



AKADEMIA GÓRNICZO-HUTNICZA

im. Stanisława Staszica w Krakowie

**WYDZIAŁ INŻYNIERII
MECHANICZNEJ I ROBOTYKI**

Praca dyplomowa inżynierska

Marcin Misiurewicz

Imię i nazwisko

Inżynieria Akustyczna

Kierunek studiów

**Interaktywne narzędzie do auralizacji
pomieszczenia wykorzystujące nagrania
ambisoniczne**

Temat pracy dyplomowej

dr. inż. Adam Pilch

Promotor Pracy

.....

Ocena

Kraków, rok 2017/2018

Kraków, 25 grudnia 2017

Imię i nazwisko: Marcin Misiurewicz
Nr albumu: 279254
Kierunek studiów: Inżynieria Akustyczna
Profil dyplomowania: Nazwa specjalności

OŚWIADCZENIE

Uprzedzony o odpowiedzialności karnej na podstawie art. 115 ust 1 i 2 ustawy z dnia 4 lutego 1994 r. o prawie autorskim i prawach pokrewnych (tj. Dz. U. z 2006 r. Nr 90, poz. 631 z późn. zm.): „Kto przywłaszcza sobie autorstwo albo wprowadza w błąd co do autorstwa całości lub części cudzego utworu albo artystycznego wykonania, podlega grzywnie, karze ograniczenia wolności albo pozbawienia wolności do lat 3. Tej samej karze podlega, kto rozpowszechnia bez podania nazwiska lub pseudonimu twórcy cudzy utwór w wersji oryginalnej albo w postaci opracowania, artystyczne wykonanie albo publicznie zniekształca taki utwór, artystyczne wykonanie, fonogram, wideogram lub nadanie”, a także uprzedzony o odpowiedzialności dyscyplinarnej na podstawie art. 211 ust. 1 ustawy z dnia 27 lipca 2005 r. Prawo o szkolnictwie wyższym (tj. Dz. U. z 2012 r. poz. 572, z późn. zm.) „Za naruszenie przepisów obowiązujących w uczelni oraz za czyny uchybiające godności student ponosi odpowiedzialność dyscyplinarną przed komisją dyscyplinarną albo przed sądem koleżeńskim samorządu studenckiego, zwanym dalej „sądem koleżeńskim”, oświadczam, że niniejszą pracę dyplomową wykonałem osobiście i samodzielnie i że nie korzystałem ze źródeł innych niż wymienione w pracy”.

.....
podpis dyplomanta

Kraków, 25 grudnia 2017

Imię i nazwisko: Marcin Misiurewicz
Nr. albumu: 279254
Kierunek studiów: Inżynieria Akustyczna
Specjalność: Nazwa specjalności

OŚWIADCZENIE

Świadomy odpowiedzialności karnej za poświadczanie nieprawdy oświadczam, że niniejszą inżynierską pracę dyplomową wykonałem osobiście i samodzielnie oraz nie korzystałem ze źródeł innych niż wymienione w pracy.

Jednocześnie oświadczam, że dokumentacja pracy nie narusza praw autorskich w rozumieniu ustawy z dnia 4 lutego 1994 roku o prawie autorskim i prawach pokrewnych (Dz. U. z 2006 r. Nr 90 poz. 631 z późniejszymi zmianami) oraz dóbr osobistych chronionych prawem cywilnym. Nie zawiera ona również danych i informacji, które używałem w sposób niedozwolony. Wersja dokumentacji dołączona przeze mnie na nośniku elektronicznym jest w pełni zgodna z wydrukiem przedstawionym do recenzji.

Zaświadczam także, że niniejsza inżynierska praca dyplomowa nie była wcześniej podstawą żadnej innej urzędowej procedury związanej z nadawaniem dyplomów wyższej uczelni lub tytułów zawodowych.

.....

podpis dyplomanta

Kraków, 25 grudnia 2017

Imię i nazwisko:	Marcin Misiurewicz
Adres korespondencyjny:	Adres
Temat pracy dyplomowej inżynierskiej:	Interaktywne narzędzie do auralizacji pomieszczenia wykorzystujące nagrania ambisoniczne
Rok ukończenia:	2018
Nr. albumu:	279254
Kierunek studiów:	Inżynieria Akustyczna
Specjalność:	Nazwa specjalności

OŚWIADCZENIE

Niniejszym oświadczam, że zachowując moje prawa autorskie, udzielam Akademii Górniczo-Hutniczej im. S. Staszica w Krakowie nieograniczonej w czasie nieodpłatnej licencji niewyłącznej do korzystania z przedstawionej dokumentacji inżynierskiej pracy dyplomowej, w zakresie publicznego udostępniania i rozpowszechniania w wersji drukowanej i elektronicznej¹. Publikacja ta może nastąpić po ewentualnym zgłoszeniu do ochrony prawnej wynalazków, wzorów użytkowych, wzorów przemysłowych będących wynikiem pracy inżynierskiej².

Kraków,

data podpis dyplomanta

¹Na podstawie Ustawy z dnia 27 lipca 2005 r. o prawie o szkolnictwie wyższym (Dz.U. z 2005 Nr 164, poz. 1365) Art. 239 oraz Ustawy z dnia 4 lutego 1994 r. o prawie autorskim i prawach pokrewnych (Dz.U. 2000 r. Nr 80, poz. 904, z późn. zm.) Art. 15a: "Uczelni w rozumieniu przepisów o szkolnictwie wyższym przysługuje pierwszeństwo w opublikowaniu pracy dyplomowej studenta. Jeżeli uczelnia nie opublikowała pracy dyplomowej w ciągu 6 miesięcy od jej obrony, student, który ją przygotował, może ją opublikować, chyba że praca dyplomowa jest częścią utworu zbiorowego."

²Ustawa z dnia 30 czerwca 2000 r. – prawo własności przemysłowej (Dz.U. z 2003 r. Nr 119, poz. 1117, z późn. zm.), a także rozporządzenie Prezesa Rady Ministrów z dnia 17 września 2001 r. w sprawie dokonywania i rozpatrywania zgłoszeń wynalazków i wzorów użytkowych (Dz.U. z 2001 r. Nr 102, poz. 1119 oraz z 2005 r. Nr 109, poz. 910).

Kraków, 25 grudnia 2017

**AKADEMIA GÓRNICZO-HUTNICZA
WYDZIAŁ INŻYNIERII MECHANICZNEJ I ROBOTYKI**

TEMATYKA PRACY DYPLOMOWEJ INŻYNIERSKIEJ
dla studenta IV roku studiów stacjonarnych

Marcin Misiurewicz
imię i nazwisko studenta

TEMAT PRACY DYPLOMOWEJ INŻYNIERSKIEJ: Interaktywne narzędzie do auralizacji pomieszczenia wykorzystujące nagrania ambisoniczne

Promotor pracy: dr. inż. Adam Pilch

Recenzent pracy: dr. hab. inż. Tadeusz Kamisiński, prof. AGH
podpis dziekana

PLAN PRACY DYPLOMOWEJ:

1. Omówienie tematu pracy i sposobu realizacji z promotorem.
2. Zebranie i opracowanie literatury dotyczącej tematu pracy.
3. Zebranie i opracowanie wyników badań.
4. Analiza wyników badań, ich omówienie i zatwierdzenie przez promotora.
5. Opracowanie redakcyjne.

Kraków,
data podpis dyplomanta

TERMIN ODDANIA DO DZIEKANATU:**20...****r.**

.....
podpis promotora

Kraków, 25 grudnia 2017

Akademia Górniczo-Hutnicza im. Stanisława Staszica

Wydział Inżynierii Mechanicznej i Robotyki

Kierunek: Inżynieria Akustyczna

Profil dyplomowania: Nazwa specjalności

Marcin Misiurewicz

Praca dyplomowa inżynierska

Interaktywne narzędzie do auralizacji pomieszczenia wykorzystujące nagrania ambisoniczne

Opiekun: dr. inż. Adam Pilch

STRESZCZENIE

[Treść streszczenia]

Cracow, December 25, 2017

AGH University of Science and Technology

Faculty of Mechanical Engineering and Robotics

Field of Study: Acoustic Engineering

Specialisations: Specialisation

Marcin Misiurewicz

Engineer Diploma Thesis

Interactive tool for room acoustics auralization using ambisonic recordings

Supervisor: dr. inż. Adam Pilch

SUMMARY

[Summary text]

*Serdecznie dziękuję . . . tu ciąg dalszych
podziękowań np. dla promotora, żony,
sąsiada itp.*

Spis treści

1. Wstęp	10
1.1. Zastosowanie dźwięku przestrzennego w symulatorach i grach	10
1.2. Cel i założenia pracy	11
2. Auralizacja	13
2.1. Otrzymywanie odpowiedzi impulsowej pomieszczenia	13
2.2. Odsłuch	16
2.3. Przekształcanie formatów ambisonicznych w format binauralny	16
3. Filtracja	19
3.1. Filtracja jako mnożenie widm	19
3.2. Algorytmy sekcjonowanego szybkiego splotu sygnałów dyskretnych	19
3.3. Odwrotne filtry Kirkeby’ego	20
4. Pomiary i Implementacja	21
4.1. Wybór rozwiązań i narzędzi	21
4.2. Pomiary	22
4.3. Obsługa programu	23
4.4. Działanie programu	26
4.4.1. Zamiana harmonik sferycznych	26
4.4.2. Zamiana formatu ambisonicznego na binauralny	28
4.4.3. Filtracja ścieżki muzycznej	28
5. Testy programu	30
5.1. Analiza czasowa	30
5.2. Testy odsłuchowe	30
6. Zakończenie	31
6.1. Wnioski	31
6.2. Dalsze możliwe prace	31

1. Wstęp

Dynamiczny rozwój rzeczywistości wirtualnej (VR) spowodował większe zainteresowanie technologią auralizacji. Początkowo przemysł VR skupił się przede wszystkim na stworzeniu realistycznych wrażeń wzrokowych, lecz jego dalszy rozwój wyznaczył kolejne cele, przede wszystkim wiarygodnych wrażeń słuchowych, gdyż człowiek potrafi „oceniać położenie źródła [dźwięku] we wszystkich kierunkach wokół obserwatora”. [1] Użytkownik miał nie tylko móc widzieć świat trójwymiarowo, ale także słyszeć dźwięk przestrzenny zgodnie z położeniem jego wirtualnych uszu. Już w roku 1962 Morton Heilig w swojej maszynie Sensorama, która miała dawać użytkownikowi realistyczne wrażenia jazdy rowerem po Brooklinie, pośród narzędzi takich jak trójwymiarowy obraz stereoskopowy i przechylenie się siedzenia użytkownika, zastosował dźwięk stereo. Wprowadzenie na komercyjny rynek mikrofonów binauralnych i ambisonicznych spopularyzowało tak zwany „dźwięk 3D”, lecz nadal taka metoda wymagała wcześniejszych nagrań w rzeczywistej przestrzeni. Dopiero nagły wzrost możliwości obliczeniowych komputerów umożliwił symulację pola akustycznego, eliminując wymóg wcześniejszych nagrań. Dzięki temu użytkownik miał akustyczne wrażenie znajdowania się w przestrzeni, która istnieje wyłącznie wirtualnie

W pierwszym rozdziale opisano wybrane, zrealizowane dotychczas prace na polu symulacji wirtualnej akustycznej rzeczywistości a także cel niniejszej pracy. W rozdziale drugim przedstawiono teorię dotyczącą auralizacji, metody otrzymywania ambisonicznej odpowiedzi impulsowej przestrzeni, metody odsłuchu dźwięku przestrzennego i technikę przetwarzania ścieżki ambisonicznej na binauralną. W trzecim skupiono się na podstawowych zagadnieniach z zakresu filtracji w czasie rzeczywistym. W czwartym rozdziale uzasadniono wybór rozwiązań problemów z poprzedniego rozdziału i opisano implementację programu. W piątym opisano wyniki testów czasu obliczeń i doświadczenia użytkownika.

1.1. Zastosowanie dźwięku przestrzennego w symulatorach i grach

Jednym z najbardziej rozpoznawalnych przykładów zastosowania dźwięku przestrzennego jest nagranie „Virtual Barber Shop”, gdzie użytkownik ma wrażenie siedzenia na fotelu fryzjerskim. Otrzymane dzięki wykorzystaniu mikrofonu binauralnego (więcej

o mikrofonie binauralnym w rozdziale 2.1) jest wyjątkowo realistyczne. Słuchacz może dokładnie ustalić lokalizację źródeł dźwięku – kranu, pod którym fryzjer myje ręce, dzwoniącego telefonu, czy drzwi do salonu. Mimo tego, jest to jedynie nagranie – słuchacz nie ma możliwości interakcji.

Istnieją także krótkie gry na systemy iOS i Android, gdzie gracz musi realizować zadania mając do dyspozycji wyłącznie dźwięk. Przykładem może być gra A Blind Legend, w której gracz wciela się w postać niewidomego rycerza. Jednak dźwięk przestrzenny jest tu wykorzystany wyłącznie jako binauralne nagrania poszczególnych efektów, takich jak świst miecza, i pomimo możliwości interakcji gracza z wirtualnym światem, lokalizacja wirtualnych źródeł dźwięku jest niezależna od położenia i obrotu głowy postaci i nie zawsze dokładnie określona. Przez to gracz może poruszać się wyłącznie dzięki wskazówkom innych postaci.

Ciekawy eksperyment przeprowadził Google LLC we współpracy z Abbey Road Studios tworząc wirtualny spacer po jednym z najslawniejszych studiów nagraniowych na świecie. Użytkownik może poruszać się po całym studio i rozglądać się bez ograniczeń. Dodatkowo podchodząc do określonych pozycji może wyświetlać informacje o poszczególnych częściach studio i jego historii a w reżyserce może wirtualnie skorzystać z uproszczonego stołu mikserskiego. Użytkownik poruszając się po studio doświadcza także binauralnego dźwięku. Silnik spaceru reaguje na obroty głowy w płaszczyźnie horyzontalnej i zmianę położenia użytkownika dostosowując do nich dźwięk. Google wykorzystuje w tym celu Web Audio, ustawiając w wirtualnej przestrzeni źródła w konkretnych pozycjach i przetwarzając otrzymany obraz na ścieżkę binauralną. Jest to możliwe dzięki temu, że spacer odbywa się w studio nagraniowym, a więc przestrzeni, w której czas pogłosu jest bardzo krótki a akustyka pomieszczenia nie jest znacząco zależna od pozycji słuchacza. Silnik nie musi więc symulować całego pola akustycznego, a tylko przetwarzać dźwięk z kilku źródeł w określonych pozycjach i przetwarzać go w czasie rzeczywistym (co wynika z możliwości interakcji użytkownika) na ścieżkę binauralną.

1.2. Cel i założenia pracy

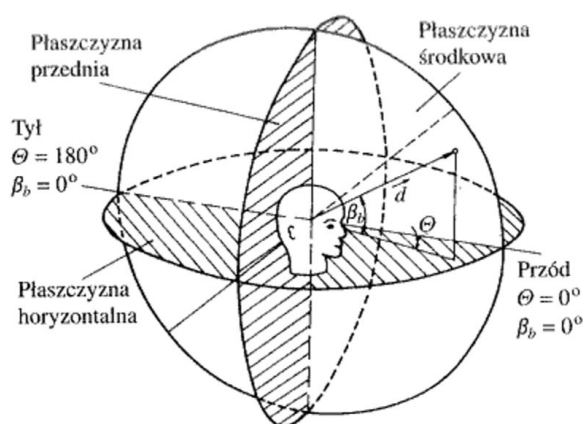
W tej pracy podjęto się realizacji wirtualnego, akustycznego spaceru, gdzie użytkownik może doświadczyć możliwie najbardziej rzeczywistej i pełnej akustyki konkretnej przestrzeni. Program, w założeniach, ma umożliwić użytkownikowi wirtualne poruszanie się po sali, w każdym punkcie obrót wirtualnej głowy w trzech osiach i odtwarzanie dźwięku, który byłby słyszany przez niego w rzeczywistości w tej pozycji. Założono więc przede wszystkim:

- **Możliwość poruszania się po sali**

W przestrzeni o zróżnicowanej akustyce odpowiedź impulsowa w różnych punktach pomiarowych może się znacznie różnić – w przypadku sali wykładowej słuchacz będzie miał zupełnie różne doświadczenia, jeśli usiądzie blisko mówcy na środku, albo pod ścianą tuż koło ustroju wytłumiającego. Aby doświadczyć pełnej akustyki przestrzeni, użytkownik musi mieć możliwość zmiany pozycji.

– **Możliwość obrotu głowy w trzech płaszczyznach**

Aby zapewnić użytkownikowi rzeczywiste doznania i swobodę weryfikowania akustyki program powinien zapewnić możliwość obrotu wirtualnej głowy w płaszczyźnie horyzontalnej, środkowej i przedniej.



Rys. 1.1. Płaszczyzny lokalizacji źródła dźwięku [1]

Zdecydowano, że w płaszczyźnie horyzontalnej powinien być możliwy pełen obrót, natomiast w pozostałych dwóch płaszczyznach nie jest on wymagany - program nie musi zapewnić możliwości ruszania głową do pozycji nienaturalnych np. odchył głowy o 90st do tyłu.

– **Możliwie najbardziej realistyczne wrażenia akustyczne**

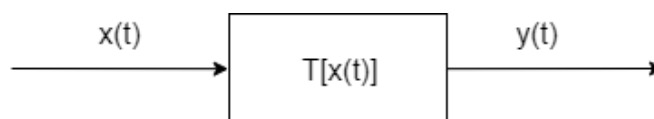
Program ma jak najwierniej odwzorowywać odpowiedź przestrzeni na zadane wymuszenie i przekazywać ją użytkownikowi w sposób jak najbardziej realistyczny, tak, aby użytkownik mógł dokładnie lokalizować źródło mając poczucie obecności w pomieszczeniu.

– **Odsłuch w czasie rzeczywistym przy niewielkim czasie reakcji programu na zmianę parametrów pozycji i obrotu** Aby zapewnić użytkownikowi ciągłość doświadczenia, program, po rozpoczęciu działania, powinien na bieżąco reagować na zmianę parametrów wejściowych – zmiany pozycji i kątów obrotu – w możliwie niewielkim czasie i płynnie dopasowując do nich dźwięk przekazywany użytkownikowi.

2. Auralizacja

2.1. Otrzymywanie odpowiedzi impulsowej pomieszczenia

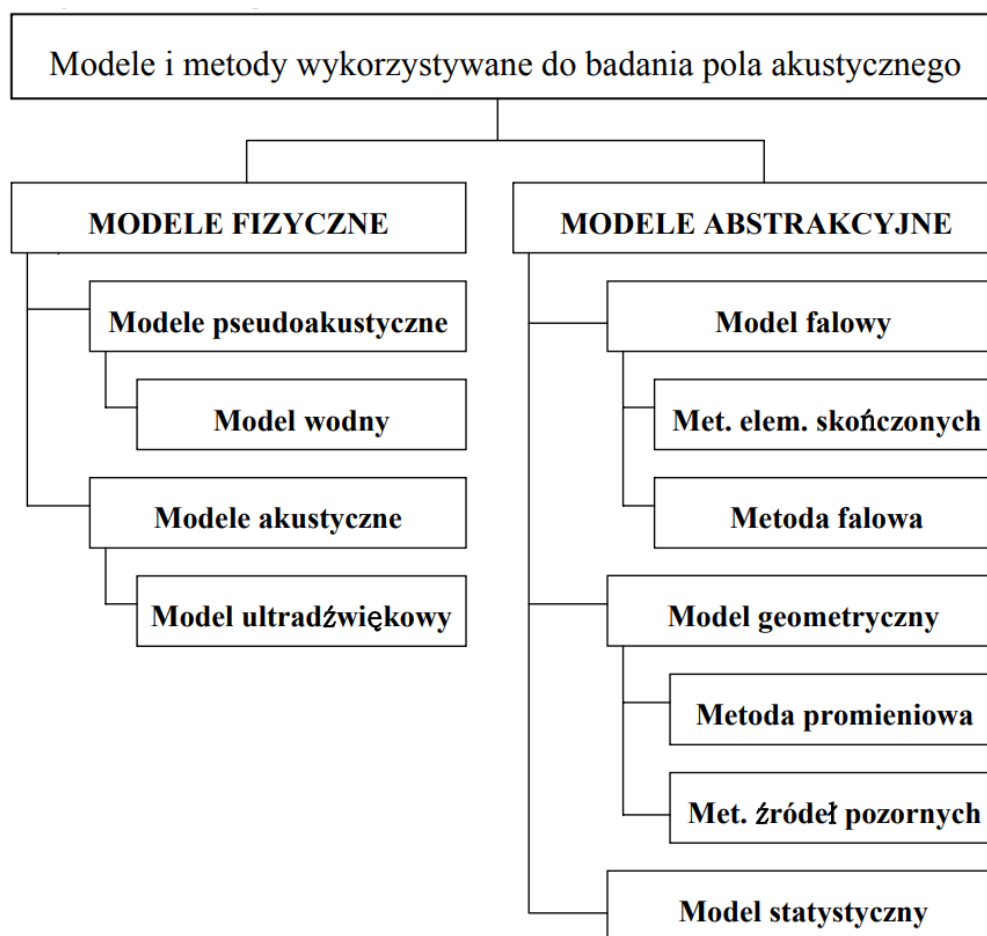
Ścieżka dźwiękowa $y(t)$, odpowiadająca wrażeniom słuchowym w danym punkcie przestrzeni podczas konkretnego pobudzenia (np. stanie na środku sali koncertowej podczas występu skrzypka), to odpowiedź $T[x(t)]$ układu, czyli przestrzeni widzianej z punktu słuchacza, na dane wymuszenie $x(t)$ [2], co pokazuje rysunek 2.1.



Rys. 2.1. Schemat działania filtra

Aby zatem, znając wymuszenie $x(t)$, wyznaczyć ścieżkę $y(t)$ należy poznać odpowiedź $T(t)$ układu, nazywaną odpowiedzią impulsową, czyli odpowiedzi układu na pobudzenie go *impulsem Diraca* [3]. Odpowiedź impulsową przestrzeni w danym punkcie można wyznaczyć jedną z metod modelowania akustycznego, przedstawionych na rysunku 2.2 [4], lub po prostu nagrywając ją w rzeczywistości.

Programy komputerowe do wyznaczania akustyki pomieszczeń, w tym do wyznaczania odpowiedzi impulsowej, takie jak CATT Acoustics, bazują przede wszystkim na metodzie śledzenia promieni. Metoda ta polega symulacji źródła, jako źródła promieni rozchodzących się wszechkierunkowo i analizy tych promieni w różnych chwilach czasowych. Promienie w momencie generacji mają tę samą energię. Odbicia są rozpatrywane zgodnie z prawem Snella, a zgodnie z zadaniem współczynnikiem pochłaniania powierzchni odbijającej maleje energia promieni. Odbiornik jest umieszczany w dowolnym, zadanym punkcie w przestrzeni. Ponieważ prawdopodobieństwo natrafienia przez promień na odbiornik punktowy jest bliskie zeru, jest to w rzeczywistości sfera o niewielkich rozmiarach. Program zbiera informacje o energii i kierunku promieni przechodzących przez odbiornik i na ich podstawie oblicza różne parametry akustyczne, w tym odpowiedź impulsową.



Rys. 2.2. Modele i metody wykorzystywane do badania pola akustycznego

Modelowanie układu metodami abstrakcyjnymi ma tę zaletę, że można wyznaczać odpowiedzi impulsowe układów, które istnieją wyłącznie w rzeczywistości wirtualnej, co może mieć przeważające znacznie podczas tworzenia gier komputerowych. Dzięki tym metodom możliwe jest również uzyskanie odpowiedzi układu w dowolnej ilości punktów, ograniczonej wyłącznie mocą obliczeniową komputera. Z uwagi na dużą złożoność obliczeniową, czas potrzebny na uzyskanie odpowiedzi impulsowej pomieszczenia jest relatywnie niewielki, ale nawet przy minimalizacji ilości promieni, oraz niewielkich rozmiarów pomieszczenia wyklucza zastosowanie tej metody w czasie rzeczywistym.

Nagrywanie odpowiedzi impulsowej jest metodą prostszą, ale jej specyfika wymaga omówienia kilku zagadnień. Nagranie zwykłym mikrofonem jest związane z utratą informacji o kierunku pochodzenia dźwięku, która jest kluczowa dla stworzenia realistycznego efektu dźwięku przestrzennego. Człowiek lokalizuje źródło dźwięku dzięki percepcji binauralnej (dokonywana za pomocą obu uszu), na którą składają się Międzyuszna Różnica Czasu (MRC) i Międzyuszna Różnica Natężeń (MRN) [1] i percepcji monouralnej (dokonywanej jednusznie) - modyfikacji widma dźwięku, głównie przez małżowinę uszną, różnej dla różnych kątów we wszystkich płaszczyznach. Funkcja, która opisuje taką zmianę widma jest nazywana „Head Related Transfer Function” (HRTF), a jej odwrotna transformata „Head Related Impulse Response” (HRIR) i jest indywidualna dla każdego człowieka. [5] MRC i MRN tworzy się poprzez binauralną różnicę dróg dotarcia dźwięku do jednego i drugiego ucha – np. dźwięk idący z lewej strony głowy najpierw dochodzi do lewego ucha i potrzebuje ok. $650 [\mu s]$ więcej aby dotrzeć do prawego ucha, a głowa tworzy cień akustyczny, wobec czego natężenie dźwięku przy prawym uchu będzie mniejsze [Ozimek].

Zastosowanie specjalnego mikrofonu binauralnego zachowałoby informacje o kierunku dochodzenia dźwięku do mikrofonu od razu w formie gotowej do odtworzenia użytkownikowi. Taki mikrofon to urządzenie składające się z makiety małżowiny usznej (czasami również głowy wraz z częścią korpusu) i zamontowanych w niej niewielkich mikrofonów. Dzięki takiej budowie mikrofon jest w stanie zapisać cenne, z punktu widzenia percepcji dźwięku przez człowieka, informacje, takie jak MRC, MRN i HRTF. Taki mikrofon nagrywa w formacie binauralnym - dwóch ścieżkach odpowiadającym dźwiękowi przekazywanych na każde z uszu.

Innym rodzajem mikrofonu, który zachowa informacje o kierunku dochodzenia dźwięku jest rodzina mikrofonów ambisonicznych. Takie mikrofonu składają się z pewnej liczby kapsułów. W pracy skupiono się na mikrofonach 4-kapsułowych (tzw. czworościennych) z racji ich największej popularności. Większość mikrofonów ambisonicznych nagrywa w tzw. A-formacie, czyli 4 ścieżkach pochodzących z poszczególnych kapsułów rozłożonych na bokach czworościanu foremego. Format ten nie jest formatem docelowym, służy natomiast do przekształcenia go w B-format. Przekształcenie takie odbywa się w teorii wyłącznie dzięki prostym operacjom arytmetycznym, w praktyce jednak należy

jeszcze zastosować odpowiednią filtrację [6]. B-format to 4 ścieżki odpowiadające 4 pierwszym harmonikom sferycznym: $Y_0^0, Y_1^{-1}, Y_1^0, Y_1^1$. Trzy ścieżki odpowiadają dźwiękowi rozchodzącemu się na trzech osiach: X, Y, i Z i niosą informację o kierunku dźwięku, a czwarta ścieżka zbiera informacje o fazie, a więc o zwrocie dźwięku. Więcej o zależnościach matematycznych w B-formacie napisano w rozdziale 2.3. B-format jest formatem niezwykle użytecznym podczas przetwarzania dźwięku przestrzennego – niesie w sobie pełną informację o kierunku dźwięku padającego na mikrofon, informacja ta jest łatwo dostępna i łatwa do przekształcania, w tym przetwarzaniu na inne formaty.

2.2. Odsłuch

Odsłuch ścieżek dźwiękowych niosących w sobie informacje przestrzenne jest ściśle związany z ich formatem. Zasadniczo są dwie możliwości: za pomocą słuchawek (format binauralny), lub za pomocą macierzy głośników (różne formaty *surround*). Odtwarzanie dźwięku przy pomocy słuchawek jest pożądane głównie przy realizacji gier komputerowych, gdzie dźwięk jest dedykowany wyłącznie dla jednej osoby – gracza, a więc przy użytkowaniu indywidualnym. Formatów *surround* używa się zwykle do innych celów np. w przemyśle filmowym. Niewątpliwą zaletą formatu binauralnego jest jej uniwersalność sprzętowa – słuchawki są produktem bardzo rozpowszechnionym i relatywnie tanim. Technologia *surround* wymaga ściśle określonego i zgodnego z formatem dźwięku zestawu głośnikowego (np. dla formatu 5.1 – pięciu głośników szerokopasmowych i jednego niskotonowego). Jej przewagą jest jednak uniwersalność względem użytkownika – format binauralny jest nagrywany przy użyciu ustandaryzowanej małżowiny (czasem także sztucznej głowy, lub przy pomocy głowy jakiejś osoby), a więc MRC, MRN i HRTF nie będzie idealnie dopasowany do indywidualnego użytkownika. Lokalizowanie wirtualnego źródła przez słuchacza będzie obarczone większym błędem. Technologia *surround* tworzy rzeczywiste pole akustyczne wokół słuchacza, więc ten problem nie zachodzi.

2.3. Przekształcanie formatów ambisonicznych w format binauralny

B-format jest formatem ambisonicznym pierwszego rzędu, czyli poszczególne ścieżki odpowiadają harmonikom sferycznym do pierwszego rzędu. Nie jest on użyteczny dla końcowego użytkownika, ale może być wygodnym środkiem do otrzymania formatu *surround* lub binauralnego. M. A. Gerzon opisał poniższą metodę konwersji B-formatu na format binauralny [Gerzon, M. A., “Ambisonic in multichannel broadcasting and video”, in J. Audio Eng. Soc., vol. 33, pp. 859-871 (1985)], natomiast Kotaro i Sonoda przybliżyli tę metodę w bardziej przystępny sposób [Kotaro Sonoda B-format for binaural listening of

higher order Ambisonics], skorzystano więc z tego opisu: zakładając, że cyfrowy sygnał binauralny $b(n)$ jest przedstawiony jako:

$$b(n) = \begin{bmatrix} b_l(n) \\ b_r(n) \end{bmatrix}, \quad (2.1)$$

gdzie $b_l(n)$ to sygnał dla lewego ucha, a $b_r(n)$ dla prawego. Nagrany sygnał ambisoniczny $a(n)$ składa się z 4 ścieżek:

$$a(n) = \begin{bmatrix} a_W(n) & a_X(n) & a_Y(n) & a_Z(n) \end{bmatrix}^T, \quad (2.2)$$

Określa się $a_W(n)$ jako ścieżkę odpowiadającą harmonikom sferycznym zerowego rzędu (mikrofon wszechkierunkowy), natomiast $a_{X,Y,Z}(n)$ jako ścieżki odpowiadające harmonikom sferycznym pierwszego rzędu (mikrofony kierunkowe na osiach X,Y,Z). Reprezentację sygnałów $b(n)$ i $a(n)$ w dziedzinie częstotliwości oznaczono jako $\mathbf{B}(\omega)$ i $\mathbf{A}\omega$. Macierz harmonik sferycznych \mathbf{Y} zdefiniowano jako:

$$\mathbf{Y}(\theta_0, \dots, \theta_{S-1}, \phi_0, \dots, \phi_{S-1}) = \begin{bmatrix} Y_0^0(\theta_0, \phi_0) & Y_0^0(\theta_1, \phi_1) & \dots & Y_0^0(\theta_{S-1}, \phi_{S-1}) \\ Y_1^{-1}(\theta_0, \phi_0) & Y_1^{-1}(\theta_1, \phi_1) & \dots & Y_1^{-1}(\theta_{S-1}, \phi_{S-1}) \\ Y_1^1(\theta_0, \phi_0) & Y_1^1(\theta_1, \phi_1) & \dots & Y_1^1(\theta_{S-1}, \phi_{S-1}) \\ Y_1^0(\theta_0, \phi_0) & Y_1^0(\theta_1, \phi_1) & \dots & Y_1^0(\theta_{S-1}, \phi_{S-1}) \end{bmatrix}, \quad (2.3)$$

gdzie S to rozmiar bazy HRTF (ilość wirtualnych głośników), a $Y_l^m(\theta, \phi)$ to harmonika sferyczna stopnia l i rzędu m , zdefiniowana jako [mcQuarrie, Tom II]:

$$Y_l^m(\theta, \phi) = \left[\frac{(2l+1)}{4\pi} \frac{(l-|m|)!}{l+|m|!} \right]^{1/2} P_l^{|m|}(\cos\theta) e^{im\phi}, \quad (2.4)$$

gdzie $P_l^{|m|}(x)$ to Stowarzyszona funkcja Legendre'a stopnia l i rzędu m . Korzystając z 2.4, separując część rzeczywistą od urojonej i wykorzystując tylko tę pierwszą otrzymano pierwsze 4 harmoniki sferyczne wykorzystane w 2.3:

$$Y_0^0 = \frac{1}{\sqrt{4\pi}} \quad (2.5)$$

$$Y_1^{-1}(\theta, \phi) = \sqrt{\frac{3}{8\pi}} \cdot \sin\theta \cdot \cos\phi \quad (2.6)$$

$$Y_1^1(\theta, \phi) = -\sqrt{\frac{3}{8\pi}} \cdot \sin\theta \cdot \sin\phi \quad (2.7)$$

$$Y_1^0(\theta) = \sqrt{\frac{3}{4\pi}} \cdot \cos\theta \quad (2.8)$$

Przekształcenie formatów za [Kotaro Sonoda] otrzymuje się według:

$$\mathbf{B}(\omega) = \mathbf{H} \cdot \mathbf{Y}^+ \cdot \mathbf{A}(\omega), \quad (2.9)$$

gdzie:

$$\mathbf{H}(\theta_0, \dots, \theta_{S-1}, \phi_0, \dots, \phi_{S-1}) = \begin{bmatrix} H_l(\theta_0, \phi_0, \omega) & H_l(\theta_1, \phi_1, \omega) & \dots & H_l(\theta_{S-1}, \phi_{S-1}, \omega) \\ H_r(\theta_0, \phi_0, \omega) & H_r(\theta_1, \phi_1, \omega) & \dots & H_r(\theta_{S-1}, \phi_{S-1}, \omega) \end{bmatrix}, \quad (2.10)$$

gdzie \mathbf{Y}^+ oznacza pseudoodwrotność macierzy \mathbf{Y} a $H_l(\theta_i, \phi_i, \omega)$ to HRTF lewego ucha dla i -tego wirtualnego głośnika znajdującego się w pozycji (θ_i, ϕ_i) . $H_r(\theta_i, \phi_i, \omega)$ to odpowiednio HRTF dla prawego ucha.

3. Filtracja

3.1. Filtracja jako mnożenie widm

Zagadnieniem nieodłącznie związanym z przekształceniami dźwięku jest filtracja, czyli spłot sygnału $x(t)$ z odpowiedzią impulsową filtru $h(\tau)$. Operacja spłotu sygnałów w dziedzinie czasu jest równoważna operacji mnożenia ich widm w dziedzinie częstotliwości, a więc rozpatrując sygnały cyfrowe [3]:

$$h(\tau)x(t - \tau)d(\tau) \Leftrightarrow X(j\omega)H(j\omega) \quad (3.1)$$

A więc, rozważając sygnały cyfrowe, zamiast wykonywania operacji spłotu można przenieść sygnał i odpowiedź impulsową filtru do dziedziny częstotliwości wykorzystując Szybką Transformatę Fouriera (FFT), wymnożyć oba widma i przenieść wynik z powrotem do dziedziny czasu wykorzystując Odwrotną Szybką Transformatę Fouriera (iFFT). [3] Warto rozważyć złożoność obliczeniową obu operacji. Złożoność zarówno FFT jak i iFFT to $O(N\log_2 N)$, gdzie N to liczba próbek sygnału. Zakładając sygnał o 8192 próbkach (przy próbkowaniu 44100 [Hz] odpowiada to ok., 0,2[s]) i filtr o 512 współczynnikach, złożoność obliczeniowa operacji spłotu to 8192 · 512 operacji pomnóż i dodaj (multiply and accumulate – MAC), a więc 8192 · 512 · 2 = 8.388.608 operacji. Dla porównania FFT sygnałów, mnożenie i iFFT to tylko 217.600 operacji. Dla długich sygnałów mnożenie po stronie widmowej jest zdecydowanie bardziej opłacalne z punktu widzenia czasu procesora.

3.2. Algorytmy sekcjonowanego szybkiego spłotu sygnałów dyskretnych

Filtracja w czasie rzeczywistym stwarza nowe wyzwania. Czas procesora jest niezwykle cenny, a program nie może powodować dużego opóźnienia pomiędzy otrzymaniem próbek z bufora wejściowego, lub zmianą parametrów filtru (np. współczynników), a wypuszczeniem próbek do bufora wyjściowego. Najprostszą i najbardziej optymalną metodą byłaby filtracja całego sygnału, bez dzielenia go na bloki. Niestety, program nie ma do

dyspozycji całości sygnału, a tylko próbki, które do tej pory przeszły przez bufor wejściowy. Powoduje to konieczność filtrowania próbek wychodzących z bufora na bieżąco. Taka operacja nosi nazwę sekcjonowanego szybkiego splotu sygnałów dyskretnych [2].

Jedną metodą realizacji takiego splotu to algorytm *overlap-add* (w polskiej literaturze czasem nazywane metodą sumowania z nakładaniem). W takim algorytmie każdy blok jest filtrowany oddzielnie przy pomocy filtru o skończonej odpowiedzi impulsowej (FIR) a następnie odpowiednio dopasowywany do reszty sygnału wyjściowego. Aby przekazać na bufor wyjściowy użyteczny blok N próbek nie wystarczy pobrać z bufora wejściowego blok N próbek i je przefiltrować, gdyż wtedy otrzymuje się próbki z tzw. stanu przejściowego. Zakładając bufor o długości L , odpowiedź filtru o długości P , i to, że długość sygnału jest zdecydowanie większa od P długość wyniku splotu $N * P$ to $L + P - 1$. Ponieważ długość bloku to L , każdy wynik splotu będzie nachodził o $P - 1$ próbek na kolejny blok i tę nachodzącą końcówkę należy dodać do kolejnego obliczonego bloku. (rysunek 8.23 oppenheim albo zieleński 13.15) [2]

Drugą metodą jest algorytm *overlap-save* (w polskiej literaturze czasem nazywane metodą sumowania bez nakładania). W tej metodzie splata się blok o długości L z odpowiedzią impulsową o długości P . Ponieważ pierwsze $P - 1$ próbek jest nieużyteczne (przez stan przejściowy), do bufora wyjściowego przekazywane jest $L - (P - 1)$ próbek zaczynając od $(P-1)$ -tej próbki wyniku splotu. Reszta próbek jest odrzucana. [2]

3.3. Odwrotne filtry Kirkeby'ego

Przy pomiarze odpowiedzi impulsowej danej przestrzeni na nagranie ma wpływ charakterystyka głośnika i charakterystyka słuchacza. Może to znacząco zniekształcić odpowiedź impulsową, a co za tym idzie znacząco zniekształcić akustykę pomieszczenia i sprawić, że użytkownik będzie miał trudności ze zlokalizowaniem wirtualnego źródła dźwięku. Aby ten wpływ wyeliminować można zastosować odwrotne filtry Kirkeby'ego zdefiniowane jako [7]:

$$C(f) = \frac{Conj[H(f)]}{Conj[H(f) \cdot H(f) + \varepsilon(f)]} \quad (3.2)$$

gdzie:

$H(f)$ to impuls zmierzony w kontrolowanych warunkach bez wpływu otoczenia - np. komorze bezechowej

$\varepsilon(f)$ - niewielki, zależny od częstotliwości, parametr regulujący

Tak zaprojektowanym filtrem można później usunąć wpływ charakterystyki mikrofonu i głośnika z nagrań odpowiedzi impulsowej otoczenia.

4. Pomiary i Implementacja

4.1. Wybór rozwiązań i narzędzi

Modelowanie akustyki sali metodami abstrakcyjnymi wymaga implementacji bardzo trudnych w realizacji algorytmów. Problem stwarza też dokładność wyniku – nawet dokładny algorytm i skomplikowany model nie odda perfekcyjnie akustyki pomieszczenia. Nagrywanie rzeczywistej odpowiedzi niesie natomiast za sobą konieczność wykonania wielu pomiarów i kwantyzacji przestrzeni. Pomimo tych wad, do realizacji celów niniejszej pracy, wybrano metodę otrzymania odpowiedzi impulsowych drogą nagrywania. Jako przestrzeń akustycznego spaceru wybrano salę filharmonii Krakowskiej, z uwagi na zróżnicowaną akustykę i dostępne nagrania ambisoniczne w różnych punktach sali.

Użycie mikrofonu binauralnego do realizacji celu niniejszej pracy pociągnęłoby za sobą ogromną liczbę nagrań. Edycja ścieżki binauralnej, tak aby „obrócić” nagranie o zadany kąt jest nieopłacalne obliczeniowo w stosunku do innych rozwiązań (takich jak nagrania mikrofonem ambisonicznym), więc należałoby zrealizować nagrania pod każdym dostępnym użytkownikowi kątem obrotu. Zakładając pomieszczenie 10m x 10m, kwantyzację przestrzeni co metr i 180 dostępnych dla użytkownika pozycji głowy otrzymuje się 1800 pomiarów. Realizacja takiej liczby pomiarów jest w praktyce bardzo czasochłonna i trudna w realizacji.

Zdecydowano się więc użyć do pomiarów mikrofonu ambisonicznego. Powoduje konieczność przetwarzania w czasie rzeczywistym B-formatu na format odpowiedni dla użytkownika, ale zmniejsza znacząco liczbę potrzebnych pomiarów. Żeby uzyskać wyniki użyteczne do realizacji celu postawionego w niniejszej pracy należy nagrać tylko kilka odpowiedzi impulsowych, w różnych punktach zadanej przestrzeni, a obroty wirtualnej głowy będą realizowane wewnątrz programu. Użytkownik będzie miał możliwość odsłuchu wyłącznie w punktach nagrań, a więc ilość wykonanych pomiarów warunkuje rozdzielczość siatki po której porusza się użytkownik.

Z uwagi na duży czas pogłosu sali filharmonii nagrania odpowiedzi impulsowych były bardzo długie – należało uwzględnić przynajmniej 2,5[s] nagrania, a więc około $1,1 \cdot 10^4$ próbek. Stworzyło to problem złożoności obliczeniowej podczas pracy w czasie rzeczywistym. Aby obliczyć jedną próbkę wyjściową, w klasycznym układzie działającym

w czasie rzeczywistym potrzebne byłoby wykonanie $1, 1 \cdot 10^4$ operacji MAC, a więc $2, 2 \cdot 10^4$ operacji podstawowych na jeden kanał. Ponieważ format binauralny ma dwa kanały – lewy i prawy – procesor musiałby poświęcić $4, 4 \cdot 10^4$ operacji na obliczenie jednej próbki wyjściowej. Zakładając próbkowanie 44100[Hz], procesor musiałby wykonać około $2 \cdot 10^{10}$ obliczeń na sekundę, co zdecydowanie przewyższa możliwości normalnych komputerów. Aby skrócić czas potrzebny na filtrację zdecydowano się więc stworzyć bufor wejściowy odpowiedniej długości i filtrować bloki wychodzące z bufora metodą mnożenia w dziedzinie częstotliwości, przy użyciu jednego z algorytmów sekcjonowanego szybkiego splotu sygnałów dyskretnych. Złożoność obliczeniowa obu metod – *overlap-add* i *overlap-save* jest podobna, więc zdecydowano się użyć metody *overlap-save* jako prostszej w implementacji. Zakładając przekazywanie do bufora wyjściowego bloków o długości 2^{14} , algorytm wymaga liczenia splotu na $1, 1 \cdot 10^4 + 2^{14}$ próbkach a więc złożoność obliczeniowa to około $1 \cdot 10^7$ operacji na sekundę.

4.2. Pomiary

Pomiary sali zostały zrealizowane przez [zespół!!!], przy użyciu mikrofonu Soundfield [model!!!] i źródła [model!!!].

[Tor pomiarowy?]

Aby wyeliminować wpływ charakterystyki źródła i mikrofonu zaprojektowano i użyto odwrotne filtry Kirkeby’ego. Aby tego dokonać, przeprowadzono pomiary mikrofonu Soundfield i źródła [model!!!] w komorze bezechowej. Na głośnik podano sinus przebiegu, a odebrany sygnał z mikrofonu przetworzono na odpowiedź impulsową przy użyciu programu [!!!]. Na jej podstawie za pomocą gotowego skryptu, wykorzystującego zależność [wzorek na kirkeby-ego!!!] w środowisku MATLAB zaprojektowano filtr FIR, ustawiając następujące parametry filtru odwrotnego: [parametry filtru!!!]. Impuls przed i po filtracji przedstawiono na rysunku:

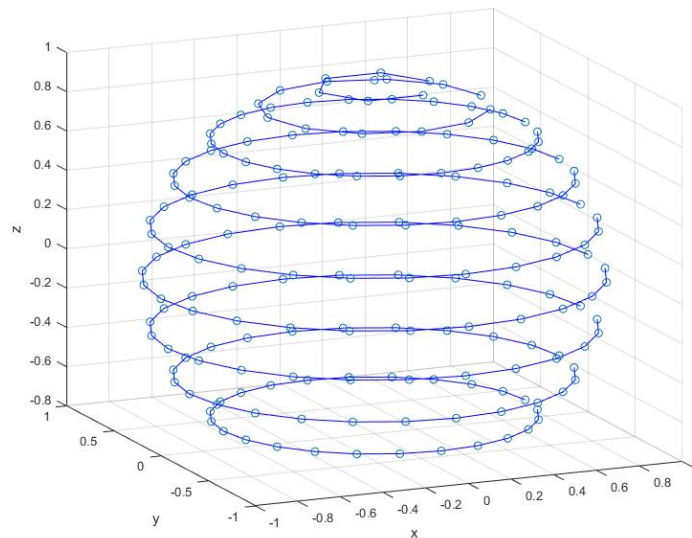
[wykres impulsu przefiltrowanego (a) i nieprzefiltrowanego (b)]

Wynik uznano za wystarczający i użyto filtru do usunięcia niepożądanych składowych z pomiarów odpowiedzi filharmonii.

Do implementacji użyto środowiska MATLAB, jako wysokopoziomowego, dobrze znanego autorowi pracy i wygodnego do implementacji zagadnień matematycznych i z zakresu cyfrowego przetwarzania sygnałów. Biblioteki tego środowiska mają wbudowane gotowe funkcje przydatne podczas realizacji niniejszej pracy, takie jak `fftfillt` – funkcja dokonująca splotu jako iloczynu widm sygnałów. Pomimo tego, że MATLAB nie jest wygodny do tworzenia Graficznego Interfejsu Użytkownika (GUI), zdecydowano się na to środowisko, gdyż planowano wyłącznie podstawowy interfejs. [zdanie do poprawy]

W pracy użyto bazy HRIR stworzonej podczas realizacji projektu „Listen” który jest częścią programu Komisji Europejskiej „Information Society Technologies”(IST) w

ramach „Fifth Framework Program”. Baza zawiera 187 impulsów nagrywanych dousznymi mikrofonami binauralnymi założonych przez konkretną osobę (bez wykorzystania sztucznej głowy). Posługując się definicjami kątów jak z rysunku ??, baza zawiera po 24 impulsy dla kątów $\theta = \{-45^\circ, -30^\circ, -15^\circ, 0^\circ, 15^\circ, 30^\circ, 45^\circ\}$ (grupy impulsów dla danych kątów θ z tego zbioru nazwano w pracy *głównymi pierścieniami*), 12 impulsy dla kąta $\theta = 60^\circ$, 6 impulsów dla kąta $\theta = 60^\circ$ (grupy impulsów dla danych kątów θ z tego zbioru nazwano w pracy *górnymi pierścieniami*) i 1 impuls dla kąta $\theta = 90^\circ$. Próbkowanie bazy HRTF pokazano na rysunku 4.1.



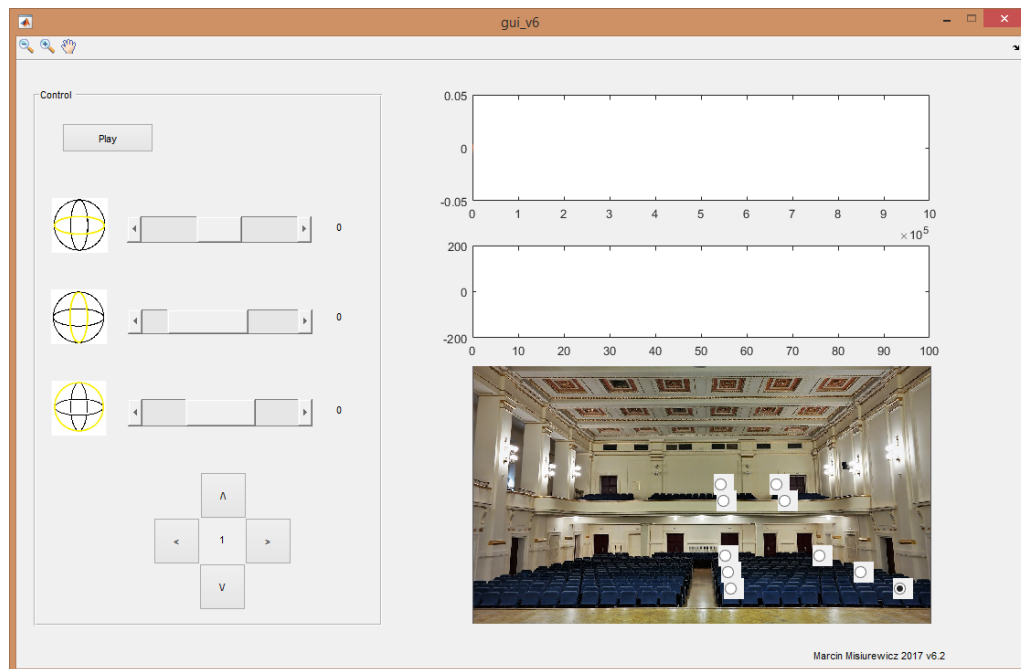
Rys. 4.1. Próbkowanie bazy HRTF - poszczególne pierścienie

4.3. Obsługa programu

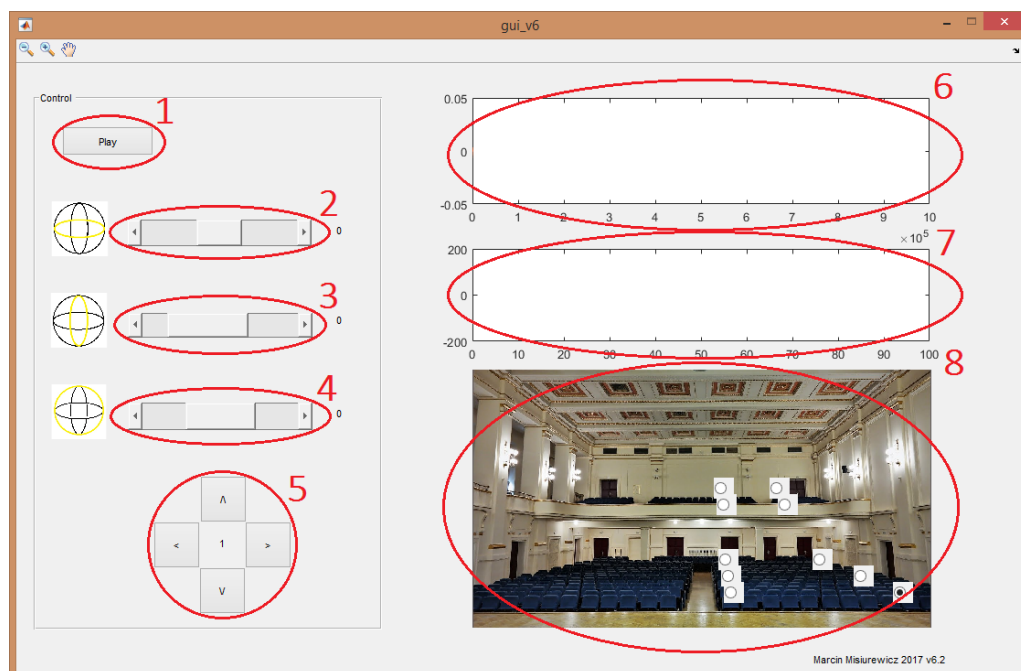
Poprzez Graphical User Interface Development Enviroment (GUIDE) zaprojektowano interfejs użytkownika. Przedstawiono go na rysunku 4.2.

W interfejsie znajdują się następujące pola:

1. Przycisk *Play*
2. Suwak obrotu głowy w płaszczyźnie horyzontalnej
3. Suwak obrotu głowy w płaszczyźnie środkowej
4. Suwak obrotu głowy w płaszczyźnie przedniej
5. Przyciski nawigacji po sali
6. Wykres przebiegu czasowego potoku wyjściowego



Rys. 4.2. Interfejs użytkownika



Rys. 4.3. Pola na interfejsie użytkownika

7. Wykres zmian kąta obrotu wirtualnej głowy w płaszczyźnie horyzontalnej
8. Wizualizacja wirtualnej pozycji użytkownika w sali
9. Przyciski nawigacji po wykresach - od lewej oddalenie, przybliżenie, przesunięcie

Format wyjściowy programu to dźwięk binauralny, więc jest zaprojektowany do odtwarzania przy użyciu słuchawek stereo. Ważne jest więc, aby użytkownik upewnił czy są poprawnie założone (lewa - prawa). Aby uruchomić narzędzie należy użyć przycisku *Play* (1). Od tej chwili program wypuszcza na kartę dźwiękową potok próbek. Używając suwaków (2)-(4) użytkownik może zmieniać kąt obrotu wirtualnej głowy, a korzystając z przycisków nawigacji po sali (5) może zmienić swoją wirtualną pozycję, która jest pokazana w przybliżeniu na zdjęciu sali (8). Aby zakończyć odtwarzanie, użytkownik może nacisnąć ponownie przycisk *Play* (1). Program automatycznie zakończy odtwarzanie, gdy skończy się wczytana ścieżka muzyczna. Po zakończeniu odtwarzania na wykresie (6) wyświetli się przebieg czasowy odtworzonej ścieżki (kolor pomarańczowy - prawy kanał, kolor niebieski - lewy kanał), a na wykresie (7) wyświetli się historia zmiany kąta płaszczyzny horyzontalnej. Po obu wykresach użytkownik może nawigować przy pomocy przycisków (9). Sytuacja po zakończeniu odtwarzania została przedstawiona na rysunku 4.4

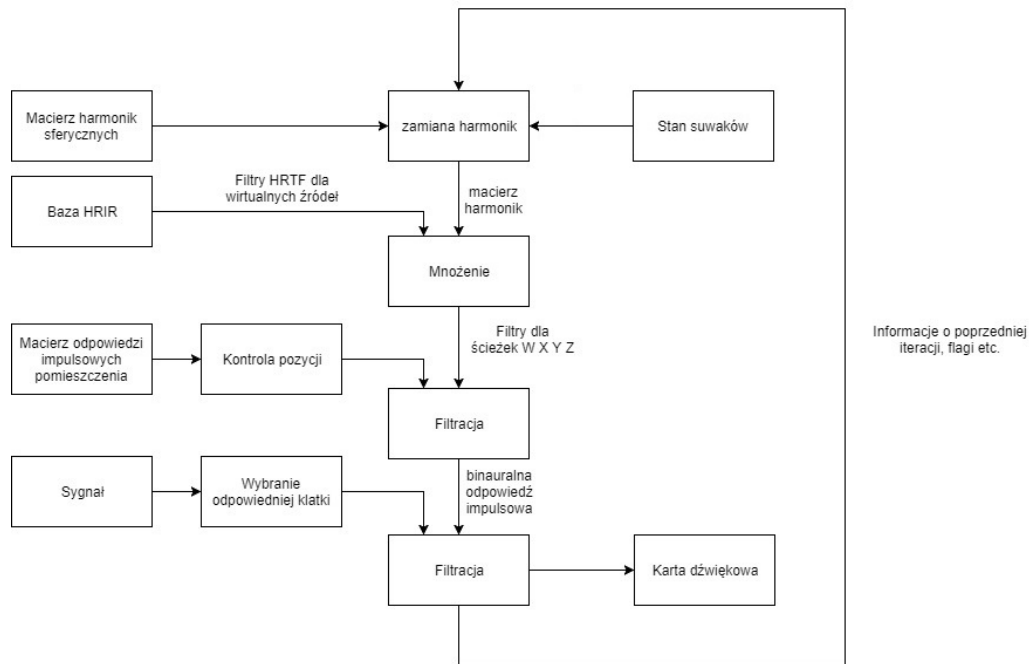


Rys. 4.4. Sytuacja po zakończeniu odtwarzania

4.4. Działanie programu

Po rozważeniu powyższych zagadnień zaimplementowano wybrane narzędzia i rozwiązania. Program tuż przed wyświetleniem interfejsu wczytuje bazę HRTF, bazę kątów ω i θ i wyznacza macierz harmonik sferycznych na podstawie (wzorów 2.6 - 2.8), a następnie pseudoodwraca tę macierz. Wczytuje także jednokanałową ścieżkę muzyczną, której odtwarzanie będzie symulowane i impulsy zmierzone w sali filharmonii.

Po naciśnięciu przycisku *Play* program zaczyna działać w pętli, której schemat przedstawiono na rysunku 4.5.



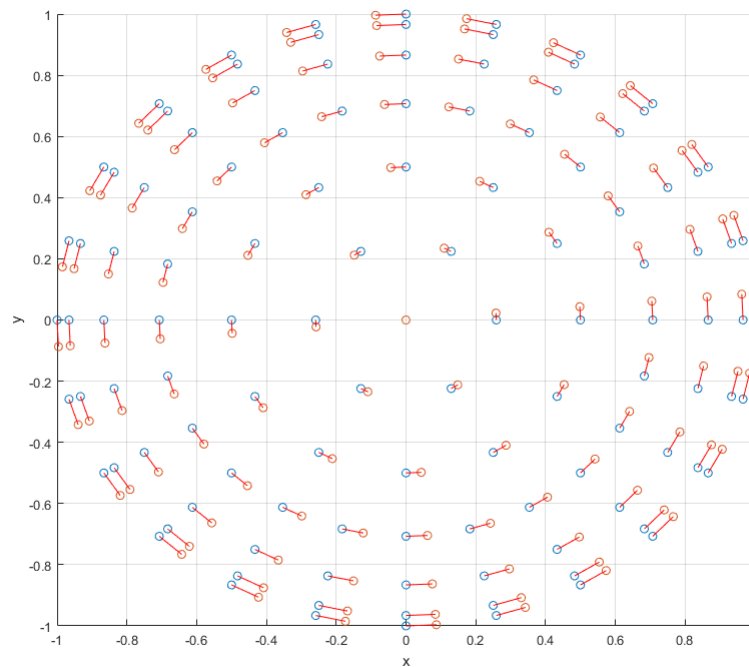
Rys. 4.5. Schemat głównej pętli programu

Najpierw program oblicza binauralną odpowiedź impulsową w konkretnym miejscu sali i pod konkretnym kątem obrotu wirtualnej głowy na podstawie czytanej pozycji i ustawienia suwaków obrotu. Następnie program filtruje nią wybraną, niewielką klatkę ścieżki muzycznej. Wynik filtracji, czyli klatka binauralnej ścieżki muzycznej, która odpowiada odsłuchowi odpowiedzi impulsowej sali w konkretnym punkcie pomieszczenia pod konkretnym kątem obrotu głowy, program przekazuje na kartę dźwiękową i przechodzi do kolejnej iteracji.

4.4.1. Zamiana harmonik sferycznych

Program dokonuje obrotu wirtualnej głowy poprzez zamianę harmonik sferycznych w macierzy. Harmoniki sferyczne definiują umiejscowienie wirtualnych głośników na sferze wokół głowy słuchacza. Ich zamiana zatem będzie powodowała zamianę wirtualnych głośników. Jeśli użytkownik chce obrócić wirtualną głowę zadany kąt θ_n, ϕ_n , to program

powinien obrócić sferę wirtualnych głośników o kąt $-\theta_n, -\phi_n$. Ponieważ baza HRTF jest spróbkowana, do dyspozycji jest tylko 187 wirtualnych głośników w konkretnych punktach. Zakładając, że użytkownik zadał kąt obrotu w płaszczyźnie horyzontalnej równy -5° , program nie będzie mógł obrócić sfery głośników – istnieje wirtualny głośnik $S_{\theta=30^\circ, \phi=105^\circ}$ (którego położenie jest zdefiniowane jako $\theta_1 = 30^\circ, \phi_1 = 105^\circ$). Obrót o 5° zakładałby, że znana jest HRTF wirtualnego głośnika $S_{\theta=30^\circ, \phi=110^\circ}$, ale baza nie uwzględnia żadnych punktów pomiędzy $S_{\theta=30^\circ, \phi=105^\circ}$ a $S_{\theta=30^\circ, \phi=120^\circ}$. Sytuację tę przedstawiono na rysunku 4.6.

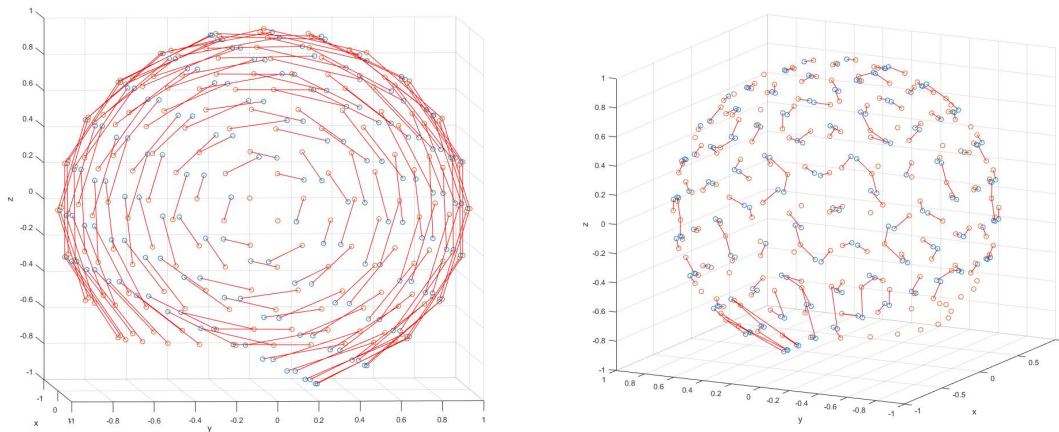


Rys. 4.6. Obrót bazy harmonik sferycznych o 5° w płaszczyźnie horyzontalnej - widok od góry. Niebieskie punkty to pierwotne miejsca głośników, czerwone to głośniki po obrocie. Czerwonymi liniami oznaczono dopasowanie do najbliższych znanych pozycji z bazy HRTF

W pracy przyjęto więc metodę zaokrąglania do najbliższej pozycji, w której znana jest HRTF głośnika i założono, że użytkownik może obracać głowę wyłącznie o wielokrotność 15° .

Obrót w płaszczyźnie horyzontalnej był stosunkowo łatwy do implementacji. Ponieważ założono powyższe, wystarczyło użyć przesunięcia kołowego w głównych pierścieniach bazy (funkcja `circshift`), dodając po prostu do kąta ϕ zadany kąt, i przesunięcia kołowego z zaokrągleniem w górnych pierścieniach. Obrót w pozostałych dwóch płaszczyznach wymagał natomiast algorytmu, który będzie obracać sferę w nietrywialny sposób. Program zamienia współrzędne sferyczne na kartezjańskie, zamienia miejscami współrzędne y i z (obróć w płaszczyźnie czołowej), lub x i z (obróć w płaszczyźnie przedniej), zamienia z powrotem współrzędne na sferyczne, dodaje zadany kąt obrotu do

θ i zamienia współrzędne na kartezjańskie, żeby zamienić y i z , lub x i z (rysunek 4.7a). Następnie wykorzystując funkcję `dsearchn` wyszukuje najbliższe głośniki istniejące w bazie (rysunek 4.7b) i zgodnie z wyszukiwaniem zamienia harmoniki sferyczne. Sytuację obrotu o 30° w płaszczyźnie środkowej przedstawiono na rysunku 4.7.



(a) Kolorem niebieskim zaznaczono pierwotne (b) Dopasowanie obróconych głośników do głośników znanych, obecnych w bazie. Czerwoną linią zaznaczono dopasowanie.

Rys. 4.7. Obrót sfery wirtualnych głośników o 30° w płaszczyźnie środkowej.

Program sprawdza, czy pozycja suwaków w stosunku do poprzedniej iteracji pętli uległa zmianie, i jeśli tak, to wykorzystując powyższe rozwiązania wykonuje obrót głośników kolejno: w płaszczyźnie horyzontalnej, środkowej i przedniej.

4.4.2. Zamiana formatu ambisonicznego na binauralny

Mając gotową, obróconą bazę harmonik sferycznych (\mathbf{Y}) program mnoży ją z macierzą HRTF (\mathbf{H}), otrzymując 4 filtry i korzystając z funkcji `fftfilt` filtruje nimi 4 ścieżki odpowiedniego nagrania ambisonicznego według 2.9. Nagranie ambisoniczne jest wybierane z bazy nagranych odpowiedzi impulsowych sali zgodnie z wybraną przez użytkownika pozycją. Dzięki temu program otrzymuje binauralną odpowiedź impulsową odpowiadającą odsłuchowi w konkretnym miejscu sali pod konkretnym kątem obrotu głowy.

4.4.3. Filtracja ścieżki muzycznej

Założono, że sygnał wyjściowy programu musi być szerokopasmowy, aby użytkownik lepiej lokalizował dźwięk, i zróżnicowany, aby możliwe było pełnej akustyki pomieszczenia. Z powyższych powodów odrzucono syntetyczną odpowiedź impulsową i

zdecydowano się na dowolną ścieżkę muzyczną. Użyto więc utworu Alt-J „Intro”, którą program splata z binauralną odpowiedzią sali.

Powstaje jednak problem złożoności obliczeniowej opisany w rozdziale 3.2 – odpowiedź sali to 131072 próbek. Implementując klasyczny splot w systemie czasu rzeczywistego – wypuszczanie na kartę dźwiękową pojedynczych obliczonych próbek – czas potrzebny na obliczenia jest zbyt duży dla standardowego komputera. Zdecydowano się więc na podzielenie sygnału na możliwie niewielkie klatki i ich filtrację z zastosowaniem algorytmu sekcjonowania szybkiego splotu sygnału dyskretnego *textitoverlap-save*, działając w dziedzinie widmowej. Najmniejszy rozmiar klatki, który pozwalał na płynne działanie programu ustalono eksperymentalnie na 8192 próbek, co spowodowało, że w najbardziej niekorzystnym przypadku (zmiana ustawień suwaków bądź pozycji przez użytkownika tuż po sczytaniu ich wartości przez program) wynosi 372 [ms].

[dawać tu opis interpolacji?]

Po przefiltrowaniu sygnału na kartę dźwiękową, a także do wektora całości przetworzonego dźwięku, przekazywana jest klatka binauralnej ścieżki muzycznej. Następnie program przechodzi do kolejnej iteracji pętli przekazując jej flagi i inne informacje, taki jak stary stan suwaków czy starą pozycję.

5. Testy programu

5.1. Analiza czasowa

5.2. Testy odsłuchowe

6. Zakończenie

6.1. Wnioski

6.2. Dalsze możliwe prace

poprawić filtry kirkebyego i na tej podstawie uzupełnić rozdział 3.3, dołożyć rysunek i Pomiary. Rysunki do overlap add i save.

Bibliografia

- [1] E. Ozimek. *Dźwięk i jego percepcja: aspekty fizyczne i psychoakustyczne*. Wydawnictwo Naukowe PWN, 2002.
- [2] A. V. Oppenheim, R. W. Schafer i J. R. Buck. *Cyfrowe Przetwarzanie Sygnałów*. Warszawa, Wydawnictwa Komunikacji i Łączności WKŁ, 1997.
- [3] Tomasz Zieliński. *Cyfrowe przetwarzanie sygnałów: od teorii do zastosowań*. Warszawa, Wydawnictwa Komunikacji i Łączności WKŁ, 2016. ISBN: 9788320616408.
- [4] A. Gołaś. *Metody komputerowe w akustyce wnętrz i środowiska*. Kraków, Wydawnictwo Akademii Górniczo-Hutniczej, 1995.
- [5] B. C. Moore. *An introduction to the psychology of hearing, 5th Edition*. Bingley, Emerald Group Publishing Ltd, 2003.
- [6] A. Farina. *A-format to B-format conversion, University of Parma*. <http://pcfarina.eng.unipr.it/Public/B-format/A2B-conversion/A2B.htm>. (odwiedzona 01.11.2017).
- [7] A. Farina. *Impulse response measurements by exponential sine sweeps*. Parma, 2008.