResponse` que define os intervalos do histograma de acordo com a

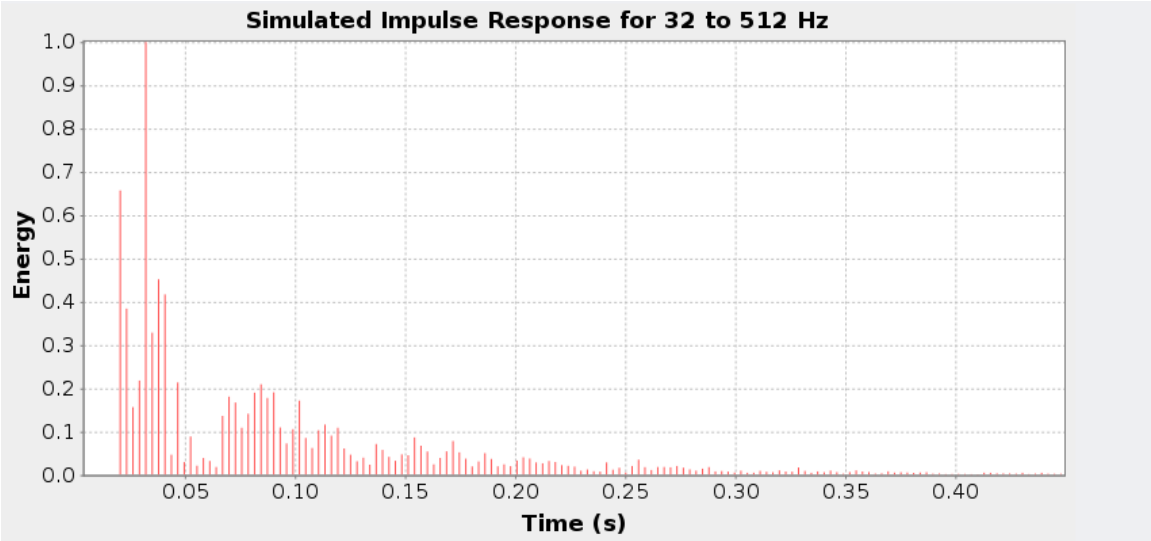

Figura 4.4: Gráfico com resposta impulsiva, gerada e exibida no módulo de simulação.

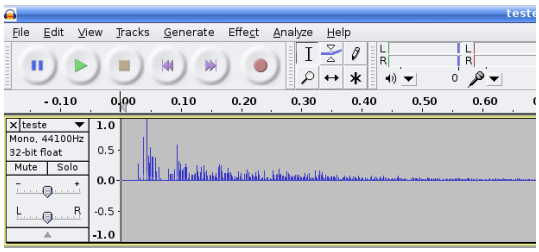

Figura 4.5: Arquivo WAV com uma resposta impulsiva simulada.

fórmula $2r/c$. Os dados usados para a simulação foram de uma sala cubóide com 10 metros de largura, 15 metros de comprimento e 4,5 metros de altura e o receptor sonoro com um raio de 0,5 metro.

Exportação da Resposta Impulsiva

É possível exportar para outros programas a resposta impulsiva obtida pela simulação da sala. O módulo de simulação cria um arquivo WAV *(Waveform audio format)* com taxa de amostragem de 44,1 KHz, um canal e 32 bits por amostra. Dessa maneira é possível usar outros programas para realizar o cálculo de parâmetros acústicos dessa RI, já que o formato WAV é usado por todos programas de simulação acústica. Na Figura 4.5, é possível ver o arquivo da resposta impulsiva no `Audacity`, um software livre para edição de arquivos de áudio.

4.2.4 Desempenho

O ambiente AcMus foi construído com o intuito de ser multiplataforma, daí a decisão de construí-lo usando a linguagem Java, que é compilada para bytecodes que são interpretados pela JVM *(Java Virtual Machine)*. Dessa maneira, os programas escritos em Java funcionam em quaisquer sistemas operacionais para os quais exista uma JVM.

Pelo ambiente AcMus ser escrito nessa linguagem, uma grande preocupação, no início do projeto AcMus, foi o desempenho dos algoritmos implementados. Porém, com o aumento do poder de processamento dos computadores e com a melhoria da JVM, podemos observar que é possível executar simulações acústicas nessa linguagem. A Tabela 4.2.4 mostra o tempo levado para executar a simulação acústica de salas cubóides, em um computador com processador AMD Sempron 3500+ de 1,8GHz.

Número de raios	**Tempo de processamento (s)**
100.000	22
500.000	95
1.000.000	190
2.000.000	394

O número de raios necessários para obter-se parâmetros acústicos confiáveis, depende do tamanho e dos coeficientes de absorção da sala a ser simulada. É interessante notar que para simular uma sala com aproximadamente 400 m^3 e paredes regulares, é necessário gerar por volta de 100.000 raios sonoros. A partir dos dados da tabela acima, podemos traçar um gráfico com o tempo de processamento por número de raios gerados, o que é mostrado na Figura 4.6.

É interessante notar que, como esperado, o tempo de processamente do algoritmo implementado cresce linearmente com o número de raios, o que é mostrado pela reta de regressão linear da Figura 4.6. Para realizar as simulações das quais obtivemos os dados acima, usamos os seguintes parâmetros: sala cubóide de 10 metros de comprimento por 10 metros de largura e 10 metros de altura, coeficiente de absorção das paredes, $\alpha = 0,02$, e k = 500. O algoritmo usado foi o método de traçado de raios implementado na classe `RayTracingGeometricAcousticSimulation`.

4.3 Uso do Módulo de Simulação através da linha de comando

Dependendo do ambiente de trabalho do usuário, pode ser interessante executar as simulações acústicas sem usar toda a estrutura oferecida pelo ambiente AcMus. Desenvolvemos o módulo de simulação acústica de forma a deixá-lo totalmente desconectado da interface gráfica. Para isso usamos o padrão arquitetural MVC [BMR+96], que descreve uma maneira de desacoplarmos o modelo, em nosso caso o modelo da simulação, de sua interface gráfica.

Dessa maneira é possível usar o módulo de simulação acústica através da linha de comando do sistema operacional, ou através de `scripts`. Para isso, basta invocar o método `main()` da classe

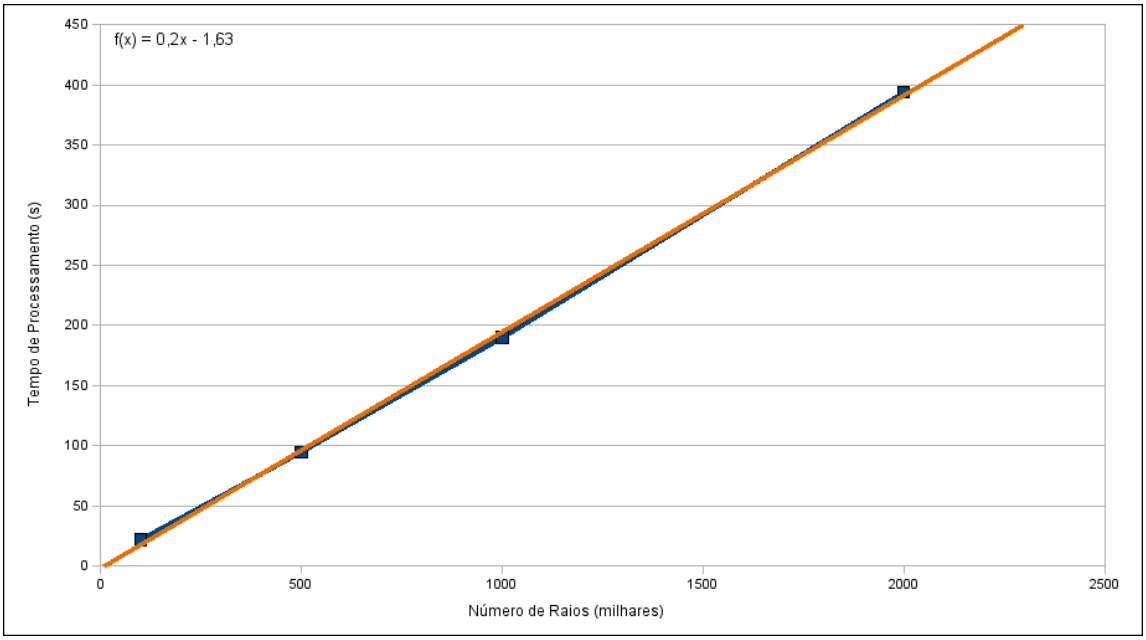

Figura 4.6: Tempo de processamento para simulação em sala cubóide.

`AcousticSimulation` indicando o arquivo de entrada com a descrição geométrica da sala, o algoritmo de simulação acústica a ser usado e os parâmetros para a execução desse algoritmo. O último parâmetro a ser definido pelo usuário é o arquivo de saída. Após executar a simulação acústica, a resposta impulsiva simulada é armazenada, usando o formato `WAV`, no arquivo de saída especificado pelo usuário.

4.4 Integração do Módulo de Simulação ao Ambiente AcMus

Uma das propostas do projeto AcMus é construir um ambiente integrado para medição e simulação acústica de salas, contribuindo também com ferramentas computacionais para melhorar a qualidade da acústica de salas [IKdQ+04].

O ambiente AcMus foi usado para realizar a medição acústica de salas de concerto na cidade de São Paulo. Os resultados obtidos contribuem para mostrar como o software está maduro na área de medição e cálculo de parâmetros acústicos de salas [FI05].

Desta maneira, ampliamos o sistema AcMus para incluir ferramentas de simulação acústica, o que pode ser visto na Figura 4.7, e criamos uma interface gráfica na qual o usuário pode facilmente incluir alguns parâmetros necessários para a realização da simulação acústica, o que pode ser visto na Figura 4.9. Após a execução do algoritmo, é produzida a RI da sala simulada.

Em primeiro lugar, o usuário define os parâmetros usados na simulação acústica, como o coeficiente

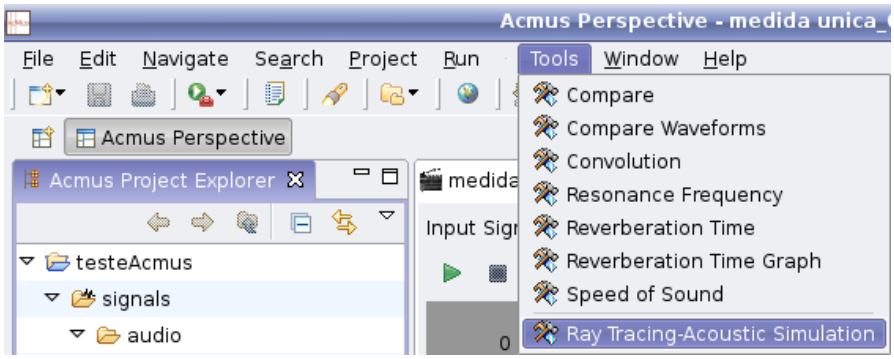

Figura 4.7: Menu de ferramentas do sistema AcMus com a opção para *Ray Tracing Simulation* em destaque

de absorção do ar, velocidade do som, posição da fonte e do receptor sonoros e a geometria da sala. É possível descrever a geometria da sala de duas maneiras: a primeira maneira é descrever uma sala cubóide, o que pode ser feito através da interface gráfica; a segunda maneira é importar um arquivo texto que tenha a definição das paredes da sala e dos respectivos coeficientes de absorção sonora de cada parede. É possível ver, na Figura 4.8, os passos envolvidos para que o ambiente AcMus faça a simulação acústica usando o algoritmo definido na classe `RayTracingGeometricAcousticSimulation`.

Como resultado da execução de um algoritmo de simulação acústica, é possível ver, inicialmente um gráfico ilustrando a resposta impulsiva obtida pela simulação, como mostra a Figura 4.9.

Em segundo lugar, é possível solicitar o cálculo de parâmetros acústicos, que são calculados pelo módulo de análise do AcMus. Salientamos novamente que a simulação é feita para cada banda de frequência, como visto na Seção 3.6.

4.5 Resumo

O ambiente AcMus conta com várias ferramentas para a medição e subsequente análise dos parâmetros acústicos de uma sala e já foi avaliado por vários pesquisadores da área de acústica e de música [Fig05, QIK+08]. Extendemos o ambiente AcMus para que ele incorporasse vários algoritmos de simulação acústica e implementamos duas variações do algoritmo de Traçado de Raios. Ao criar o módulo de simulação acústica, definimos várias interfaces e implementações de componentes que podem ser usados por outros algoritmos de simulação acústica, baseados na geometria acústica de salas.

Por fim, mostramos as principais funções do módulo de simulação acústica e como é possível criar novos algoritmos de simulação dentro módulo. Mostramos também que a linguagem Java se mostrou eficiente a ponto de nos permitir realizar os cálculos necessários pelos algoritmos de simulação acústica implementados.

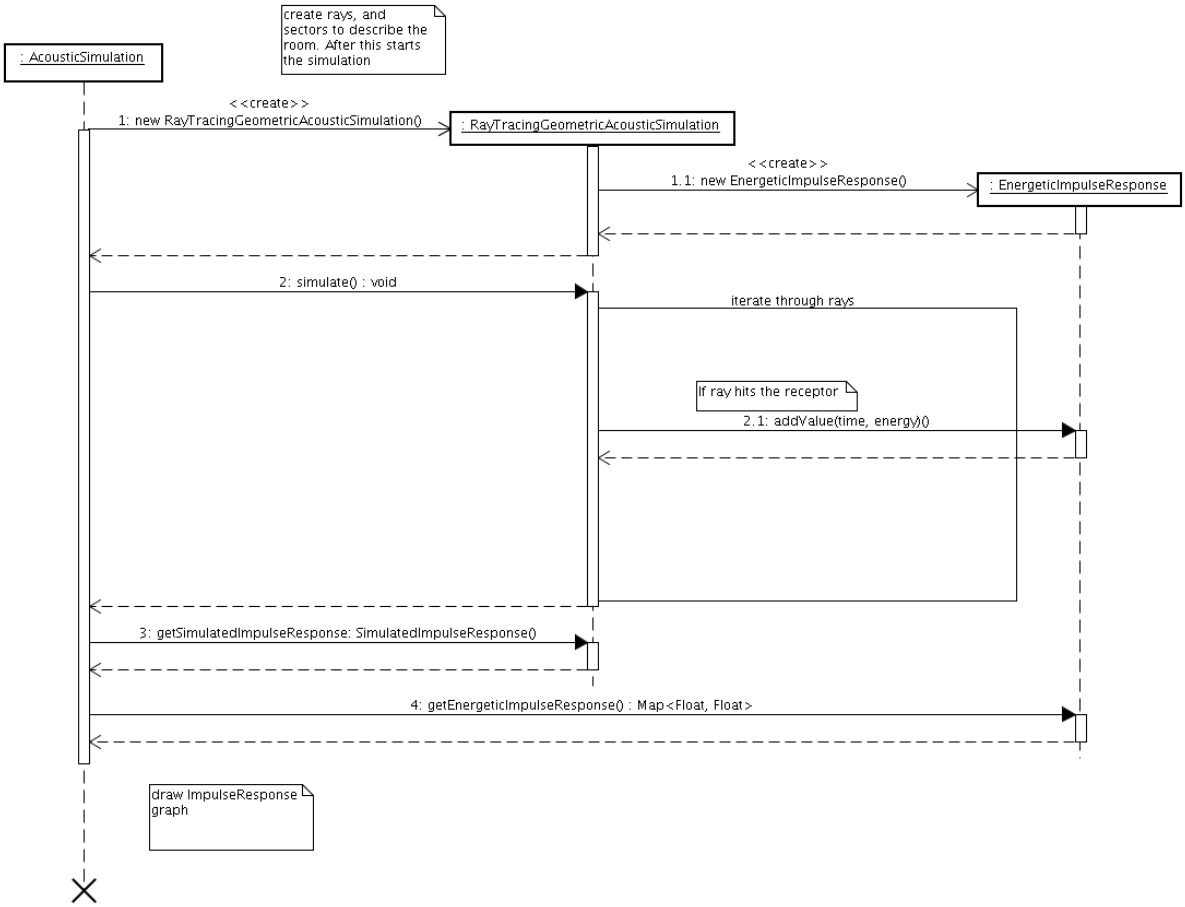

Figura 4.8: Dinâmica de uma simulação acústica.

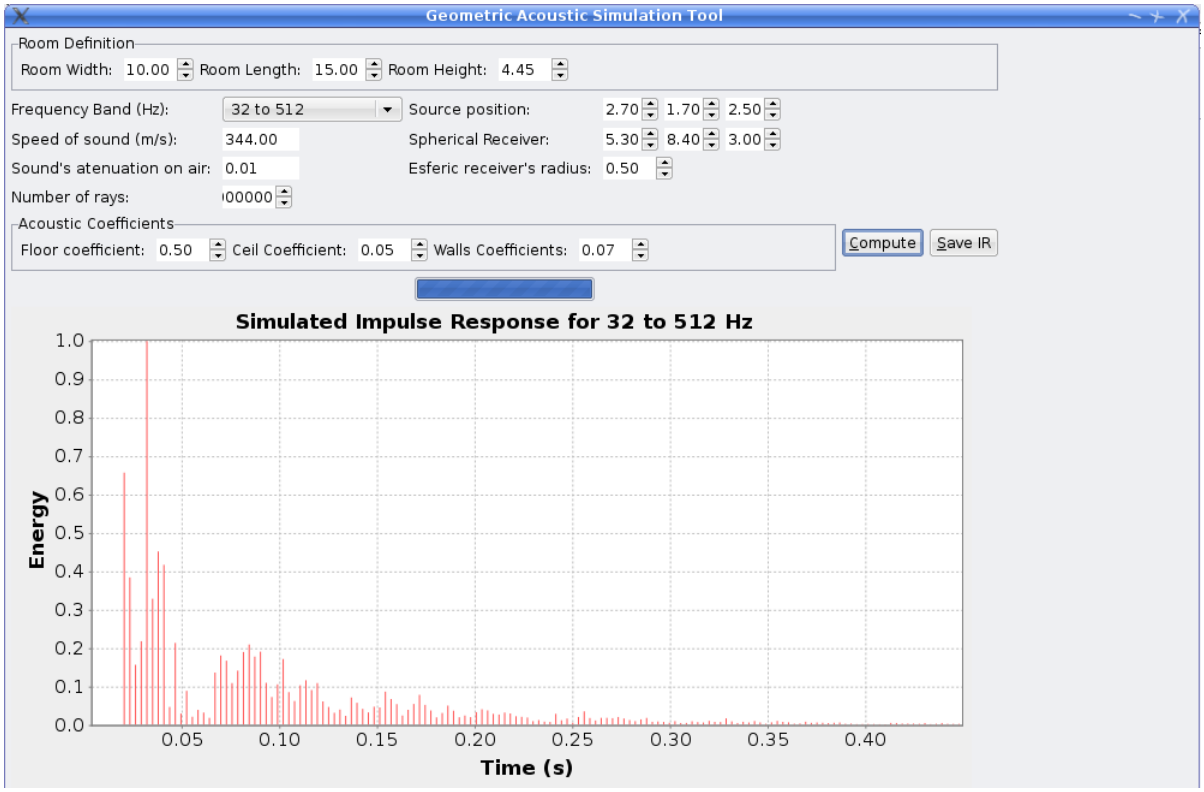

Figura 4.9: Tela do AcMus para simulação acústica de salas cubóides.

Capítulo 5

Avaliação do Módulo de Simulação Acústica

Há uma tendência entre arquitetos e engenheiros em fazer projetos que levem em conta vários aspectos de uma construção, como conforto térmico e acústico, o que gera uma demanda, por parte desses profissionais, por programas de computador que consigam descrever as características dos ambientes projetados [İLA07].

Em face à demanda por tais programas e pela necessidade de entender os seus resultados, o PTB *(Physikalisch-Technische Bundesanstalt)*, o instituto de metrologia da Alemanha, organizou em 1994 a primeira avaliação internacional dos programas de simulação acústica, chamada de *1st International Round Robin on Room Acoustical Simulations*, que contou com 16 participantes de 7 países. O objeto da avaliação foi um auditório localizado na sede do PTB e foram avaliados vários parâmetros acústicos, como T30, EDT, D50, C80, TS, todos filtrados na banda de frequência de 1 kHz. Porém, os resultados obtidos pelos vários participantes mostraram grandes discrepâncias e na época pareceu que seria impossível fazer uma correta predição dos parâmetros acústicos de uma sala [Sch08].

Já na segunda edição do *1st International Round Robin on Room Acoustical Simulations"*, realizada em 1996-98, foi simulada a sala de concertos chamada *ELMIA-hall* localizada em Jönköping, Suécia. Os resultados obtidos nessa segunda edição foram bem melhores, porém a complexidade da geometria dessa sala causou muitos problemas. Assim, para a terceira e mais recente edição do evento, chamada *3rd International Round Robin on Room Acoustical Simulation*, que ocorreu em 2002, uma sala muito mais simples foi escolhida, o estúdio de música do instituto PTB.

Realizamos a avaliação dos algoritmos de simulação implementados no módulo de simulação acústico do ambiente AcMus, usando os dados fornecidos para o *3rd International Round Robin on Room Acoustical Simulations phase 1*. Nessa fase, o estúdio do PTB é modelado de uma forma muito simples, consistindo de 7 paredes, todas com o mesmo coeficiente de absorção. Esse modelo foi usado para poder comparar os resultados obtidos pelos participantes do estudo, sem precisar levar em conta a influência

decorrente da precisão da descrição geométrica da sala, já que todos participantes usaram exatamente os mesmos dados, sem precisar fazer ajustes na representação do modelo.

Para avaliar nosso programa, comparamos os parâmetros acústicos calculados por ele em relação aos parâmetros acústicos calculados pelos programas participantes dessa edição do *3rd International Round Robin on Room Acoustical Simulations* fase 1. Na seção seguinte, descrevemos as condições, parâmetros e o modelo da sala usado em nossa simulação.

5.1 Ambiente de Simulação

Usamos a implementação `RayTracingGeometricAcousticSimulation` que trabalha com o receptor sonoro de raio fixo, o qual definimos como sendo 0,1 metro devido à proximidade entre a fonte sonora e o receptor sonoro. Devemos salientar que o valor para o raio do receptor sonoro não é definido pelos organizadores do experimento. Usamos a fonte sonora onidirecional pseudo-aleatória implementada pela classe `MonteCarloAcousticSource`. Todas simulações foram feitas usando um computador pessoal equipado com processador Intel(R) Core2 Duo 3,0 GHz, 3 GB de memória e sistema operacional Ubuntu 8.04 com núcleo Linux 2.6.24.

5.1.1 Sala Simulada

Como descrevemos acima, o modelo da sala que usamos em nossa simulação é uma versão simplificada da estrutura do estúdio de música do PTB. É possível ver na Figura 5.1 como é a geometria de nosso modelo.

Todas as paredes da sala são modeladas com o mesmo coeficiente de absorção, $\alpha = 0,1$, que é o mesmo para todas bandas de oitava 125 Hz, 250 Hz, 500 Hz, 1kHz, 2 kHz, 4kHz. Os vértices da sala e suas paredes são descritos nas tabelas a seguir. Note que todas as coordenadas nas tabelas seguintes são dadas em metros.

Após definir o modelo da sala a ser simulada, usamos as condições normais que são usados em todas as fases do *3rd International Round Robin on Room Acoustical Simulations*, ao qual passaremos a nos referir como RR3.

Condições Normais e Parâmetros a Serem Calculados Para esta edição do RR3 foram usadas as seguintes condições normais para a realização das simulações:

- temperatura na sala: 20 ^oC
- umidade relativa do ar: 50%
- pressão estática do ar: 1000 hPa

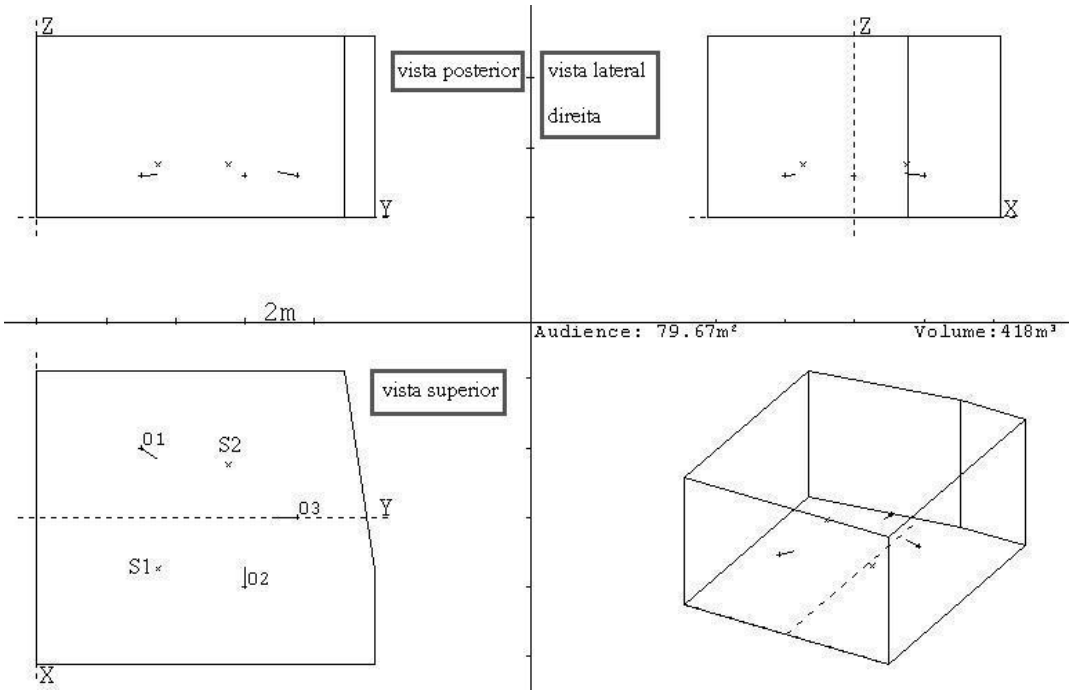

Figura 5.1: Vistas posterior, lateral direita e superior do estúdio de música do PTB [Bun02].

vértice	**x**	**y**	**z**
1	4,22	0,00	0,00
2	4,22	9,74	0,00
3	4,22	8,86	0,00
4	-4,22	0,00	0,00
5	1,54	9,74	0,00
6	4,22	0,00	5,23
7	4,22	9,74	5,23
8	-4,22	8,86	5,23
9	-4,22	0,00	5,23
10	1,54	9,74	5,23

Tabela 5.1: Descrição dos vértices usados na sala de referência do RR3.

plano	**vértice**	**vértice**	**vértice**	**vértice**	**vértice**
1 (piso)	1	2	5	3	4
2	4	9	6	1	
3	3	8	9	4	
4	5	10	8	3	
5	2	7	10	5	
6	1	6	7	2	
7 (teto)	9	8	10	7	6

Tabela 5.2: Descrição das paredes da sala de referência, em relação aos vértices dados na Tabela 5.1

T30	Tempo de Reverberação, em segundos
EDT	Tempo Inicial de Decaimento, em segundos
D50	Definição, em %
C80	Clareza, em dB

Tabela 5.3: Parâmetros acústicos calculados.

Usando as condições normais dadas, calculamos os parâmetros acústicos definidos na Tabela 5.3.

A fim de avaliar as características acústicas de um ambiente é necessário usar vários parâmetros acústicos, usados para descrever determinadas características acústicas do ambiente. Em particular, vamos calcular quatro parâmetros acústicos: Tempo de Reverberação (T30), Tempo Inicial de Decaimento (EDT), Definição (D50) e Clareza (C80).

O Tempo de Reverberação, T30, descreve o tempo necessário para que a energia sonora decresça 30 decibéis.

O Tempo Inicial de Decaimento (EDT) é o Tempo de Reverberação, porém medido somente até que a energia sonora decresça 10 decibéis. O decaimento inicial da energia sonora é muito importante para a impressão de reverberação. Este parâmetro consiste de um número relativamente pequeno de reflexões sonoras.

A Definição (D50) é dada pela razão entre a energia sonora inicial (primeiros 50 milisegundos do som) e energia sonora total (definida pela duração total do som).

A Clareza (C80) é calculada pela diferença (em decibéis) entre a energia sonora recebida pelo ouvinte nos primeiros 50 milisegundos e a energia sonora reverberante.

A definição de cada um destes parâmetros e a forma de calculá-los pode ser encontrada na norma ISO 3382 [Sta97]. Em nosso estudo, estes parâmetros são calculados para as posições dos receptores e fontes sonoras dadas na seção seguinte.

5.1.2 Emissor e Receptor Sonoros

De acordo com a norma ISO 3382, que trata de medições acústicas de salas, as fontes sonoras devem ser onidirecionais. Assim para todas as posições de fontes sonoras dadas na Tabela 5.4, usamos a fonte sonora do tipo `MonteCarloAcousticSource`; usamos também receptores onidirecionais, de raio fixo. Para avaliar o campo sonoro na sala, os parâmetros acústicos são medidos em 3 posições distintas com emissores em 2 posições distintas, descritos na Tabela 5.4.

rótulo	**x**	**y**	**z**
S1	1,5	3,5	1,5
S2	-1,5	5,5	1,5
R01	-2,0	3,0	1,2
R02	2,0	6,0	1,2

Tabela 5.4: S1 e S2 são os emissores sonoros, R01 e R02 são os receptores sonoros.

Com o ambiente de simulação definido de acordo com as subseções acima, realizamos as simulações acústicas em nossa implementação no ambiente AcMus. Descrevemos a seguir os resultados obtidos.

5.2 Resultados da Simulação

Efetuamos a simulação do estúdio de música do PTB, descrito na Seção 5.1, realizando uma execução do programa para cada banda de oitava para refletir o coeficiente de absorção do ar em cada uma das bandas. Usamos o fator de reflexão K=500 e geramos 100.000 raios. A Tabela 5.5 mostra os parâmetros acústicos resultantes de nossa simulação, comparando-os aos resultados obtidos pelos participantes do RR 3 fase 1. Inicialmente, analisamos os parâmetros acústicos obtidos ao fazer a simulação acústica para a fonte na posição S1 e o receptor sonoro na posição R01.

Software	**125 Hz**	**250 Hz**	**500 Hz**	**1000 Hz**	**2000 Hz**	**4000 Hz**
RR3	1,90±0,22	1,91±0,29	1,86±0,17	1,83±0,17	1,74±0,15	1,51±0,19
AcMus	1,95	1,92	1,85	1,68	1,82	1,56

Tabela 5.5: EDT: fonte S1 e receptor R1. Valores em segundos.

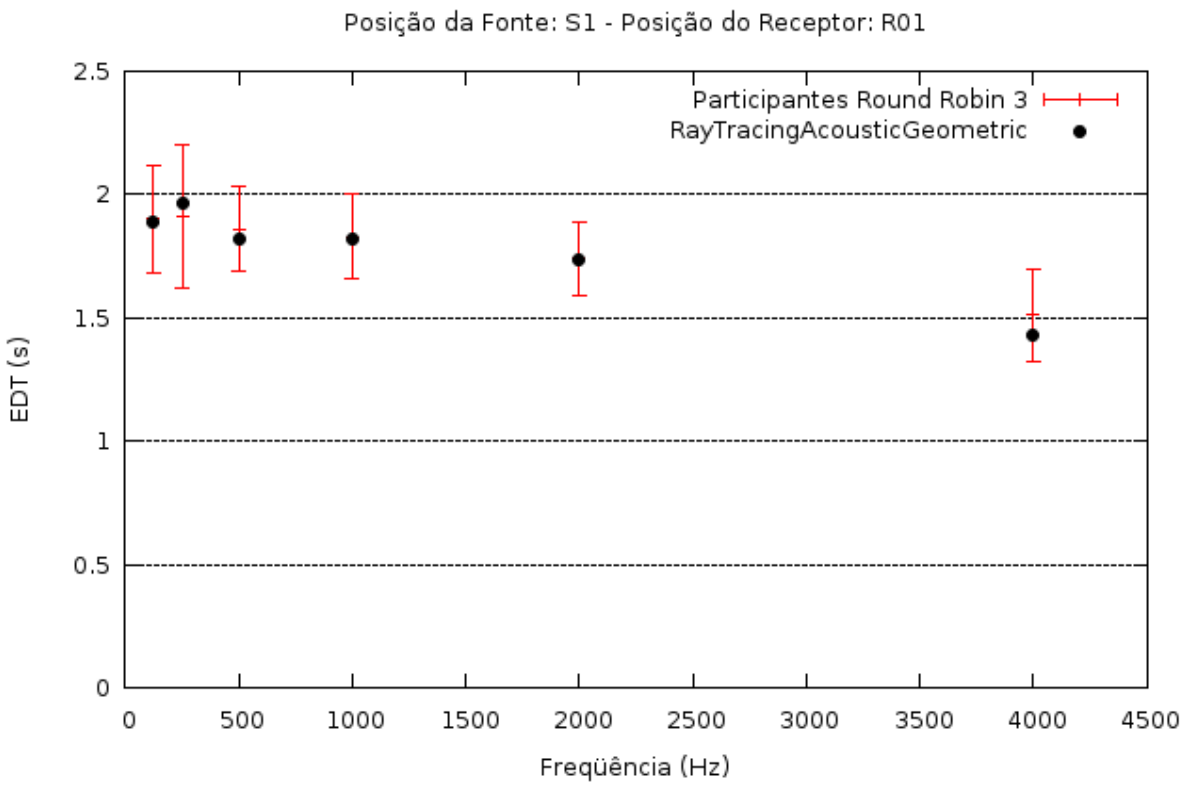

Figura 5.2: EDT obtido pelo AcMus marcado com pontos pretos, média e respectivo desvio padrão dos resultados obtidos pelos participantes do RR 3 marcado com a barra vertical.

O valor do parâmetro Tempo Inicial de Decaimento (EDT) obtido, ao simular a sala da fase 1 do RR 3 com o emissor na posição S1 e o receptor na posição R01 usando a nossa implementação `RayTracingGeometricAcousticSimulation`, foi em média igual aos valores obtidos pelos participantes do experimento. É interessante notar que o erro é maior nas primeiras bandas de oitava, o que pode ocorrer devido à abstração da acústica geométrica de salas não modelar bem frequências mais baixas. Já o erro nas bandas de oitava de 1000 e 2000 Hz é bem pequeno e é possível notar também que o valor obtido pelo AcMus coincide com a média dos valores obtidos pelos participantes do experimento.

A partir da Tabela 5.5 é possível construir a Figura 5.2, que mostra graficamente que o algoritmo `RayTracingGeometricAcousticSimulation` se comportou dentro do esperado, já que refletiu

os mesmos valores dos demais participantes do RR 3 fase 1.

Software	**125 Hz**	**250 Hz**	**500 Hz**	**1000 Hz**	**2000 Hz**	**4000 Hz**
RR3	1,83±0,19	1,83±0,22	1,82±0,18	1,79±0,17	1,70±0,15	1,48±0,16
AcMus	1,77	1,74	1,80	1,68	1,65	1,23

Tabela 5.6: T30: fonte S1 e receptor R1. Valores em segundos.

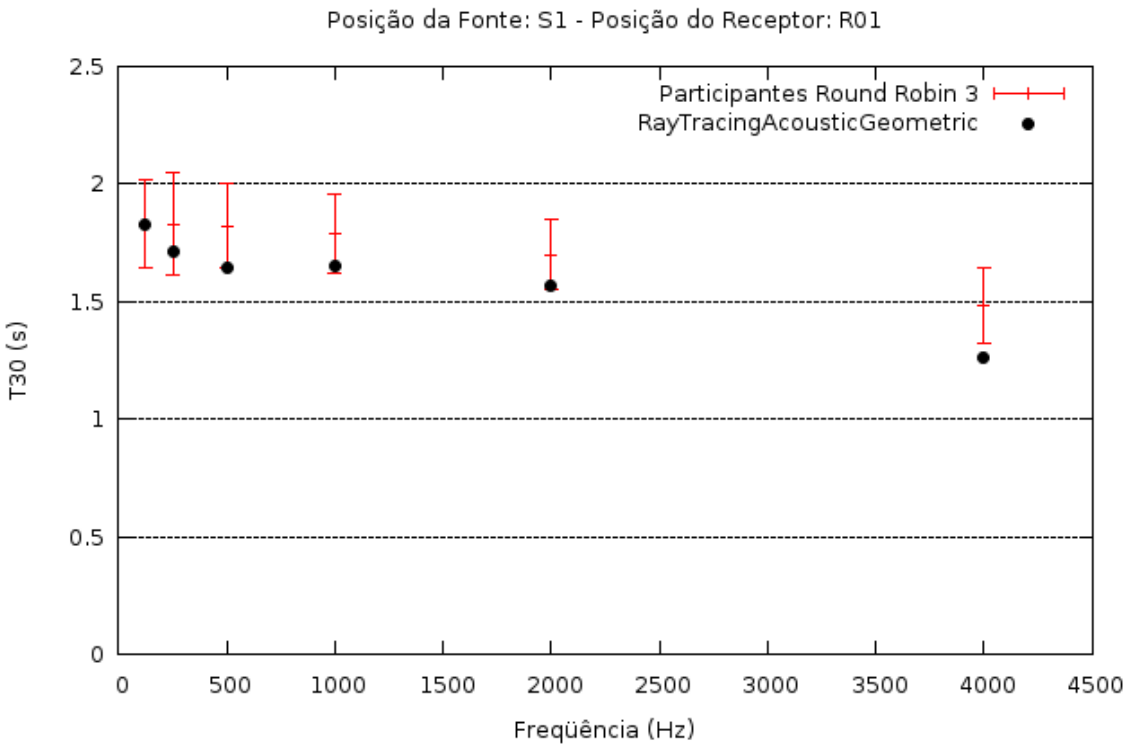

Figura 5.3: T30 obtidos pelo AcMus, marcado com pontos pretos, média e respectivo erro dos outros participantes.

Como explicado, o parâmetro acústico T30 é calculado a partir do tempo para ocorrer um decaimento de 30 dB na energia do som e é medido entre -5 dB e -35 dB. Os resultados experimentais podem ser vistos na Tabela 5.6. A partir da Tabela 5.6 criamos a Figura 5.3, que mostra que nossa simulação se encontra dentro da média e erro dos participantes do experimento, para as bandas de oitava de 125, 250, 500, 1000, 2000 Hz. O valor obtido por nossa implementação para esse parâmetro só se mostrou discrepante, em relação ao valor calculado pelos outros participantes, para a banda de oitava de 4000 Hz. Nessa banda o valor de T30 obtido por nossa implementação foi de 1,26 s, já a média desse parâmetro, calculado pelos participantes do experimento, foi de 1,48 s $\pm 0,16$s.

Para tentar elucidar o motivo de tal discrepância para o resultado obtido para este parâmetro, calcu-

lado em 4000 Hz, realizamos vários testes usando diferentes valores para o coeficiente de atenuação do ar (m), e notamos que o parâmetro T30 é sensível aos diferentes valores desse coeficiente. Porém ao variar o coeficiente m, o parâmetro T30 varia também em todas as outras bandas de oitava. Assim existe uma correlação entre o valor do coeficiente de atenuação do ar e o parâmetro T30, como era esperado, porém não conseguimos esclarecer o motivo do parâmetro T30 calculado pelo AcMus apresentar valores diferentes, para a banda de oitava de 4000 Hz, dos valores obtidos pelos participantes do experimento. Outra hipótese para a diferença dos resultados é, que cada programa tem sua forma de calcular o coeficiente de atenuação do ar m e o problema estaria na forma como calculamos esse coeficiente no AcMus.

Software	**125 Hz**	**250 Hz**	**500 Hz**	**1000 Hz**	**2000 Hz**	**4000 Hz**
RR3	33,85±3,06	33,21±5,35	34,78±2,48	34,48±2,48	36,06±2,82	40,45±3,44
AcMus	28,44	22,90	25,80	28,98	31,78	35,87

Tabela 5.7: D50: fonte S1 e receptor R1

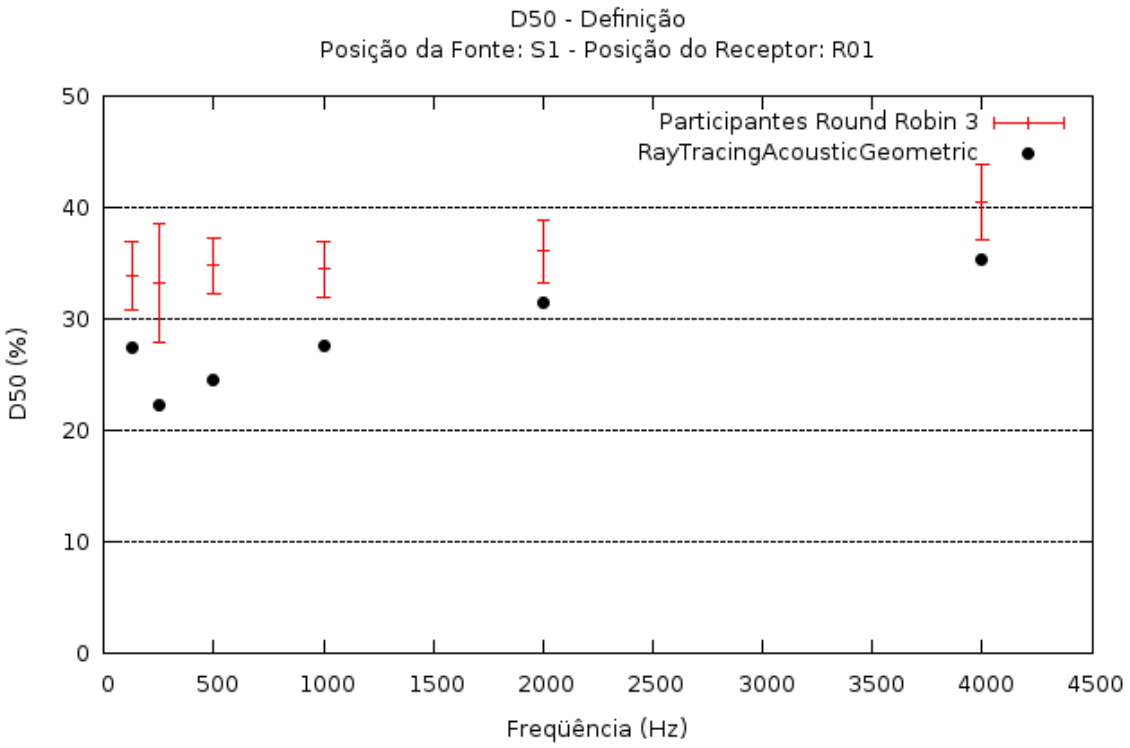

Figura 5.4: Parâmetro D50 calculado pelo AcMus e pelos participantes do RR 3.

O parâmetro D50, indica a definição do som que chega no receptor sonoro e é caracterizado principalmente pelos raios sonoros que chegam diretamente ao receptor. Dessa maneira, são avaliados a

relação entre a energia sonora dos primeiros 5 ms e a energia total que chega no receptor sonoro. Os valores obtidos por nossa simulação mostram-se deslocados para baixo, aproximadamente 7% a menos em cada banda de frequência.

Fizemos também a simulação acústica da modelo de sala da fase 1 do RR para a fonte sonora na posição S2 e o receptor sonoro na posição R02. Essa avaliação da Resposta Impulsiva em mais de uma posição da sala é necessária para avaliar o campo sonoro na sala, já que o campo apresenta características diferentes para diferentes posições de uma mesma sala. Os resultados obtidos pelos participantes do RR3 e pelo AcMus podem ser vistos na Tabela 5.7.

Software	**125 Hz**	**250 Hz**	**500 Hz**	**1000 Hz**	**2000 Hz**	**4000 Hz**
RR3	1,90±0,24	1,90±0,26	1,86±0,17	1,82±0,14	1,74±0,16	1,51±0,18
AcMus	1,91	1,92	1,95	1,82	1,53	

Tabela 5.8: EDT: fonte S2 e receptor R2. Valores em segundos

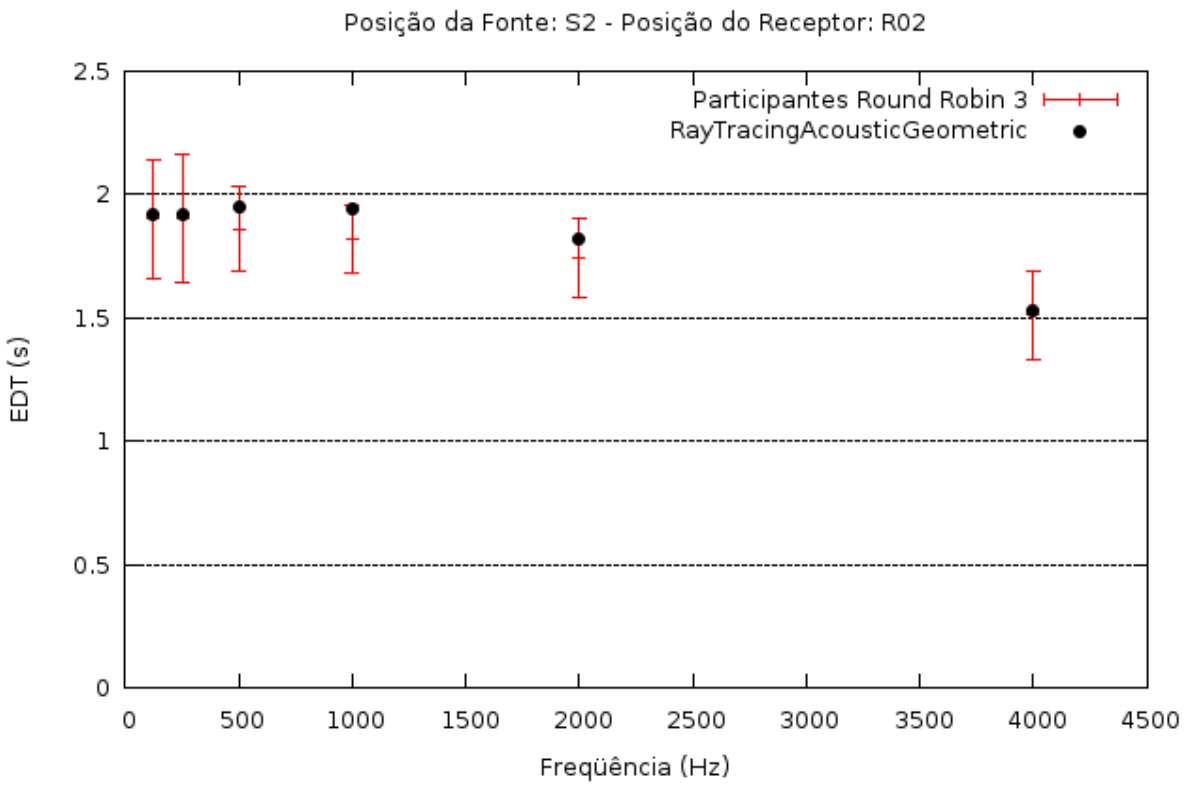

Figura 5.5: EDT calculado pelo AcMus e dos participantes do RR3.

É possível ver a partir da Tabela 5.8 e da Figura 5.5, que os valores calculados por nossa implementação são bem próximos dos valores calculados pelos programas participantes do RR3 fase 1. Nenhum dos

valores obtidos ficaram fora das margens de erro, o que indica que nossa implementação se comportou como os outros programas.

Software	**125 Hz**	**250 Hz**	**500 Hz**	**1000 Hz**	**2000 Hz**	**4000 Hz**
RR3	1,86±0,19	1,86±0,17	1,83±0,18	1,81±0,16	1,72±0,15	1,49±0,15
AcMus	1,89	1,79	1,70	1,64	1,60	1,21

Tabela 5.9: T30: fonte S2 e receptor R2

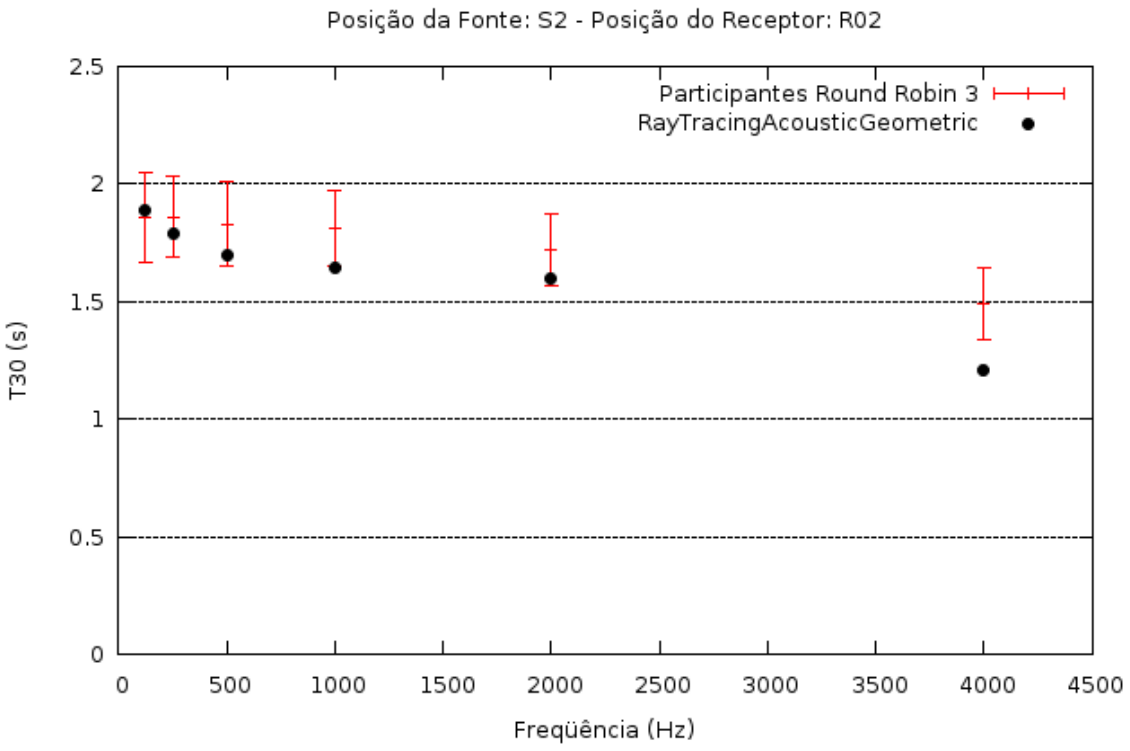

Figura 5.6: T30 calculado pelo AcMus e pelos participantes do RR 3.

O tempo de reverberação T30 calculado para nossa simulação ficou dentro da média e margem de erro dos outros programas para as primeiras bandas de oitava. Porém para a última banda de oitava usada, 4000 Hz, a nossa implementação apresentou um valor abaixo do esperado.

O último parâmetro acústico que mostraremos é o parâmetro D50 para a a fonte posicionada em S2 e para o receptor sonoro posicionado em R02. A Figura 5.7 sintetiza os valores obtidos para as bandas de oitava avaliadas no RR3, mostradas na Tabela 5.10.

Software	**125 Hz**	**250 Hz**	**500 Hz**	**1000 Hz**	**2000 Hz**	**4000 Hz**
RR3	34,42±3,49	33,97±2,79	33,98±3,30	34,69±2,66	35,93±2,45	40,09±3,06
AcMus	27,63	26,00	28,17	28,97	28,57	32,83

Tabela 5.10: D50: fonte S2 e receptor R2

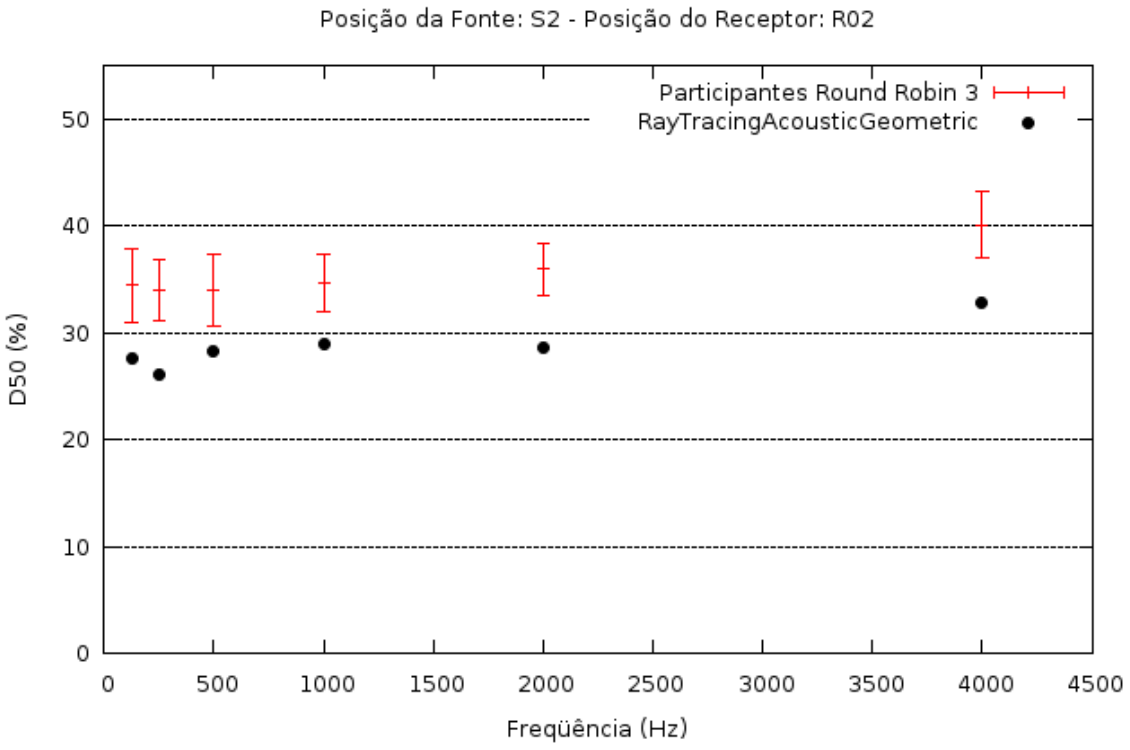

Figura 5.7: D50 calculado pelo AcMus e pelos participantes do RR 3.

Da Figura 5.7 e da Tabela 5.10 podemos ver que o parâmetro D50 calculado pelo AcMus apresenta um valor inferior aos valores de D50 calculados pelos participantes do RR 3.

5.3 Resumo

A nossa implementação do algoritmo de Traçado de Raios realizada na classe `RayTracingGeometricAcousticSimulation` gerou parâmetros acústicos semelhantes aos parâmetros calculados pelos participantes do RR 3. Isso indica que nossa implementação consegue obter parâmetros acústicos para descrever o comportamento do campo sonoro numa sala com precisão semelhante à dos outros participantes do RR 3 fase 1.

O único parâmetro que se mostrou com grandes problemas, foi a Definição (D50), pois tanto para a simulação com o emissor S1 e receptor R01, quanto para simulação com emissor S2 e receptor R02, nossa implementação apresentou valores abaixo dos obtidos pelos demais programas de simulação acústica.

Outra característica interessante é o tempo de processamento necessário para realizar a simulação acústica da sala do PTB na fase 1, pois como os algoritmos são implementados em Java, havia dúvidas em relação ao tempo necessário para concluir as simulações. A simulação para cada posição de fonte e receptor sonoros, levou em média 10 segundos, indicando que a linguagem Java é suficientemente eficiente para atingir os objetivos aqui propostos.

Capítulo 6

Conclusão

O principal objetivo deste trabalho foi pesquisar como funcionam os algoritmos de simulação acústica e, com esse conhecimento, construir o módulo de simulação acústica do ambiente AcMus a fim de disponibilizar uma ferramenta computacional livre que possa auxiliar nos projetos de salas para prática musical.

No início do trabalho, apresentamos os alguns algoritmos usados para a simulação acústica de salas. Apresentamos também, de forma detalhada, os passos necessários para a implementação de um algoritmo de traçado de raios, deixando claro que existe mais de uma forma para se construir tal algoritmo, como explicamos no Capítulo 3, Seção 3.4.

Já na segunda parte do trabalho, mostramos como foi feita a implementação do módulo de simulação acústica dentro do ambiente AcMus, levando em conta o objetivo de tornar tal módulo facilmente ampliável. Com este objetivo em mente, criamos uma arquitetura modular usando padrões de projeto tais como `Strategy` e `Template Method` e usamos também o padrão arquitetural de `Plug-ins` a fim de permitir a criação de componentes plugáveis ao ambiente, o que foi em grande parte facilitado pelo uso do ambiente Eclipse.

Vimos que é possível usar a linguagem Java para realizar a simulação acústica de salas usando inclusive equipamentos de baixo custo, como computadores domésticos e, obter, mesmo assim, os resultados da simulação acústica em pouco tempo. Ampliamos o ambiente AcMus tornando-o, a nosso saber, o único ambiente integrado para medição, simulação e análise acústicas. Este software poderá ser usado como base para novas pesquisas científicas, conduzidas por: engenheiros acústicos, interessados nas novas técnicas de medição e simulação acústica, ou experimentação de novos materiais; arquitetos que busquem fazer projetos integrados de ambientes, que, por exemplo, além de considerarem o conforto térmico, levem também em consideração o conforto acústico dos ambientes a serem construídos; cientistas da computação, que podem pesquisar novos algoritmos que melhorem a eficiência das simulações acústicas.

Mesmo tratando-se de um programa com interface gráfica amigável, com tutorial de uso incluído na

instalação padrão, entendemos que para fazer um bom uso do AcMus e para obter resultados plausíveis, é muito importante que o usuário tenha bons conhecimentos sobre acústica de salas, técnicas de medição, métodos de simulação acústica, etc. Existem variados fatores, como o correto posicionamento do emissor sonoro, no caso de uma simulação, a serem levados em conta quando se quer avaliar a qualidade acústica de uma sala. Somente pessoas com experiência são capazes de discernir quais fatores levar em conta ou não antes de usar o computador para realizar os cálculos desejados. O usuário deve saber avaliar os resultados obtidos, o que demanda um conhecimento prévio das características de cada parâmetro acústico e os problemas inerentes a cada um deles. Assim o conhecimento tanto da ferramenta computacional quanto de acústica de salas é essencial para a realização de simulações e medições acústicas plausíveis.

O ambiente AcMus é liberado sob a licença `GPL`, podendo ser obtido pela Web, através do sítio do projeto (`http://gsd.ime.usp.br/acmus`). É possível também, obter o código fonte diretamente do repositório de controle de versões `(Subversion)` através do sítio (`https://acmus.svn.sourceforge.net/svnroot/acmus`).

6.1 Trabalhos Futuros

O algoritmo de traçado de raios que implementamos não trabalha com reflexões difusas e sabemos que tal tipo de reflexão pode tornar a Resposta Impulsiva simulada mais próxima da Resposta Impulsiva medida de uma certa sala. Assim um trabalho interessante seria ampliá-lo para trabalhar com tais tipos de reflexões.

Acreditamos que resultados interessantes podem ser obtidos ao se fazer a medição da acústica de uma sala e usar os resultados dessa medição como parâmetros para uma simulação da acústica dessa mesma sala. Assim a medição poderia ser usada para ajustar e deixar a simulação acústica ainda mais fidedigna.

Descrevemos de modo sucinto algumas técnicas de simulação acústica que resolvem numericamente a equação de onda. Devido a grande capacidade de procedimento disponível nos computadores pessoais atuais e a melhorias teóricas em tais técnicas, achamos que um promissor trabalho futuro é a implementação desses tipos de simulação acústica dentro do ambiente AcMus.

Acreditamos que o algoritmo de traçado de raios implementado no AcMus pode ser melhorado se, a partir dele, for criado um método híbrido de simulação. Atualmente existem métodos híbridos para simulação acústica que apresentam bons resultados [TC06]. Assim um possível trabalho futuro seria compor o método de traçado de raios atualmente existente no AcMus com o método das fontes virtuais.

O formato de arquivo CLF *(Common Loudspeaker Format)* foi criado em Abril de 2005, por alguns produtores de software, como CATT-Acoustic e Odeon A/S dentre outros, com o objetivo de ser um formato aberto para que fabricantes de alto-falantes pudessem publicar as características de seus produtos para os usuários finais de tais equipamentos. Um trabalho técnico interessante seria permitir que o

AcMus lesse esse formato de arquivos e usasse os dados dos alto-falantes como se fossem fontes sonoras simuladas e, assim seria possível simular a acústica de auditórios com mais precisão, já que geralmente os auditórios contam com sistemas de alto-falantes.

Uma promissora área de pesquisa em Ciência da Computação é o uso das unidades de processamento gráfico *Graphical Processing Unit - GPU* para auxiliar na computação de algoritmos que demandam muito processamento, como o algoritmo de traçado de raios, por exemplo. Estes dispositivos além de apresentarem uma grande velocidade de processamento, são capazes de trabalhar com vários fluxos de execução paralelos, o que é muito útil para diversos algoritmos, novamente podendo beneficiar a execução de algoritmos de traçado de raios. Com a criação da plataforma Nvidia CUDA é possível criar, de maneira simples, algoritmos para serem executados na GPU [RO08]. Assim uma possível ampliação do AcMus poderia ser feita ao implementar seus algoritmos de simulação acústica nesta plataforma.

Referências Bibliográficas

[AB79] Jont B. Allen and David A. Berkley. Image method for efficiently simulating small-room acoustics. *Journal of the Acoustical Society of America*, 65(4):943–950, 1979.

[Ber06] Leo L. Beranek. Analysis of sabine and eyring equations and their application to concert hall audience and chair absorption. *The Journal of the Acoustical Society of America*, 120(3):1399–1410, 2006.

[BM58] G.E.P. Box and Mervin E. Muller. A note on the generation of random normal deviates. *Annals of Mathematical Statistics*, 29(2):610–611, 1958.

[BMR+96] Frank Buschmann, Regine Meunier, Hans Rohnert, Peter Sommerlad, Michael Stal, Peter Sommerlad, and Michael Stal. *Pattern-Oriented Software Architecture, Volume 1: A System of Patterns*. John Wiley & Sons, August 1996.

[Bor84] Jeffrey Borish. Extension of the image model to arbitrary polyhedra. *Journal of the Acoustical Society of America*, 75(6):1827–1836, June 1984.

[Bun02] Physikalisch-Technische Bundesanstalt. Data of the reference room. http://www.ptb.de/en/org/1/17/173/refraumkoordinaten2.htm, 2002. Acessado em Outubro de 2008.

[Elo05] David O. Elorza. Room Acoustics Modeling Using the Raytracing Method: Implementation and Evaluation. Master's thesis, University of Turku, 2005.

[FI05] Fabio L. Figueiredo and Fernando Iazzetta. Comparative study of measured acoustic parameters in concert halls in the city of São Paulo. *Proceedings of the 2005 International Congress and Exposition on Noise Control Engineering, Rio de Janeiro, Brazil.*, 2005.

[Fig05] Fabio L. Figueiredo. Parâmetros acústicos subjetivos: Critérios para a avaliação da qualidade acústica de salas de música. Master's thesis, Universidade de São Paulo, 2005.

[Fou07] Eclipse Foundation. The Eclipse Platform. *Online em http://www.eclipse.org. Visitado em Maio*, 2007.

[GBD07] Marcio H.A. Gomes, S.R. Bertoli, and J.G. Dedecca. Implementação de métodos para simulação acústica e auralização de salas. *Revista da Sociedade Brasileira de Acústica (SOBRAC)*, 2007.

[GHJV95] Erich Gamma, Richard Helm, Ralph Johnson, and John Vlissides. *Design patterns: elements of reusable object-oriented software*. Addison-Wesley Longman Publishing Co., Inc. Boston, MA, USA, 1995.

[Gil05] David Gilbert. JFreeChart Developer Guide, 2005.

[Gla89] Andrew S. Glassner. *An Introduction to Ray Tracing*. Morgan Kaufmann, 1989.

[Guo05] Jackwind L. Guojie. *Professional Java native interfaces with SWT*. John Wiley & Sons, 2005.

[GVG04] Marcio H.A. Gomes, M. Vorländer, and SNY Gerges. Measurement and use of scattering coefficients in room acoustic computer simulations. In *IV Congreso Ibero-Americano de Acústica, European Acoustics Symposium*, Guimarães (Portugal), 2004.

[IFM04] Fernando Iazzetta, Fabio L. Figueiredo, and Bruno S. Masiero. Parâmetros Subjetivos em Salas Destinadas à Prática Musical. In *IV Congreso Ibero-Americano de Acústica, European Acoustics Symposium*, Guimarães (Portugal), 2004.

[IKdQ+04] Fernando Iazzetta, Fabio Kon, Marcelo G. de Queiroz, F.S.C. da Silva, and Marcio de Avelar Gomes. AcMus: Computational Tools for Measurement, Analysis and Simulation of Room Acoustics. *Proceedings of the European Acoustics Symposium*, 2004.

[IKS01] Fernando Iazzetta, Fabio Kon, and Flavio S. C. Silva. ACMUS: Design and Simulation of Musical Listening Environments. In *Proceedings of the 8th Brazilian Symposium on Computer Music*, Fortaleza, Brazil, August 2001.

[İLA07] Mustafa E. İLAL. The quest for integrated design system: a brief survey of past and current efforts. *METU JFA*, page 2, 2007.

[Inc03] Nagata Acoustics Inc. Nagata acoustics news. http://www.nagata.co.jp/news/news0312-e.htm, 2003. Acessado em Outubro de 2008.

[JQ07] Zhongjin Jiang and Xiaojun Qiu. Receiving radius determination in ray-tracing sound prediction of rectangular enclosure. *Journal of Sound and Vibration*, 301(1):391–399, Mar. 2007.

[Kul85] Andrzej Kulowski. Algorithmic representation of the ray tracing technique. *Applied Acoustics*, 18(6):449–469, 1985.

[Kut00] Heinri Kuttruff. *Room Acoustics*. Spon Press (UK), 2000.

[Leh92] Hilmar Lehnert. Systematic errors of the ray-tracing algorithm. *Applied Acoustics*, 38:207–221, 1992.

[Lew93] T. Lewers. A combined beam tracing and radiant exchange computer model of room acoustics. *Applied Acoustics*, 38:161–178, 1993.

[LK97] A.M. Law and W.D. Kelton. *Simulation Modeling and Analysis*. McGraw-Hill Higher Education, 1997.

[LNY00] B. M. Shield L. N. Yang. Development of a ray tracing computer model for the prediction of the sound field in long enclosures. *Journal of Sound and Vibration*, 229(1):133–146, Jan. 2000.

[Mas04] Bruno Sanches Masiero. Métodos de medição de resposta impulsiva em salas de pequeno porte. estudo e implementação de métodos de medição de resposta impulsiva em salas de pequeno porte. Technical report, Universidade de São Paulo, 2004.

[QIK+08] Marcelo Queiroz, Fernando Iazzetta, Fabio Kon, Márcio H. A. Gomes, Fábio L. Figueiredo, Bruno Masiero, Leo K. Ueda, Luciana Dias, Mário H. C. Torres, and Leandro F. Thomaz. Acmus: an open, integrated platform for room acoustic research. *Journal of the Brazilian Computer Society*, 3(14):87–103, 2008.

[RGL08] Nikunj Raghuvanshi, Nico Galoppo, and Ming C. Lin. Accelerated wave-based acoustics simulation. In *SPM '08: Proceedings of the 2008 ACM symposium on Solid and physical modeling*, pages 91–102, New York, NY, USA, 2008. ACM.

[RGN06] Jens H. Rindel, A.C. Gade, and M.L. Nielsen. The virtual reconstruction of the ancient Roman concert hall in Aphrodisias, Turkey. *The Sixth International Conference on Auditorium Acoustics*, pages 316–323, 2006.

[Rin00] Jens H. Rindel. The use of computer modelling in room acoustics. *Journal of Vibroengineering*, 3(4):41–72, 2000.

[Rin02] Jens H. Rindel. Modelling in auditorium accoustics - from the ripple tank and scale models to computer simulation. In *The 3rd EAA European Congress on Acoustics. Forum Acusticum, Sevilla*, 2002.

[RO08] Antonio Rueda and Lidia Ortega. Geometric algorithms on cuda. In *International Conference on Computer Graphics Theory and Applications*, pages 107–112, 2008.

[Sav99] Lauri Savioja. *Modeling Techniques for Virtual Acoustics*. PhD thesis, Helsinki University of Technology, 1999.

[Sch08] Andreas Schmidt. Room acoustics round robin. *http://www.ptb.de/en/org/1/17/173/roundrobin.htm, acessado em Outubro*, 2008.

[SL01] Lauri Savioja and Tapio Lokki. Digital Waveguide Mesh for Room Acoustic Modelling. *ACM SIGGRAPH and Eurographics Campfire: Acoustic Rendering for Virtual Environments. Utah, USA*, 2001.

[Sta97] ISO Standard. 3382. Acoustics–Measurement of the reverberation time of rooms with reference to other acoustical parameters. *International Standards Organization*, 1997.

[Sta08] Richard Stallman. The free software definition. *Online em http://www.fsf.org/licensing/essays/free-sw.html. Visitado em Maio*, 2008.

[TC06] Roberto Aizik Tenenbaum and Thiago Schaaf Camilo. Método híbrido para simulação numérica de acústica de salas: Teoria, implantação computacional e validação experimental. *Acústica Vibrações, Rio de Janeiro*, 37:3–14, 2006.

[UFIK05] Leo K. Ueda, Fabio L. Figueiredo, Fernando Iazzetta, and Fabio Kon. A user-friendly graphical system for room acoustics measurement and analysis. In *Anais do 10.o Simpósio Brasileiro de Computação Musical. Belo Horizonte*, pages 118–129, 2005.

[Vor00] M. Vorländer. Room acoustical simulation algorithm based on the free path distribution. *Journal of Sound and Vibration*, 232(1):129–137, Apr. 2000.

[Wei08] Eric W. Weisstein. Sphere point picking. *OnLine em http://mathworld.wolfram.com/SpherePointPicking.html. Visitado em Agosto*, 2008.

[XKJ03] Zeng Xiangyang, Chen Ke'an, and Sun Jincai. On the accuracy of the ray-tracing algorithms based on various sound receiver models. *Applied Acoustics*, 64(4):433–441, Apr. 2003.