



M Ű E G Y E T E M 1 7 8 2

**Budapesti Műszaki és Gazdaságtudományi Egyetem**

Villamosmérnöki és Informatikai Kar

Méréstechnika és Információs Rendszerek Tanszék

# Többcsatornás zengető megvalósítása VST környezetben

BSC SZAKDOLGOZAT

*Készítette*  
Molnár Péter

*Konzulens*  
dr. Bank Balázs

2012. december 9.

# Tartalomjegyzék

<b>Kivonat</b>	<b>4</b>
<b>Abstract</b>	<b>5</b>
<b>Bevezető</b>	<b>6</b>
<b>1. Konvolúciós zengetők (rövid kitekintés)</b>	<b>11</b>
<b>2. Módszerek a zengetés megvalósításához</b>	<b>14</b>
2.1. A zengés mérése, impulzusválasza . . . . .	14
2.2. Korai visszaverődések . . . . .	16
2.2.1. A tükörforrások módszere . . . . .	16
2.2.2. FIR szűrős rendszerek . . . . .	17
2.3. A késői zengetés . . . . .	19
2.3.1. Teremjellemzők . . . . .	19
2.3.2. Az egyszerű késleltetés . . . . .	20
2.3.3. A fésűszűrő . . . . .	20
2.3.4. A mindentáteresztő szűrő . . . . .	22
2.3.5. A szűrők kombinálása . . . . .	22
2.3.6. Feedback Delay Network . . . . .	24
<b>3. Feedback Delay Network alapú zengető fejlesztése</b>	<b>26</b>
3.1. Jot és Chaigne konstrukciója . . . . .	26
3.2. A FDN egyszerűsítése . . . . .	27
3.3. A késleltetővonalak hosszának meghatározása . . . . .	28
3.4. A veszteségi szűrők . . . . .	29
3.4.1. A lecsengési idő . . . . .	30
3.4.2. A vágási frekvencia . . . . .	30
<b>4. Korai visszaverődések megvalósítása</b>	<b>33</b>
4.1. A paraméterek meghatározása . . . . .	33
4.1.1. A késleltetővonalak hossza . . . . .	33

4.1.2.	A visszaverődések iránya . . . . .	35
4.1.3.	Frekvenciafüggő elnyelés modellezése . . . . .	35
4.2.	Panorámázás a hangszórók között . . . . .	36
4.2.1.	VBAP két dimenzióban . . . . .	36
4.2.2.	VBAP három dimenzióban . . . . .	39
<b>5.</b>	<b>A diffúz szakasz kimeneteinek korrelálatlansága</b>	<b>41</b>
<b>6.</b>	<b>A VST plug-in</b>	<b>45</b>
<b>7.</b>	<b>Összefoglalás, értékelés</b>	<b>47</b>
	<b>Irodalomjegyzék</b>	<b>50</b>
	<b>Függelék</b>	<b>51</b>
F.1.	A VST plug-in C++ osztályai és függvényei . . . . .	51
F.2.	A forráskód és a VST plug-in linkje . . . . .	53
F.3.	CD melléklet . . . . .	53

## HALLGATÓI NYILATKOZAT

Alulírott *Molnár Péter*, szigorló hallgató kijelentem, hogy ezt a szakdolgozatot meg nem engedett segítség nélkül, saját magam készítettem, csak a megadott forrásokat (szakirodalom, eszközök stb.) használtam fel. Minden olyan részt, melyet szó szerint, vagy azonos értelemben, de átfogalmazva más forrásból átvettem, egyértelműen, a forrás megadásával megjelöltem.

Hozzájárulok, hogy a jelen munkám alapadatait (szerző(k), cím, angol és magyar nyelvű tartalmi kivonat, készítés éve, konzulens(ek) neve) a BME VIK nyilvánosan hozzáférhető elektronikus formában, a munka teljes szövegét pedig az egyetem belső hálózatán keresztül (vagy autentikált felhasználók számára) közzétegye. Kijelentem, hogy a benyújtott munka és annak elektronikus verziója megegyezik. Dékáni engedéllyel titkosított diplomatervek esetén a dolgozat szövege csak 3 év eltelte után válik hozzáférhetővé.

Budapest, 2012. december 9.

---

*Molnár Péter*  
hallgató

# Kivonat

A zengés az egyik leggyakoribb és legtermészetesebb hangeffektus, mellyel az ember találkozik. A jelenség különösen zárt termekben észrevehető, ezért például színházak vagy koncerttermek tervezésénél fontos megvizsgálni az adott terem akusztikai jellemzőit a zengés szempontjából is. A hanghatás eredményeképp egyrészt nagyobb termekben is eljut a hallgatóság füléig akkora hangenergia, amellyel élvezhető az előadás.

Hangfelvételek során nem mindig áll rendelkezésünkre megfelelő akusztikájú terem, ezért felmerül az igény arra, hogy a zengést mesterségesen állítsuk elő. Erre már több hardveres és szoftveres megoldást is kitaláltak. Dolgozatomban a megvalósítás szoftveres módszereire térek ki.

Amint az a zengés impulzusválaszának a jellegén is látszani fog, a zengető megtervezéséhez két fő hatást kell figyelembe venni: a korai hangvisszaverődéseket és az utózengési jelenséget. A kimeneten e kettő eredőjéből képezzük az eredményt. Látni fogjuk, hogy a terem tulajdonságai alapján hogyan határozhatók meg a zengés paraméterei. Külön-külön kitérek a korai visszaverődések és a diffúz szakasz késleltetővonalakon alapuló modellezésére, kezdve a legegyszerűbb jelfolyamhálózattól egészen a Feedback Delay Network konstrukciókig.

Az általam megvalósított szoftveres zengető működését részletesen ismertetem: melyik módszert választottam, és miért. Szó lesz arról, hogy a veszteségi szűrők hogyan befolyásolják az utózengés jellegét. Ezek a szűrők határozzák meg a zengés lecsengési idejét és a vágási frekvenciáját is. Ezeket az értékeket két szűrőparaméterrel lehet változtatni.

Ma már nagyon elterjedtek a többcsatornás hangrendszerek (5.1, 7.1, stb.), így követelmény, hogy több hangszóró esetén egyes visszaverődések irányát is érzékelni lehessen, illetve a hangszórókon az utózengés korrelálatlanságát is a lehető legnagyobb mértékben biztosítsuk.

A Steingberg cég által kifejlesztett VST SDK alkalmazásával írtam meg a hangeffektet megvalósító programot, melyből a fordító egy plug-in fájlt generál. Az effektet bármilyen, több kimeneti csatornát kezelő, VST-t támogató host programmal használni lehet.

# Abstract

Reverberation is one of the most significant and natural sound effects to be encountered. The phenomenon can be noticed particularly in closed rooms, that is why it is important to take into consideration the acoustic characteristics from the reverberation point of view when designing for example a theatre or a concert hall. Due to the effect a sufficient amount of sound energy gets to the audience's ears even in larger rooms.

Often an acoustically optimal room is not available during sound recording sessions, so the need to produce an artificial reverberator comes up. Today there are many solutions available for both hardware and software realizations. I am focusing on the software methods in my thesis.

As we will see later from the impulse response of reverberation, there are two things that need to be considered when designing a reverberator: the early echoes and the late reverberation. The final result comes up as the sum of these two effects. It will be seen how the parameters of the reverberation can be determined from the acoustic characteristics of a room. I am going to introduce the delay line based models of the early echoes and the late reverberation separately, from the simplest signal flow network to the Feedback Delay Network constructions.

I am giving a detailed description of the software reverberator I have implemented: which method I chose and why. This includes also the loss filters that affect the attributes of the late reverberation. These filters determine the reverberation decay time and the cutoff frequency. These two values can be modified by setting the filter parameters.

Using multi-channel sound systems is very common today (5.1, 7.1, etc.), so it is a requirement the direction of the early reflections can be correctly perceived and also to provide outputs that are uncorrelated on the speakers as much as possible.

I have implemented the program which models the sound effect using the VST SDK developed by Steinberg GmbH., after compiling a plug-in file is generated. The effect can be used by any host program that supports multi-channel outputs and the VST standard.

# Bevezető

## A zengés fogalma, szerepe akusztikai célú termek tervezésében

Dolgozatomban a zengetés megvalósítási lehetőségeivel foglalkozom. A zengés esztétikai szempontból fontos hangjelenség, mivel azonban hangfelvételkor természetes zengést a legtöbb esetben nem tudjuk biztosítani, ezért vált indokolttá, hogy az effektust mesterséges eljárással pótoljuk. Kezdetben a feladatot zengőszoba alkalmazással oldották meg. Később megjelentek először az analóg, majd a digitális zengető eszközök, napjainkban pedig már elterjedtek a szoftverrel megvalósított zengetők. Munkám során a VST fejlesztőkörnyezet alkalmazásával valósítottam meg egy többcsatornás, Feedback Delay Network alapú szoftveres zengetőt.

Egy hangforrás által kibocsátott hanghullám a szabad térben minden irányban terjed. A hangforrástól egy adott  $l$  távolságra lévő érzékelőig van egy közvetlen út, melyen keresztül ez a hang  $c$  sebességgel  $t = l/c$  idő alatt eljut. Környezetünkben azonban számos olyan felület van, amely a hangenergia egy részét visszaveri, és ezek a hangvisszaverődések  $t$ -hez képest késleltetve, kisebb intenzitással szintén eléri az érzékelőt. Ez a helyzet akusztikai szempontból akkor válik igazán érdekessé, ha a teret, melyben a hanghullám terjed, sík felületek (falak) veszik körül. Ez esetben azt tapasztaljuk, hogy a hangforrás által kibocsátott hang időbeli eloszlása úgy alakul, hogy a hangforrás kikapcsolásakor nem esik vissza a teljesítmény hirtelen zérusra, hanem fokozatos csökkenés következik be.

A zengés hangeffektus tehát ezeknek a visszaverődéseknek az eredményeképp jön létre. Nagyon lesarkítva mondhatnánk azt is, hogy a zengés visszhangok sorozata, azonban ez bonyolultabb jelenség annál, hogy külön-külön vizsgálhatnánk az egyes visszaverődéseket. A reflexiók időbeli sűrűsége végtelenül nagy, és ezeket egyenként az emberi fül nem képes megkülönböztetni. Helyette azt fogjuk hallani, hogy az eredeti hang időben folytonosan lecseng.

A zengés jellege az adott terem tulajdonságaitól függ. Teljesen máshogy zenget például egy fürdőszoba, mint egy koncertterem. Egyik legfontosabb jellemző az utó-zengési idő, vagy lecsengési idő: az az idő, amely alatt az amplitúdó ezredrészére csökken. Ennek a meghatározására többféle számítási módszer is létezik. Egyik meg-

közelítés egy statisztikai fogalom, a szabad átlagos úthossz bevezetése volt [11]. A fogalomhoz Clausius neve köthető, aki statisztikai megfontolások alapján meghatározta a szabad átlagos úthossz kifejezését téglatest alakú termekre:

$$d = \frac{4V}{S}, \quad (1)$$

ahol  $V$  a terem térfogata,  $S$  pedig a terem összes felülete, amelyen a hang visszaverődik. Ezt a formulát felhasználva született meg a lecsengési idő legáltalánosabb képlete, mely (levezetés nélkül) így néz ki:

$$T = 0,16 \frac{V}{-S \ln(1 - \alpha)}, \quad (2)$$

ahol  $\alpha$  a felület átlagos elnyelési tényezője. A képletből azt látjuk, hogy a lecsengés ideje függ a térfogattól, a jellemző felület nagyságától és a felületek elnyelési tulajdonságától, a legutóbbit a felület anyaga határozza meg. Az elnyelés foka frekvenciafüggő, ez tovább árnyalja a zengés hangzását. A lecsengési időnek fontos szerepe van a hangerősség növelésében is. Mivel adott teljesítményű hang térbeli energiasűrűsége a térfogattal fordított arányban csökken, ezért nagyobb termek esetén az ellensúlyozás végett növelni kell egyrészt a hangforrás teljesítményét, másrészt az utözengési időt is. Természetesen tetszőleges teremméret-növekedés esetében még így sem tudjuk kompenzálni az energiasűrűséget, ezért az akusztikai termék mérete sem lehet akármekkora. Például egy hangversenytermet hangerősítés nélkül körülbelül 24 000 - 28 000  $m^3$ -ig lehet méretezni.

## Mesterséges zengetés

Ha egy zenei mű felvételekor szeretnénk, hogy olyan hangzása legyen, mintha élő koncerten szólna, akkor egyik lehetőség lenne az, hogy keresünk hozzá megfelelő akusztikájú termet. Erre a lehetőségek korlátozottak, ennek oka főleg a terem használatának a költsége. A számítástechnika fejlődésének köszönhetően ma már igen elterjedtek az alternatív szoftveres megoldások a zengetés megvalósítására is. Természetesen nemcsak zengést, hanem számtalan egyéb hangeffektust és hangszert is sikerült digitálisan modellezni.

Mesterséges zengetés még jóval a digitális korszak előtt is fontos feladat volt. A legrégebbi eljárás zengőszobák alkalmazása volt [11]. A zengőszobában a fölvelt hangot hangszóróval kisugározták, a keletkezett utözengésű hangot mikrofonnal újra fölvetették. A keletkezett zengő hangot alacsonyabb szinten az eredetihez hozzákeverték.



A zengőszoba előnye a háromdimenziós zengőtér jó hangminősége, hátránya viszont a nagy helyfoglalása és a helyhez kötöttsége. Az első zengőszobával megvalósított utózengés meghosszabbítás O. Vierling nevéhez köthető, aki 1940-ben alkalmazta az eljárást a Berliini Állami Opera akusztikájának javítására [11].

A zengőszoba egydimenziós megfelelőjét, a rugós zengetőt 1951-ben H. Michels valósította meg. [11]. Egy 60 cm hosszú, 3,2 mm vastag acélrugót két végén gumi-szalaggal rögzített, hogy a külső rezgések hatását kiküszöbölje. A rugót egyik végén mágneses vágófej vezérli. A rugó átveszi a rezgéseket, de rezgése saját rugalmassági adatai szerint cseng le. A másik végén mágneses hangszedő veszi fel a már zengéssel rendelkező hangjelet, melyet körülbelül 70 dB-lel kell erősíteni. Előnye a kis méret, továbbá egyszerű felépítésű, olcsó. Minősége viszont elmarad a zengőszobáétól.

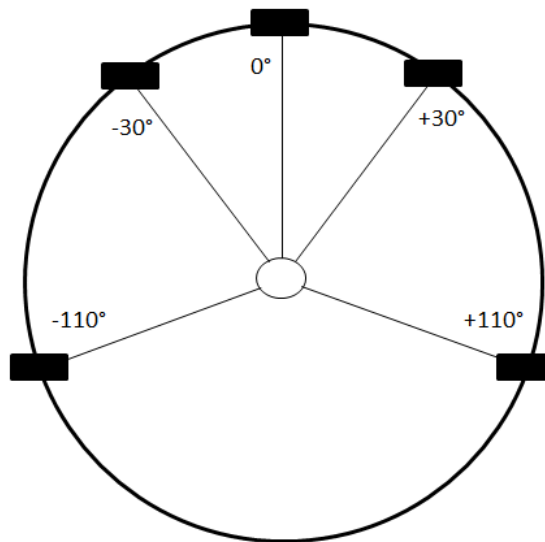
W. Kuhl 1958-ban megvalósította kétdimenziós acéllemezes zengetőt [11]. A berendezés egy 2x1 m nagyságú vékony acéllemez, melynek gerjesztése és a zengetett hang generálása az előző megoldáshoz hasonló. Az utózengés ideje a lemez oldalához közelítő csillapító anyagú lappal módosítható. A zengető előnye, hogy nagy lecsengési idő állítható be. Továbbá a sajátfrekvenciák (módusok) sűrűsége nem változik a frekvenciával, ezért a mély hangokon elég nagy a módusok sűrűsége. Hátránya viszont a hangszín érzékelhető megváltozása, és kétdimenziós volta miatt nem éri el a zengőszoba minőségét. A zengetőlemez népszerű volt a stúdiótechnikában. Az Elektromerstechnik német cég 1983-ig több, mint 4000 példányt gyártott belőle.

Az 1980-as években, ahogy a digitális technika egyre inkább beépült hétköznapjainkba, egyértelművé vált, hogy az akusztikai segédeszközök megvalósítása a számítógépeken fog történni. A első digitális zengetők egyike volt az EMT 251 típusú zengetője, melynek nemcsak utózengési idejét, hanem három frekvenciasávban a zengés jellegét is változtatni lehetett. A számítógépes zeneszerkesztésben jelentős mérföldkőnek számított, amikor a Steinberg nevű német cég 1996-ban kifejlesztette a Virtual Studio Technology (VST) szabványt. [2] A VST fejlesztőkörnyezet segítségével implementálhatjuk a PC-n azokat a virtuális hangszereket (valódi hangszer utánzatát vagy szintetizátort) és effektusokat, amelyekre az elvi modellt már megalkották. Ezeket a szoftveresen megalkotott VST hangszereket vagy hanghatásokat plug-in-nak nevezzük. A plug-in nem önálló program, hanem ún. host program segítségével használható. Ma már nagyon sok hangrögzítő, zeneszerkesztő alkalmazás támogatja ezt a szabványt (Cubase, Acoustica, Fruity Loops Studio, AudioMulch, stb.).

## Többcsatornás hangrendszerek

Szakdolgozatomban a célom a VST fejlesztőkörnyezet alkalmazásával egy olyan mesterséges zengető megvalósítása, amely több hangszóróból álló rendszerekre is alkalmazható. Így egy surround hangrendszeren megszólaltatva úgy fogjuk érzékelni, mintha a visszaverődések valóban különböző irányokból érkeznének meg hozzánk.

Zenében még most is az egyik leggyakoribb a kétcsatornás sztereó alkalmazása, de elterjedtek a surround rendszerek is. Például koncert DVD-t már nehéz elképzelni 5.1-es kimenet nélkül. Az 5.1 nevében is benne van, hogy néz ki ez a rendszer: az 5-ös szám jelöli a normál hangszóróknak a számát, a hatodik hangszóró funkciója pedig a mély hangok lesugárzása. Az 1. ábra [1] mutatja a hangszórók javasolt elrendezését. Ez alapján különböztetjük meg a center hangszórót, illetve a két első (front) és a két hátsó (rear) hangfalat.



1. ábra. Hangszórók elrendezése 5.1 esetén

Az előzőt továbbgondolva kapjuk a 7.1-es surround rendszert. Itt a különbség annyi, hogy kiegészítették még két hátsó csatornával a hangrendszert, hogy a hang még jobban körülvegyen minket. A hasonlóság az előbbihez képest, hogy mind a kettő esetén elsősorban két dimenzióban határozták meg a pozíciókat, tehát a kimeneti csatornák egy síkban helyezkednek el.

A térhatás tovább fokozható, háromdimenziós hangszóró-elrendezésekkel. A surround hangrendszert követte az Auro 3D generáció [5]. Az Auro 3D hangrendszerekben vannak magassági hangszórók, így a tér minden irányából halljuk a hangokat. Különböző konfigurációk esetén változhat a magassági hangszórók száma: például 9.1 esetén négy, 11.1 esetén pedig hat magassági csatorna szerepel az elrendezés-

ben. A technológia még nem terjedt el, melynek az az oka elsősorban, hogy drága, és ilyen sok hangszórót nem könnyű elhelyezni egy szobában. A zengetőben is figyelmen kívül hagytam, mivel nem állt rendelkezésemre ennyi kimeneti csatorna. Azért alkották ezt meg, hogy bemutassák, hogy az audió jelfeldolgozás esetén ma már nem a mintavételi frekvencia további növelése eredményez jól hallható javulást a hangzásban. A hangszórók számának a növelésével tehető valóságosabbá a térérzet keltése.

## **A dolgozat felépítése**

Az első fejezetben ismertetem, hogy milyen algoritmusok léteznek a zengetés megvalósítására. Elsősorban a késleltetővonal alapú rendszerekről lesz szó, de röviden kitérek a konvolúciós zengetésre is, amely valóságosabb eredményt ad, viszont kevésbé paraméterezzhető.

A második fejezet annak a zengetőnek a működését mutatja be részletesen, amelyet megvalósítottam, a diffúz szakaszra fektetve a hangsúlyt. Bemutatom a fésűszűrőn alapuló Feedback Delay Network (FDN) felépítését, egyszerűsítési lehetőségét. Szintén ebben a fejezetben lesz szó a veszteségi szűrők beállításáról is.

A következő részekben pedig az effekt több csatornára való kiterjesztésére térek ki, ezek: az irányszámítás a korai visszaverődések esetén, illetve az FDN kimenő jelei korrelálatlanságának a maximalizálása.

# 1. fejezet

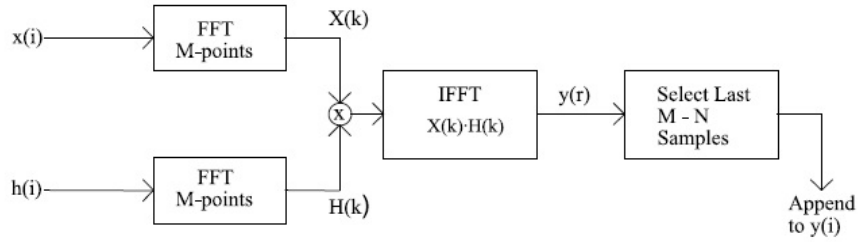
## Konvolúciós zengetők (rövid kitekintés)

Ha a feladatot úgy szeretnénk megközelíteni, hogy egy valós terem zengési karakterisztikáját a lehető legpontosabban szimuláljuk, akkor jönnek szóba a valós idejű konvolúciós algoritmusok. A probléma digitális jelfeldolgozás esetén, hogy  $N$  hosszúságú FIR szűrőre a művelet elvégzésekor időtartományban  $N$  darab szorzást kell elvégezni minden egyes bejövő mintánál. Ahhoz, hogy megfelelő minőségű zengést produkáljunk egy koncertterem esetére, 48 kHz mintavételi frekvencia mellett legalább  $2^{16}$  pontra lenne szükségünk, erre pedig csak nagyon nagy költségek árán lehetne megfelelő architektúrát kialakítani.

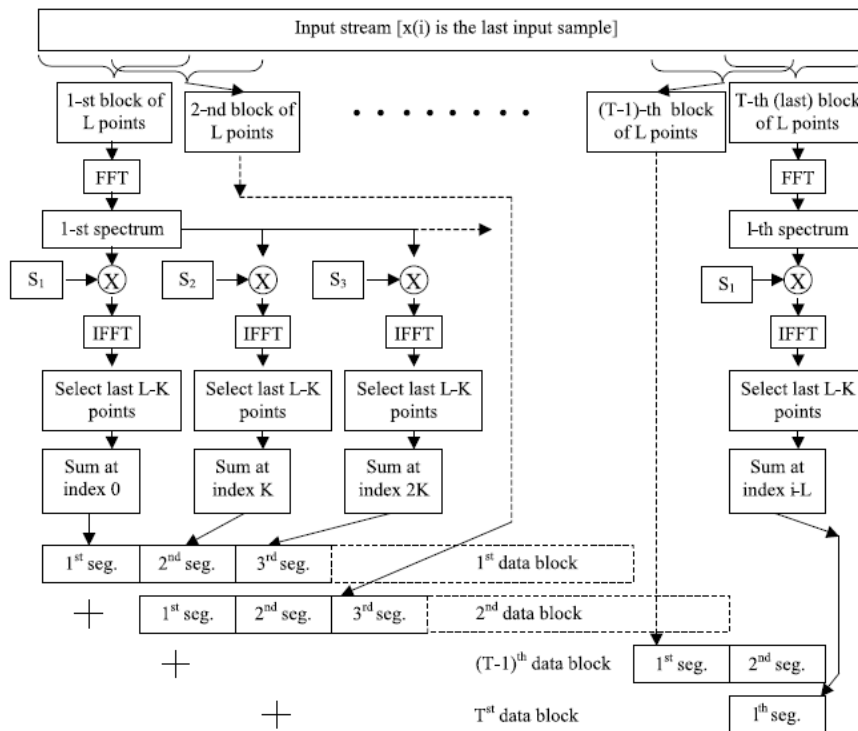
Frekvenciatartományban a konvolúció szorzásra redukálódik, így növelhetjük a hatékonyságot az FFT és IFFT algoritmusokkal. Közvetlenül alkalmazva az FFT eljárást nem fogjuk megkapni az elvárt eredményt, mégpedig azért, mert az algoritmus a jel feldolgozandó szegmensének periodicitását okozza. Ennek az oka az, hogy a mintavételezett jelekre alkalmazott vonatkozó konvolúciós tétel valójában a konvolúciónak és  $N$ -nel való eltoltjának az összegét adja. Ez a cirkuláris konvolúció. Az overlap-and-save [12] algoritmussal (1.1. ábra) kiküszöbölhető a probléma. A lényege, hogy  $M$  pontú FFT esetén csak az utolsó  $M-N$  mintát küldi ki a kimenetre, majd a kimeneti regiszter tartalmát  $N-M$ -mel jobbra eltolja, mielőtt a következő minta szegmensén elvégzi a konvolúciót. Tipikusan  $M = 2N$  értéket választanak.

Az overlap-and-save algoritmus egy módosított változata az, amikor az impulzusválaszt egyenlő hosszúságú blokkokra bontjuk, ezeket a blokkokat külön-külön önálló impulzusválaszként kezeljük, és mindre elvégezzük a fent említett műveletet.  $K$  mintából álló blokk esetén az overlap-and-add FFT ablaka jellemzően  $L = 2K$  hosszúságú. Gyakorlatilag így minden blokkhoz tartozik egy  $s_i$  frekvenciatartománybeli szűrő, melyek együtthatóit a bemeneti jel transzformáltjával megszorozva kapjuk meg az egyes kimeneti blokkokat. A kimenet jel végül az egyes blokkok megfelelően

késleltetett összegzéséből adódik (1.2. ábra).



1.1. ábra. Az overlap-and-save algoritmus sematikus ábrája [12]



1.2. ábra. A partícionált konvolúció [12]

Mivel a partícionált kialakítás esetén lényegesen kevesebb pontra kell elvégezni a Fourier-transzformációt, a be- és a kimenet között a késleltetés csökken, ha a blokkok száma nem túlságosan nagy. Azonban ha a partícióink száma  $P$ , akkor  $P - 1$  összeadást kell elvégeznünk a kimenetre, és  $P$ -szer több memóriahivatkozás szükséges a feladat elvégzéséhez. Ugyanakkor a be- és kimeneti bufferek mérete a kevesebb ponton elvégzendő FFT végett  $L$  méretűre, ezen felül a szorzások száma is számottevően csökken. A SIMD (Single Instruction, Multiple Data) architektúrát támogató processzorokra lefordítva a programot egyes aritmetikai műveleteket egy utasításban végrehajtja, ezzel gyakorlatban jelentősen javít az algoritmus hatékonyságán, szemben a nem partícionált eljárással.

Ma már léteznek a piacon konvolúciós zengetők. Az ezredfordulón a Sony megjelentette a saját zengető eszközét [3], amely több különböző módban is működött, például mono bemenet mellett sztereó és négycsatornás kimenetet is képes előállítani, emellett programválasztási opcióval is rendelkezik: koncert, templom, stúdió, stb.

Szoftver formájában a Vienna Synchronic Library által 2009-ben kiadott Vienna Suite 1.1-es csomagja [4] tartalmaz kilenc audio plug-in-t, mely magában foglal egy konvolúciós zengetőt is, ötféle teremimpulzusválasszal, 96 kHz mintavételi frekvenciáig. Támogatja a WAV (windows audio) és FLAC (free lossless audio codec) formátumú hangfájlokat.

A konvolúciós zengetés előnye, hogy ez a módszer adja vissza egy terem akusztikai jellemzőit a legvalóságosabban. A valós idejű feldolgozás egy közepes teljesítményű számítógépen elfogadható, viszont a késleltetővonal alapú zengetőkhöz képest nagyobb a számításgigénye. Komolyabb hátránya viszont, hogy a zengetés paramétereinek a beállításaira korlátozottak a lehetőségeink, például a korai reflexiók késleltései nem módosíthatók, vagy a lecsengési idő sem állítható be tetszőlegesen.

## 2. fejezet

# Módszerek a zengetés megvalósításához

### 2.1. A zengés mérése, impulzusválasza

Jelfeldolgozás szempontjából a zengést tekinthetjük egy szűrőnek, feltételezve, hogy az adott vizsgált terem lineáris időinvariáns rendszert alkot. Tekintsünk egy pontszerű hangforrást, mint a rendszer egyetlen bemenetét, a két kimenet pedig a jobb és bal fülbe eljutó hangnyomás. Időtartományban a bemenet és a terem hallószervek helyzetétől függő impulzusválaszainak a konvolválásával kapjuk meg az egyes kimeneteket [8]:

$$y_J(t) = \int_0^{\infty} h_J(\tau)x(t - \tau)d\tau \quad (2.1a)$$

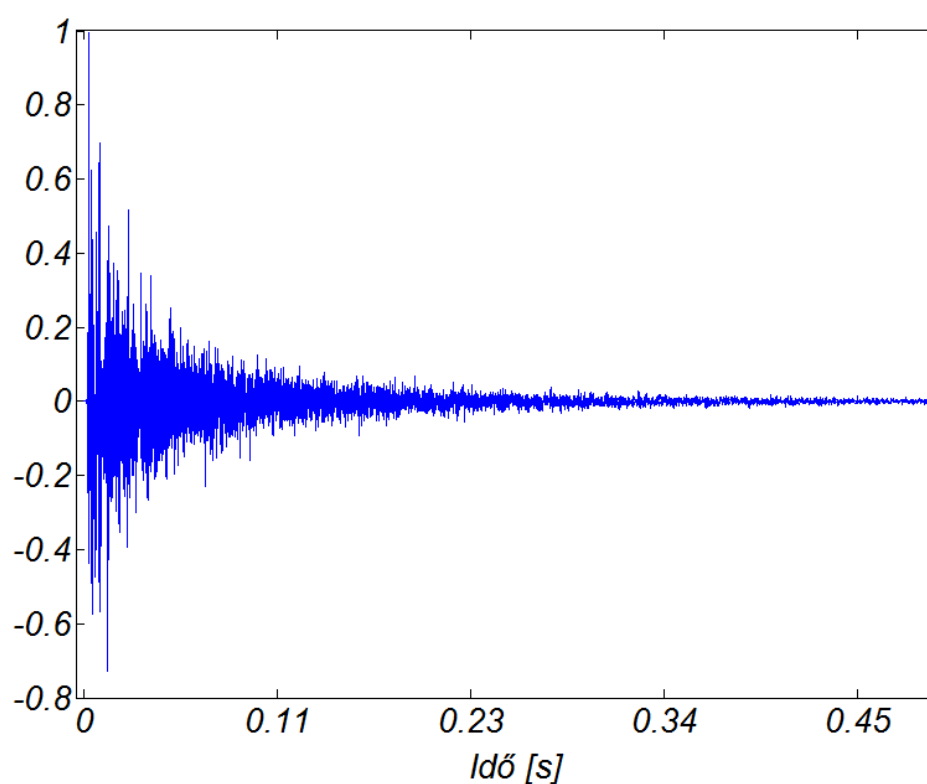
$$y_B(t) = \int_0^{\infty} h_B(\tau)x(t - \tau)d\tau, \quad (2.1b)$$

ahol  $x(t)$  a forrásból származó hang,  $y_J(t)$  és  $y_B(t)$  pedig a hang, amelyet a jobb, illetve bal fülünk érzékel. A szoftveres zengető esetében a kimeneti jelek alatt természetesen a hangszórók által megszólaltatott hangok értendők, de a megközelítés ugyanúgy érvényes ekkor is, ahol  $h_k(t)$  és  $y_k(t)$  a  $k$ -edik hangszóróra vonatkozó impulzusválasz, illetve kimeneti jel.

Amikor a zengést valamilyen formában modellezni szeretnénk, az első felmerülő kérdés az, hogy hogyan tudjuk egy terem zengését megmérni. A mérés során az impulzusválaszra vagyunk kíváncsiak, ugyanis ez önmagában fontos fizikai tartalommal bír. Természetes impulzusszerű hangforrásokat könnyen elő lehet állítani: pisztolylövés, lufi kidurranása vagy egy taps.

Elektronikus jelgenerátorral egységimpulzust egy gömbkarakterisztikát követő hangszóróra kivezelve közvetlenül lehet mérni, mely azonban gyenge jel-zaj viszonyt eredményez, a jel kis energiája miatt [8]. Nagyobb energiát biztosíthatunk úgynevezett sweep jellel vagy zajjal, javítva az SNR-t, melyekből dekonvolúcióval kaphatjuk meg az impulzusválaszt.

A terem impulzusválasza függ a hangforrás, az érzékelő pozíciójától és az iránykarakterisztikától is. Tehát elméletben végtelen sok különböző eredményt kapnánk, ha mindig változtatnánk a helyzetüket. Ugyanakkor statisztikailag lényeges eltérés nem mutatkozna meg, így egy mérés alapján lehet következtetni az akusztikai jellemzőkre.



**2.1. ábra.** A zengés egy jellemző impulzusválasza

Az 2.1. ábrán a zengés egy tipikus impulzusválasza látható. Megfigyelhetők az elején egyenként kiugró impulzusok, amelyek az idő múlásával egyre kisebb amplitúdójúak és egyre nagyobb a sűrűségük. Ez alapján modellalkotásnál két fő dolgot szokás figyelembe venni: a korai diszkrét visszaverődéseket (kb. 100 ms-ig), és az utózengést vagy késői zengést. A termen belüli pozícióktól leginkább a korai visszaverődések függenek.

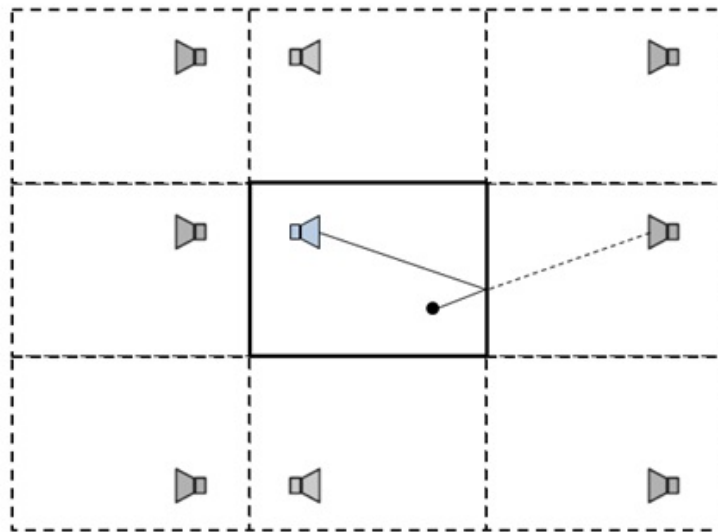


## 2.2. Korai visszaverődések

Az alábbi fejezetben a terem egyszerű geometriai modelljéről lesz szó ahhoz, hogy megvalósítsuk a korai visszaverődéseket. Ha a szoba, amelyet vizsgálunk, téglalatest alakú, akkor a modell egyszerű. Ehhez ismernünk kell a méreteket, a hangforrás illetve a hallgató helyzetét. A forrástól a hallgatóig közvetlen úton a hanghullám egyszerűen a két pont közötti távolság és a hangsebesség hányadosaként kapott idő alatt jut el. Ha ezt az időt tekintjük a referenciának, akkor azt szeretnénk tudni, hogy a falakról visszavert hangok ehhez képest mekkora késleltetéssel érkeznek meg. Ennek a megállapításra szolgál a tükörforrások módszere [8].

### 2.2.1. A tükörforrások módszere

A 2.2. ábra mutatja a tükörforrások módszerének alap gondolatát. A lényeg, hogy a hangforrást a falak mentén tükrözzük, és meghatározzuk a tükrözött forrás és az érzékelő közti távolságot. Ebből az eltelt időhöz egy osztással eljutunk, majd kivonva ebből a referencia időt megkapjuk, hogy az eredeti hanghoz képest mekkora késést szenved el a reflektált hanghullám.



2.2. ábra. A tükörforrások módszere

Ha a késleltetést mintákban szeretnénk megkapni, akkor az alábbi formulát alkalmazva már programozni is lehet:

$$k_i = \frac{l_i}{c} \cdot f_s = t_i \cdot f_s, \quad (2.2)$$

ahol  $c$  a hangsebesség,  $l_i$  az  $i$ -edik reflektált hang által megtett távolság,  $t_i$  a késlel-

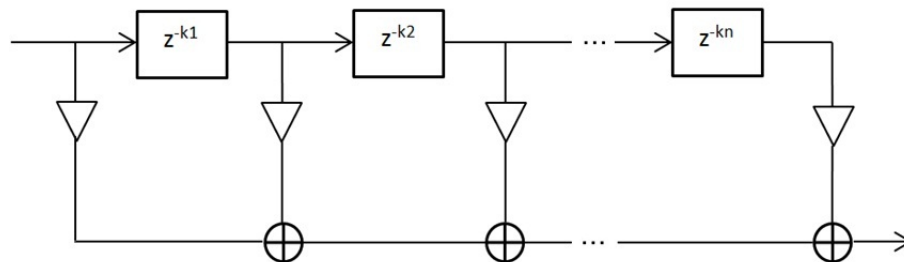
tetés másodpercekben,  $f_s$  pedig a mintavételi frekvencia. Tipikusan  $f_s = 44,1\text{kHz}$ .

A forrást a falak mentén egyszer tükrözve kapjuk meg az elsőrendű visszaverődéseket. Magasabb rendű reflexiókhoz a tükröforrásokat még tovább tükrözzük, például a harmadik esetén már három tükrözést kell elvégezni. A probléma az, hogy ha  $N$  rendet veszünk figyelembe, a tükröforrások száma  $N$ -nel közel exponenciálisan növekszik, s ezáltal komoly mértékben megemelheti a számításigényt, amellyel a processzort túlságosan leterhelhénk. Ezért ez az algoritmus nem is alkalmas arra, hogy az utózengést modellezzük, nem beszélve arról, hogy nagyon pontatlan is volna erre használni.

A korai zengetésnél ugyanakkor legfeljebb nagyjából 80–100 ms késleltetést szokás figyelembe venni. Nagyon kis késleltetések esetén megváltoztathatja a hangforrás látszólagos pozícióját, nagyobbak esetén pedig a hangforrás látszólagos mérete nőhet meg [8].

### 2.2.2. FIR szűrős rendszerek

A korai visszaverődésekhez kézenfekvő megoldás FIR rendszerek alkalmazása. A 2.3. ábrán látható szűrő késleltetőelemeit megfeleltetjük az 2.2 egyenletben meghatározott késleltetések értékeivel. Az erősítések nagyságát pedig a falak energiaelnyelési tényezőiből képezzük le. Tegyük föl, hogy minden visszaverődésnél ugyanakkora az energiaveszteség. Elsőrendű reflexió esetén a visszavert hang amplitúdója  $a$ -szoros lesz az eredetinek, másodrendűnél ez az érték  $a^2$ , és így tovább, ahol  $a < 1$ .



2.3. ábra. Korai reflexiók realizálása FIR szűrővel

Az elnyelés mértéke természetesen frekvenciafüggő, és ha ezt is szeretnénk figyelembe venni, akkor legegyszerűbb esetben  $a$  erősítés helyére képzeljünk el egy elsőfokú aluláteresztő szűrőt. Valóságos elnyelést ilyen szűrővel nem lehet nagy pontossággal szimulálni. A 2.1. táblázat ismerteti néhány természetes anyag elnyelését a frekvencia függvényében [11]. Az elnyelési tényezők ez esetben a beeső és visszavert hanghullám teljesítményére vonatkoznak. Vakolatoknál és természetes anyagoknál a frekvencia függés nem olyan számottevő. Egyenetlen, ráncolt felületeken, például

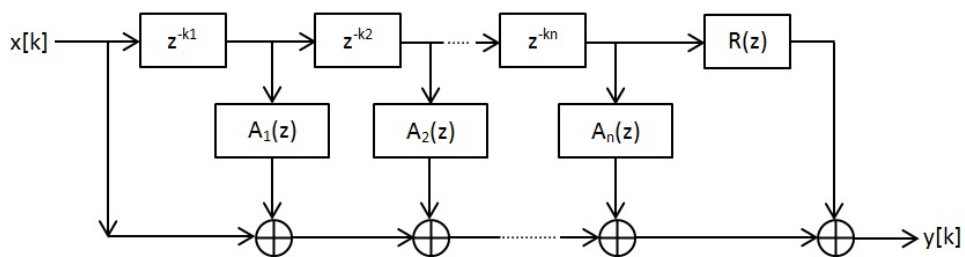
függvényökön az elnyelési tényező növekedése jellemző magasabb frekvenciákon.

**2.1. táblázat.** *Néhány anyag elnyelési értékei*

Anyag neve	Frekvencia [Hz]					
	125	250	500	1000	2000	4000
Öntött beton	0.010	0.012	0.016	0.019	0.023	0.035
Mészelt téglafal	0.012	0.013	0.017	0.020	0.023	0.025
Síme művakolat	0.03	0.03	0.04	0.04	0.04	-
Fapadló	0.09	0.09	0.08	0.09	0.10	-
Farostszövet ráncolva	0.03	0.12	0.15	0.27	0.37	0.42
Bársonyfüggöny ráncolva	0.05	0.22	0.28	0.38	0.50	0.64

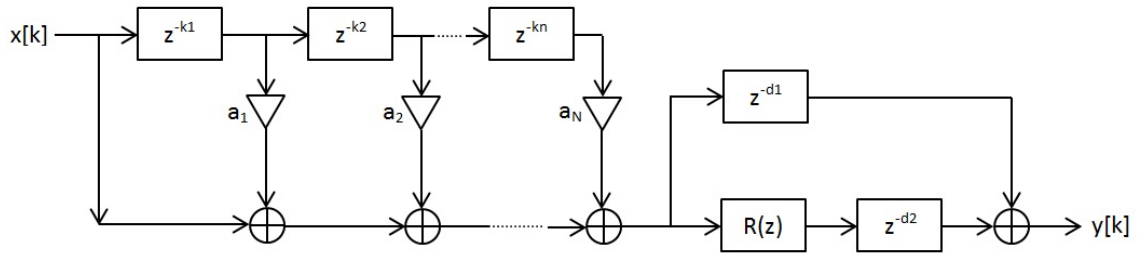
Az összes reflexióra külön szűrőt tervezni nem túl hatékony megoldás, mivel a számításgigányt jelentősen növelheti, és a korai visszaverődéseket sem fogjuk egyenként hallani. Kielégítő megoldás lehet az, hogy csoportokra bontjuk a késleltetővonalak kimeneteit, és egy csoportban mindre ugyanazt a szűrőt alkalmazzuk. Például a reflexiók rendje alapján csoportosíthatunk, és ugyanúgy járunk el, mint a fentebb említett esetben, a különbség, hogy most nem  $a$ -val számolunk, hanem  $A(z)$  átviteli függvénnyel.

Hatékony zengetéshez született többféle konstrukció is, mely FIR szűrőket közvetlenül alkalmaz korai visszaverődések modellezésére. A Schroeder-féle [8] ötlet az, hogy a késői zengetővel kombináljuk a FIR szűrőt (2.4. ábra).  $R(z)$  a zengetőnek az átviteli függvénye, amelynek a bemenetére az utolsó FIR késleltetővonal kimenetét kötjük. Így a korai visszaverődés a kimeneten az utószög előtt megjelenik. Itt az erősítéseket már helyettesítettem  $A_i(z)$  szűrőkkel, melyek reprezentálják a falak elnyelésének frekvenciafüggését.



**2.4. ábra.** *FIR szűrő kombinálva a zengetővel*

Moorer megoldása annyiban különbözik az előbbitől, hogy ez esetben a FIR kimenetére kötjük a késői zengetőt, a visszhangok sűrűségének a növelése végett. A 2.5. ábrán szerepel egy  $d1$  és egy  $d2$  késleltetés is, melyeket úgy kell megválasztani, hogy a korai reflexió legnagyobb késleltetéssel rendelkező kimenete összhangban legyen a zengető kimenetével.



2.5. ábra. FIR és  $R(z)$  kaszkádosítása

## 2.3. A késői zengetés

Ebben a részben többféle már létező módszert is ismertetek a zengetés megvalósítására. A különböző felépítések egymástól nem függetlenek, amelyek időben később jelentek meg, azok tulajdonképpen a korábbi elrendezések továbbgondolt változatai. A zengető annál jobb, minél nagyobb a visszaverődések időbeli sűrűsége, minél nagyobb a módussűrűség, fontos, hogy lehetőleg ne változzon meg az eredeti hangszín.

### 2.3.1. Teremjellelmzők

A termék egyik legjelentősebb paramétere, amelyet egy zengető megvalósításánál szükséges figyelembe venni, az a már bevezető részben említett szabad átlagos úthossz, melyet a  $d = 4V/S$  formulával határozunk meg. Tulajdonképpen ettől függ a zengető késleltetővonalainak hossza.

Egy másik, nagyon fontos jellemző a terem természetes rezonanciafrekvenciáinak, azaz módusainak a száma. Tökéletes téglatest alakú teremre matematikailag zárt alakban meghatározható a módusszám az akusztikai hullámegyenletet megoldva:

$$f_n = \frac{c}{2} \sqrt{\left(\frac{n_x}{L_x}\right)^2 + \left(\frac{n_y}{L_y}\right)^2 + \left(\frac{n_z}{L_z}\right)^2}. \quad (2.3)$$

$L_x$ ,  $L_y$  és  $L_z$  a terem  $x$ ,  $y$  és  $z$  irányú hosszparaméterei,  $n_x$ ,  $n_y$  és  $n_z$  pedig nemnegatív egész számok,  $c$  a hangsebesség. Becslés is létezik az adott  $f$  frekvencia alatti módusok számára [8]:

$$N_f \approx \frac{4\pi V}{3c^3} f^3, \quad (2.4)$$

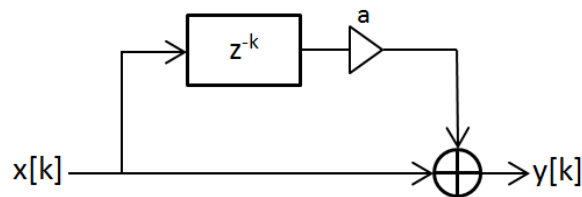
$V$  teremtérfogat esetén. Frekvencia szerint deriválva megkapjuk a Hz-enkénti módusok számát, azaz a módussűrűséget:

$$\frac{dN_f}{df} \approx \frac{4\pi V}{c^3} f^2, \quad (2.5)$$

ennek a reciproka pedig a módusok közötti távolság ( $\Delta f$ ). Például egy  $15000m^2$  térfogatú koncertterem módussűrűsége  $1kHz$ -en körülbelül 4630, így  $\Delta f$  alig több, mint  $2 \cdot 10^{-4} Hz$ . A célunk az, hogy a zengető megvalósításánál kellően nagy módussűrűséget kapjunk, miközben a visszaverődési sűrűség is a lehető legkisebb mértékben csökkenjék, ugyanis később látni fogjuk, hogy egymással ellentétes követelmények.

### 2.3.2. Az egyszerű késleltetés

Bármilyen hangeffektus megvalósításáról legyen szó, a legalapvetőbb építőelem a késleltetés. Egy egyszerű visszhang realizálása egyetlen késleltetővonallal és a hangenergia veszteségét reprezentáló erősítőelemmel lehetséges (2.6. ábra).

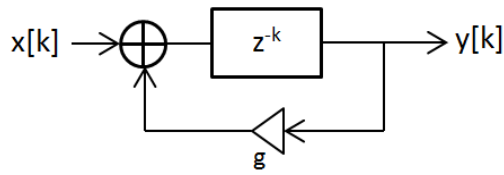


2.6. ábra. Egyszerű visszhang megvalósítása

Tulajdonképpen a korai reflexiókat megvalósító konstrukciónak is ez az alapja. Arról is volt szó, hogy a zengető több, mint pusztán sok-sok visszhang késleltetett összegét képező rendszer. Az eredeti hang megszűnésekor a térben az energia rohamosan csökken, elméletileg azonban soha nem éri el a nullát: a kis energiájú állapot hosszú ideig tart, majd eléri azt a szintet, amikor már nem halljuk. Vagyis a rendszer bemenetére adott válasz végtelen idejű. Ilyen rendszerekben biztosan vannak visszacsatolt késleltetések, vagyis a zengetőt ilyen visszacsatolt késleltetőelemeket tartalmazó szűrőkből, illetve ezek kombinációiból kell megépíteni. A zengetéshez leggyakrabban alkalmazott két szűrőtípus a mindentáteresztő szűrő és a fésűszűrő.

### 2.3.3. A fésűszűrő

A digitális jelfeldolgozás területén a zengető algoritmusok kutatása és az első mesterseges zengetők megalkotása még a korai hatvanas évekre tehető. Schroeder alapgon-  
dolata a fésűszűrő alkalmazása volt [8]. A fésűszűrő egyszerűen egyetlen késleltetés, melynek a kimenetét a bemenetre vezetjük vissza (2.7. ábra).

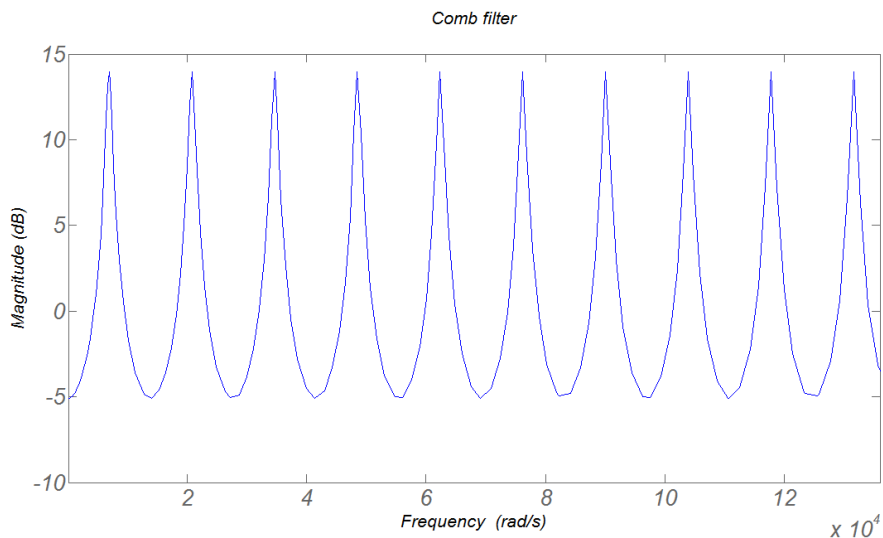


2.7. ábra. Fésűszűrő

A szűrő  $z$ -tartománybeli átviteli függvényét az alábbi módon írhatjuk föl:

$$H(z) = \frac{z^{-k}}{1 - gz^{-k}}, \quad (2.6)$$

ahol  $k$  a késleltetés nagysága mintákban,  $g$  pedig a visszacsatolás erősítése. Az időtartománybeli válasza pedig  $k$  mintánként  $g$  alapú exponenciálisan csökkenő impulzusok sorozata. A rendszer pólusai  $g$ -nek  $k$ -adik komplex gyökei, az ezekhez tartozó frekvenciákon veszi fel a szűrő frekvenciamenete a maximum értékeit, melyekből  $k$  darab van. A frekvenciamenet így fésű jellegű lesz (2.8. ábra), innen kapta a nevét is.

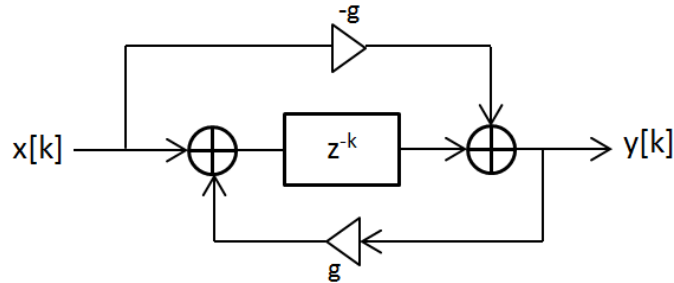


2.8. ábra. A fésűszűrő frekvenciamenete

Önmagában a fésűszűrő még nem alkalmas zenetetésre. Kis késleltetés megválasztásával növelhetjük a visszaverődések sűrűségét, ennek azonban az az ára, hogy a frekvenciatartományban kevés csúcspontja lesz. A módussűrűség megegyezik a késleltetés nagyságával, így kis késleltetés kis módussűrűséget is eredményez. Nagy  $k$  esetén viszont az időtartományban lesz problémánk, ugyanis ekkor az eredmény kis visszaverődési sűrűség, azaz inkább visszhang jellegűvé válik.

### 2.3.4. A mindentáteresztő szűrő

A fésűszűrő kis módosításával kapjuk meg a mindentáteresztő szűrőt. A különbség annyi, hogy szerepel benne egy előrecsatoló ág is, melynek az erősítése a visszacsatolás erősítésének az ellentettje (2.9. ábra).



2.9. ábra. Mindentáteresztő szűrő

A mindentáteresztő szűrő átviteli függvénye:

$$H(z) = \frac{z^{-m} - g}{1 - gz^{-m}}, \quad (2.7)$$

A szűrő jellemző tulajdonsága, hogy az átvitele minden frekvencián egységnyi, ahogy a nevében is benne van. A frekvenciamenet következtében azt várhatnánk, hogy a fésűszűrő módusproblémáját ezzel ki tudjuk küszöbölni. Gyakorlatban a hangzása nem fog sokban különbözni a másikhöz képest, ez pedig a hangok ember általi feldolgozása miatt van: a fülünk rövid idejű frekvenciaanalízist képes „elvégezni”, az elvileg végtelen idő helyett.

A mindentáteresztő szűrők és a fésűszűrők önmagukban nem alkalmasak hatékony zengetés megvalósítására, viszont a későbbi konstrukciók realizálásához fontos alapelemek lesznek.

### 2.3.5. A szűrők kombinálása

Az alapelemeink már megvannak ahhoz, hogy egy jó zengetőt készítsünk. Hogyan használjuk föl ezeket? Schroeder továbbfejlesztett konstrukciója fésűszűrők és mindentáteresztők kombinációjából épül fel. Az egyes elemeket egymással tudjuk sorban vagy párhuzamosan összekapcsolni.

Két fésűszűrő soros kapcsolata semmiképp nem jó megoldás, így ugyanis csak azok a frekvencia komponensek jelennek meg a kimeneten, amelyeken minkét szűrő átereszt. Mindentáteresztők sorba kötése ugyanakkor ilyen szempontból nem okoz problémát: bármennyit hozzáfűzve megmarad mindentáteresztő jellege.

Több párhuzamosan kapcsolt fésűszűrő azt eredményezi, hogy jelentősen megnövelhetjük a frekvenciamenet csúcsainak számát, tehát az eredő módussűrűség az egyes szűrők módussűrűségeinek összege lesz, sőt a visszaverődési sűrűség is összegződik ezáltal.

Schroeder előállt egy konstrukcióval, mely négy fésűszűrő párhuzamos kapcsolásából, a kimeneten pedig két mindentáteresztő soros kapcsolásából áll [8]. A fésűszűrők késleltetővonalait úgy választotta meg, hogy a legnagyobb és legkisebb késleltetés aránya közel 1,5 legyen (30-45 ms között), míg a mindentáteresztőké egy nagyságrenddel kisebb. Így a fésűszűrők produkálják az utózengéshez a megfelelő hosszúságú lecsengést, a mindentáteresztők szerepe pedig a visszaverődések számának megsokszorozása.

A zengető már elég jónak mondható volt kis lecsengési idők esetére. Nagyobb utózengési időhöz már nem volt tökéletes. Több nem várt jelenség is hallható volt, például:

- a hang fokozatos elhalása helyett lüktető reflektált hangok,
- a hangszín megváltozása.

Később Moorer újragondolta Schroeder zengetőjét, és néhány módosítást javasolt. Az egyik, hogy négyről hatra növelte a fésűszűrők számát, hogy hosszabb utózengést érjen el vele, és növelje a módussűrűséget. Másrészt pedig minden fésűszűrő visszacsatoló ágába elhelyezett egy-egy elsőfokú aluláteresztő szűrőt abból a fizikai megfontolásból, hogy a hang energiájának egy részét a terjedés közben a levegő elnyeli. Ez magasabb frekvenciákon kisebb lecsengési időt produkált, valósághűbbé téve az effektust. Ugyan, ezeknek köszönhetően jobbnak bizonyult az eredeti Schroeder-féle változattól, a lüktetést és a hangszínváltozást ez sem tudta teljesen kiküszöbölni.

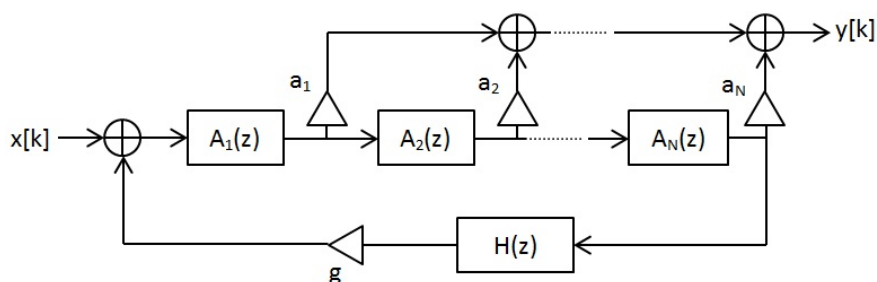
Léteztek olyan elképzelések is a feladat megoldásához, amelynél kicsit másfelől közelítették meg a problémát, és elsősorban a mindentáteresztő szűrőre fókuszáltak, mint alap építőelemre. A legegyszerűbb megvalósítás néhány ilyen elem sorba kapcsolása volt. Moorer kísérlete során viszont több problémát is felvetett. Minél magasabb a hálózat rendje, annál lassabban alakul ki megfelelő reflexiósűrűség. Továbbá az elhalás „simasága” erősen függ a késleltetések és az erősítések megválasztásától. A kimeneten pedig, különösen impulzusszerű hangok esetén, furcsa hangzást okoz, nagyon megváltozik a hangszín.

A továbbgondolások egyike volt, hogy a szűrő késleltetőjét helyettesítsük egy késleltető és egy másik mindentáteresztő soros kombinációjával, létrehozva egy úgynevezett beágyazott mindentáteresztő szűrőt. Időtartományban ez egy előnyös konstrukció, mivel a belső szűrő által létrehozott visszhangok a külső vonalon keresztül



vissza vannak vezetve a bemenetre. Így a valós termékekhez hasonlóan a visszaverődések sűrűsége időben növekszik.

Gardner előállt egy struktúrával [8], melynek a kimenete sorba kapcsolt mindentáteresztő szűrők kimeneteinek az összege, az utolsó mindentáteresztő kimenete pedig egy aluláteresztő szűrőn és egy erősítésen keresztül visszacsatolódik a bemenetre. Ennek a blokkvázlata a 2.10. ábrán látható.  $A_1(z)$ ,  $A_2(z)$ ,  $A_N(z)$  maguknak a mindentáteresztőknek az átviteli függvényei,  $H(z)$  pedig az aluláteresztő szűrő átviteli függvénye. Gardner tapasztalata alapján, ha a mindentáteresztők kimenete megfelelő késleltetéssel van visszacsatolva a bemenetre, akkor nagy mértékben csökken a kellemetlen hangzás.



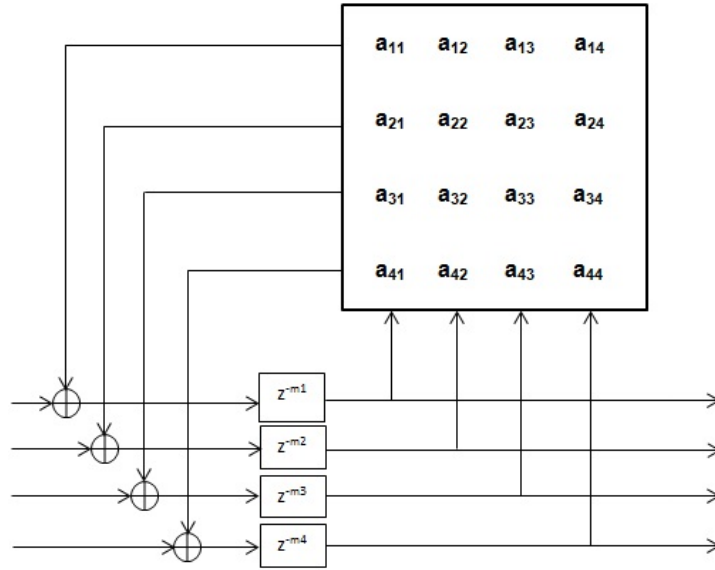
2.10. ábra. Gardner-féle mindentáteresztő szűrő alapú zengető

### 2.3.6. Feedback Delay Network

A Feedback Delay Network (visszacsatolt késleltetések hálózata) egy olyan rendszer, amely tulajdonképpen a fésű-, illetve a mindentáteresztő szűrő többszörös, vektoros általánosítása. A visszacsatolt késleltetések komplexebb hálózatot alkotnak úgy megtervezve, hogy a rendszer veszteségmentes legyen, azaz a kimeneti vonalakon megőrizzük a bemenő jelek összenergiáját.

A Stautner-Puckette zengető [8] négycsatornás, négy késleltetővonalból és egy 4x4-es visszacsatoló mátrixból álló rendszer. A visszacsatoló mátrixnak köszönhetően az összes késleltetés kimenete visszajut az összes késleltető bemenetére. Az elrendezés, melyet az 2.11. ábra szemléltet, visszavezethető a fésűszűrőre, pontosan annak a négy bemenetű, négy kimenetű kiterjesztett változata.

A zengető biztosan stabilis, ha a visszacsatoló mátrixot úgy választjuk meg, hogy az egy ortogonális mátrix szorzata egy  $g$  együtthatóval, ahol  $|g| < 1$ . A  $g$  határozza meg az utózungési időt. Ilyen például a következő mátrix:



2.11. ábra. Stautner és Puckett négycsatornás FDN zengetője

$$\mathbf{A} = \frac{g}{\sqrt{2}} \begin{bmatrix} 0 & 1 & 1 & 0 \\ -1 & 0 & 0 & -1 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & -1 & 0 \end{bmatrix}. \quad (2.8)$$

Stautner és Puckett jelezték, hogy a rendszerre jellemző a lüktető jellegű utózengetés és a hangszín módosítása. Utóbbi csökkentése végett javasolták a késleltetővonalak hosszának véletlenszerű változtatását.

Jot a fenti FDN struktúrát alkalmazva alkotta meg az egy bemenettel és egy kimenettel rendelkező zengetőt. Fontos tulajdonságai, hogy ezzel időben és frekvenciában is tetszőleges sűrűséget lehet elérni, miközben a hangszín sem módosul. A módosítás annyiból áll, hogy ugyanazt a bemenő jelet küldi az összes késleltetővonal bemenetére, a kimeneteket pedig összegzi.

## 3. fejezet

# Feedback Delay Network alapú zengető fejlesztése

Az alábbi fejezetben ismertetem, hogyan valósítottam meg a szakasz modellezését. Ehhez a fésűszűrő alapú veszteségmentes FDN-t választottam, az energia elnyelését pedig úgynevezett veszteségi szűrők által vettem figyelembe.

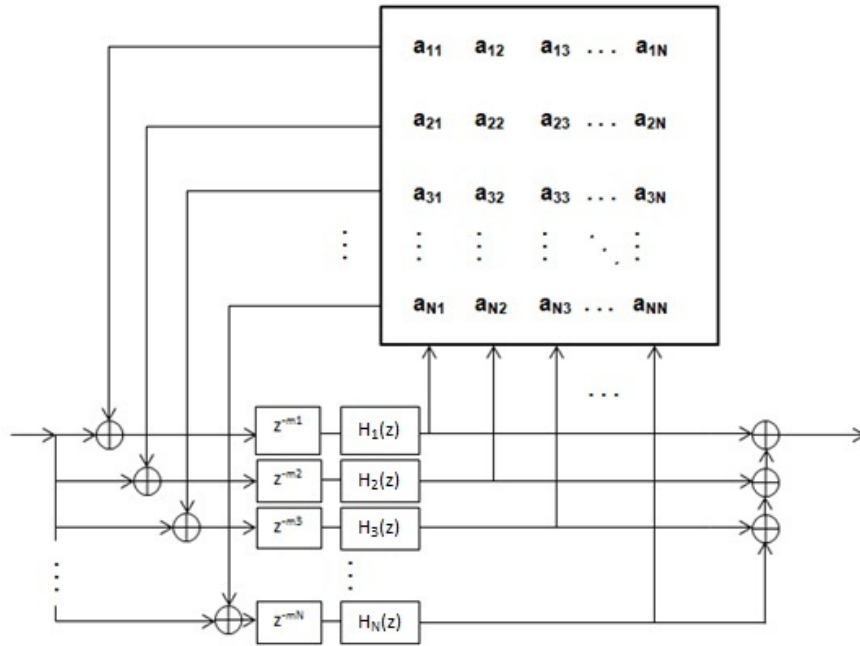
### 3.1. Jot és Chaigne konstrukciója

A veszteségmentes prototípus, mint az utőzengést megvalósító hálózat, megfelelő választás, többek között az előző fejezet végén említett tulajdonságok miatt is. Az 2.11. ábrán látható FDN kiterjeszthető négyenél több késleltetővonalra. A 3.1. ábra ezt szemlélteti, ha  $N$  darab késleltetésünk van.

Ha a veszteségi szűrőktől eltekintünk, akkor ez a rendszer veszteségmentes, ha  $\mathbf{A}$  mátrixot úgy választom meg, hogy az ortogonális legyen [8]. A megoldáshoz egy speciális mátrixot, az úgynevezett Householder mátrixot alkalmaztam. A mátrix általános formulája:

$$\mathbf{A}_N = \mathbf{J}_N - \frac{2}{N} \mathbf{u}_N \mathbf{u}_N^T, \quad (3.1)$$

ahol  $\mathbf{J}_N$  permutációs mátrix,  $\mathbf{u}_N$  pedig  $N$  hosszúságú 1-esekből álló oszlopvektor. Speciális eset, amikor  $\mathbf{J}_N$  éppen az egységmátrix. Ha ez fennáll, akkor a 3.1 egyenlet átírható az alábbi módon:



3.1. ábra. Jot és Chaigne zengetője, veszteségi szűrőkkel

$$\mathbf{A}_N = \mathbf{E}_N - \frac{2}{N} \begin{bmatrix} 1 & 1 & \dots & 1 \\ 1 & 1 & \dots & 1 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & \dots & 1 \end{bmatrix}. \quad (3.2)$$

Világos, hogy  $\mathbf{A}_N$  elemei két különböző értéket vehetnek föl. Például legyen  $N = 4$ . Ha az egységmátrixból kivonjuk a csupa 0.5 értékű elemekből álló mátrixot, akkor  $\mathbf{A}_N$  főátlójának elemei 0.5-re adódnak, az összes többi -0.5 értékű lesz.

A saját zengetőm 32 darab késleltetővonalból áll, ami azt jelenti, hogy a főátlókban 0.9375, egyéb helyeken pedig -0.0625 értékeket vesz föl. A módussűrűség növekszik a késleltetővonalak számának a növelésével, tehát következtetésem szerint ez egy megfelelő konstrukció. A kérdés, hogy a számítógéptől mekkora teljesítményre van szükség, hogy az ehhez szükséges műveleteket valós időben el tudja végezni.

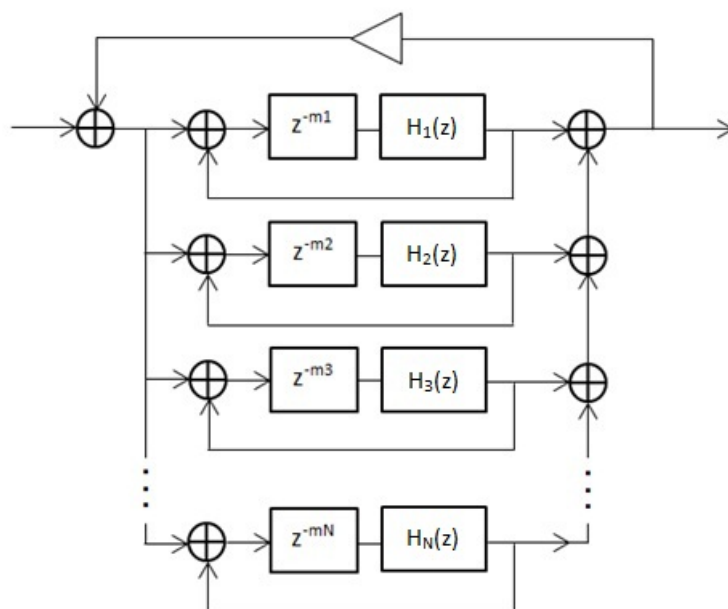
### 3.2. A FDN egyszerűsítése

Egy 32 késleltetővonalból álló hálózat visszacsatoló mátrixa is 32-ed rendű. Ez azt jelenti, hogy a processzornak  $32^2 = 1024$  szorzást kellene elvégeznie egyetlen mintára. Ezt önmagában ma már egy nagyteljesítményű számítógép lehet, hogy képes kezelni, de mivel az egész effekt magában foglalja:

- a korai visszaverődéseket,
- a hangszórók közötti panorámázást,
- és a kis korreláltságú kimenetek előállítását,

tehát összességében nagyon sok műveletet kell elvégezni.

Azonban van lehetőség a konstrukció számításigényének minimalizálására. Az egyszerűsítés a Householder mátrix [8] alapján elvégezhető. Az egységmátrix jelentése az, hogy az  $i$ -edik késleltetés kimenetét visszacsatolom önmagába ( $1 \leq i \leq 32$ ). A késleltetővonalak kimeneteinek összegzése és  $-\frac{2}{N}$  súllyal visszavezetése a hálózat bemenetére pedig az egyenlet második tagjának felel meg.



**3.2. ábra.** Az egyszerűsített FDN alapú zengető

Az egyszerűsített felépítés látható a 3.2. ábrán. Ugyanazt a funkciót látja el, mint az eredeti veszteségmentes zengető, a számításigényben viszont nagy a különbség. Mintánként több, mint ezredrészére csökkent a szorzás száma, ugyanis a késleltetővonalak számától függetlenül csak egy szorzást kell elvégezni. Ez egy modern PC-nek nem okoz gondot az effekt egyéb műveletei mellett sem.

### 3.3. A késleltetővonalak hosszának meghatározása

Kiválasztottam tehát azt a zengető algoritmust, amit a VST plug-inban fogok implementálni. Arról viszont még nem volt szó, hogy az egyes késleltetővonalak hosszát miként választom meg. A dolgozat korábbi részében esett már szó a szabad átlagos

úthosszról, mint az akusztikai célú terem egyik legfontosabb tulajdonságáról, melyet téglatest alakú termeknél a  $d = \frac{4V}{S}$  egyenlettel meg lehet határozni.

A szabad átlagos úthosszt a hang sebességével elosztva megkapom azt az átlagos időtartamot, ameddig egy hanghullám két falról történő reflexió között terjed. Az időt mintákba átszámolva a késleltetővonalak átlagos hossza:

$$\bar{m} = \frac{d}{c} f_s. \quad (3.3)$$

Ez lesz a késleltetővonalak átlagos hossza. Ezt betartva, bizonyos mértékig szabad kezet kapok az egyes késleltetések értékeinek a megválasztásához. Ezt úgy oldottam meg, hogy harminckét véletlenszerűen választott egész számot egy tömbben eltároltam. Mind a harminckét érték különbözik egymástól, a legkisebbé 1153, a legnagyobb pedig 2957. A harminckét eltárolt elemnek kiszámoltam az átlagát. Ez arra jó, hogy most már létezik referencia mind az egyes késleltetővonalak hosszához, mind pedig  $\bar{m}$ -hez. Legyen a tömb elemeinek a jelölése  $d_i^{ref}$ , az átlagé pedig  $\bar{m}^{ref}$ . Az  $\bar{m}$ -et a 3.3 alapján már kiszámítottam. Amire kíváncsi vagyok, az az  $i$ -edik késleltetés nagysága. Ezt az alábbi módon határoztam meg:

$$d_i = \left[ \frac{\bar{m}}{\bar{m}^{ref}} d_i^{ref} \right], \quad (3.4)$$

vagyis a modellezett terem méretéből meghatározott átlagos késleltetővonalhossz és a referencia átlag hányadosával skáláztam a tömb elemeit, és ennek az alsó egész részét kapja értéként  $d_i$ .

Például van egy 40x25x10 m nagyságú téglatest alakú termünk. Az átlagos szabad úthossz értéke:

$$d = \frac{4V}{S} = 4 \cdot \frac{40 \cdot 25 \cdot 10}{2 \cdot 40 \cdot 25 + 2 \cdot 40 \cdot 10 + 2 \cdot 25 \cdot 10} = 12.12m,$$

ebből az átlagos késleltetővonal hossz  $f_s=44100$  Hz esetén:  $\bar{m} = 1553.76$ , kerekítve 1554. Legyen a referencia átlag 1000, ekkor  $d_i = 1.554 \cdot d_i^{ref}$ , minden  $d_i$ -re.

### 3.4. A veszteségi szűrők

Egy VST zengetőnél alapvető elvárás, hogy a lecsengési idő (decay time) változtatható paraméter legyen. A lecsengési idő az a zengés kezdetétől számított idő, mely alatt a hang intenzitása 60 dB-lel csökken. A lecsengés ideje a frekvenciától is függ,

modellünkben a vágási frekvenciával állítjuk be. A vágási frekvencia az a frekvencia, amelyen a lecsengési idő az alacsony frekvenciás lecsengés idejének éppen a fele, melyet szintén tudunk módosítani, amikor a plug-int használjuk.

Visszatekintve a 3.2. ábrára, volt arról szó, hogy ez egy veszteségmentes rendszer. Az ábrán minden késleltetővonal kimenete egy  $H_i(z)$  szűrő bemenetére van kötve. Ezeket veszteségi szűrőknek nevezzük, melyek jellemzően aluláteresztő szűrők, ezek az energiveszteség modelljei. Az elsőfokú aluláteresztő szűrő átviteli függvénye:

$$H(z) = g \frac{1 + a}{1 + az^{-1}}, \quad (3.5)$$

ahol a  $g$  és az  $a$  paraméterek módosításával lehet változtatni a zengetés fenti két tulajdonságát.

### 3.4.1. A lecsengési idő

A kisfrekvenciás lecsengést  $g$ -vel egyértelműen meg tudom határozni. A szűrőt  $-60$  dB-re specifikálom,  $z$  helyére beírva  $e^{j\vartheta} = e^{j\frac{2\pi f}{f_s} - t}$  [8]:

$$20 \lg \left| H_i(e^{j\frac{2\pi f}{f_s}}) \right| = \frac{-60d_i}{T_d f_s}, \quad (3.6)$$

ahol  $T_d$  a kisfrekvenciás lecsengési idő,  $H_i$  a  $d_i$  késleltetéshez tartozó veszteségi szűrő átviteli függvénye. Kisfrekvenciás lecsengésnél a szűrő amplitúdóját csak a  $g$  határozza meg,  $H_i$ -t helyettesíthetjük  $g_i$ -vel. Az egyenletet átrendezve meghatározható a szűrők DC erősítése:

$$g_i = 10^{-\frac{3}{T_d f_0}}. \quad (3.7)$$

Az egyenletben  $f_0$  a mintavételi frekvencia és a  $d_i$  késleltetés hányadosa. A VST-ben  $T_d$  a közvetlenül állítható paraméter, ha ezt megváltoztatom, akkor host program meghívja azt a függvényt, amelyikben  $g_i$ -re a számítást el tudja végezni. A lecsengés idejét 0 és 10 s között tudom beállítani.

### 3.4.2. A vágási frekvencia

A vágási frekvenciát közelítéssel határozom meg, ehhez pedig a kiindulópont a húros hangszerek fizikai alapú hangszintézise [6]. A hangszer lecsengési idejét ugyanezzel a szűrővel lehet modellezni. A vágási frekvencia pontos meghatározásához a szűrő

specifikációja így néz ki:

$$T(f) = -\frac{1}{f_0 \ln \left| H_i \left( e^{j \frac{2\pi f}{f_s}} \right) \right|}. \quad (3.8)$$

A frekvenciától függő lecsengési idő meghatározása közelítéssel:

$$T(f) = T_d \frac{1}{1 + \left( \frac{f}{f_c} \right)^2}, \quad (3.9)$$

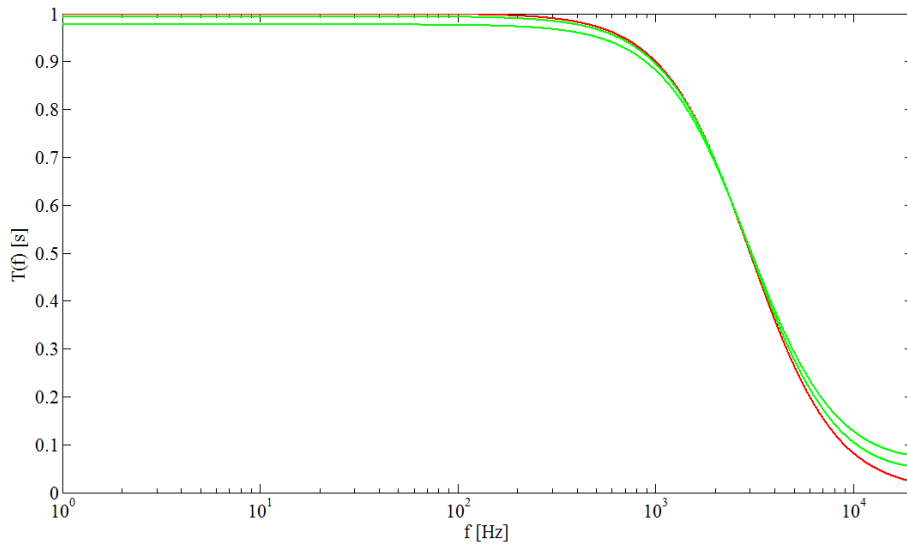
ahol  $f_c$  a vágási frekvencia. Keresendő az  $a$  értéke. Bevezettem a  $b_3$  paramétert [6]:

$$b_3 = -\frac{f_0}{f_s^2} \frac{a}{2(a+1)^2}, \quad (3.10)$$

melyet ha  $a$ -ra megoldok, akkor a következő egyenletet kapom:

$$a = -1 - \frac{f_0}{4b_3 f_s^2} + \sqrt{f_0 2b_3 f_s^2 + \frac{f_0}{4}} \cdot \frac{1}{2b_3 f_s^2}. \quad (3.11)$$

A 3.11 kifejezésben  $b_3 = (2\pi f_c^2)/T_d$  formulát [6] behelyettesítve egyértelműen meghatározom  $a$  értékét a lecsengési időből. Az egyenlet hosszú és bonyolult, ugyanakkor  $a$  pontos meghatározásának nehézsége miatt indokoltá vált a közelítés.



**3.3. ábra.** *Lecsengési idő frekvenciafüggése közelítéssel*



A 3.3. ábrán látható a lecsengési idő frekvenciafüggésére kapott eredmény. A piros vonal a pontos, átviteli függvényből számított érték, a zöld vonalak pedig 500 és 2000 hosszúságú késleltetésekre kapott közelített értékek,  $T_d = 1s$ , és  $f_c = 3kHz$  esetén. A húros hangszerek szintézisére alkalmazott közelítés megfelelőnek bizonyul zengetés esetére is.

## 4. fejezet

# Korai visszaverődések megvalósítása

### 4.1. A paraméterek meghatározása

Több kimeneti csatorna alkalmazása realiztikusabb hangzást ad, ha a kimenő jeleket a korai visszaverődések irányának megfelelően osztjuk ki az egyes hangszórók között. Az első fejezetben bemutatott tükörforrások modelljét alkalmazva a visszaverődések késleltetésének a nagyságán kívül azt is meg kell határozni, hogy honnan jönnek az egyes hangok. Az irány ismeretében alkalmazok olyan számítási módszert, mely megadja, hogy mely hangszórókon, és milyen intenzitással szólaljon meg az adott hang.

#### 4.1.1. A késleltetővonalak hossza

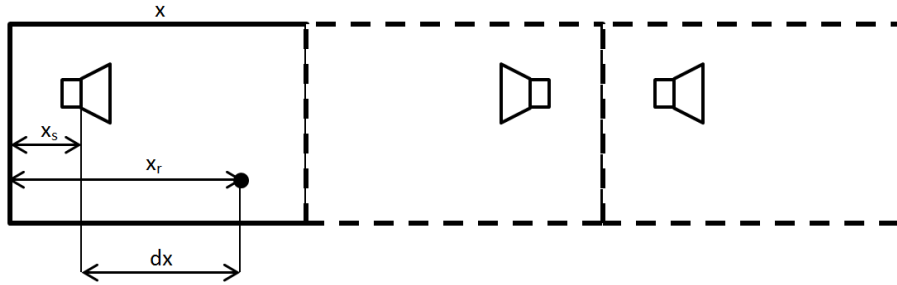
A hangszóró és a hallgató helyzetét előre definiáltam, ahol a hangszóró pozícióját  $x_s$ ,  $y_s$  és  $z_s$ , a hallgatóét pedig  $x_r$ ,  $y_r$  és  $z_r$  koordináták adják meg. A kettő közötti távolságot a pozíciók ismeretében ki tudom számolni:

$$d = \sqrt{(dx)^2 + (dy)^2 + (dz)^2} = \sqrt{(x_s - x_r)^2 + (y_s - y_r)^2 + (z_s - z_r)^2}. \quad (4.1)$$

A feladat a tükörforrások pozícióinak meghatározása, melyekből a távolságok kiszámíthatók. A 4.1. ábrán szemléltetem, hogyan néz ki ez az  $x$  irányú komponens esetére.

A „jobbra” tükrözés pozitív  $x$  irányt jelent, „balra” pedig negatívát. Fontos tudni, hogy milyen irányban indulok el, továbbá lényeges információ az is, hogy abban az irányban a tükrözések száma páros vagy páratlan. Nézzük azt az esetet, amikor  $+x$  irányban egyszer tükrözök. Az eredeti és a tükrözött forrás közti távolság  $2x - 2x_s$ , ahol  $x$  a terem hossza. Ehhez  $dx$ -et hozzáadva kapjuk meg a hallgató és a tükörforrás közötti távolságot. Fontos az előjelre figyelni! Ha  $x_r$  nagyobb, mint  $x_s$ , akkor  $dx$

negatív. Számolhatunk  $dx = x_r - x_s$  alakkal is, akkor viszont már így jön ki a végeredmény:  $x_{rs} = 2x - 2x_s - dx$ .



4.1. ábra. Korai reflexiók számítása

A tükrözést még egyszer alkalmazva a források közötti távolság  $2x$ , a hallgató és a tükrörforrás távolsága  $x_{rs} = 2x + dx$ . Az elv ugyanez magasabb rendű reflexióknál is, csak a tükrözések száma változik. Ezért a számítás módja is általánosítható. Legyen  $r_x$  a  $+x$  irányú tükrözések száma. A tükrözött forrás és a hallgató közötti távolság meghatározásához az alábbiak szerint kell eljárni:

$$x_{rs} = \begin{cases} (r_x + 1)x - 2x_s + dx, & \text{ha } r_x \text{ páratlan,} \\ r_x x + dx, & \text{ha } r_x \text{ páros.} \end{cases} \quad (4.2)$$

Negatív irányba tükrözés esetén, a képletet a 4.2-höz hasonló elv alapján határoztam meg:

$$x_{rs} = \begin{cases} -((r_x - 1)x + 2x_s - dx), & \text{ha } r_x \text{ páratlan,} \\ -(r_x x - dx), & \text{ha } r_x \text{ páros.} \end{cases} \quad (4.3)$$

A keresett távolság  $y$  és  $z$  irányú összetevőire a formulák ugyanúgy érvényesek, ha az összes paraméternek az adott irányban megfeleltetett komponenseit helyettesítem be. A hallgató és a tükrözött forrás közötti távolság végül a következő módon számolható:

$$l = \sqrt{(x_{rs})^2 + (y_{rs})^2 + (z_{rs})^2}. \quad (4.4)$$

A levezetés eredményeképpen kiszámítható, hogy egy reflektált hanghullám mekkora utat tesz meg a hang forrásától a hallgatóig. Ez a távolság a közvetlenül megtett út hosszához képest ad érdemi információt. Ezért a késleltetések számát mintákban

a két távolság különbségéből határoztam meg:

$$k = \frac{l-d}{c} f_s, \quad (4.5)$$

ahol  $c = 344 \text{ m/s}$  a hangsebesség, és  $f_s = 44\,100 \text{ Hz}$  a mintavételi frekvencia.

### 4.1.2. A visszaverődések iránya

A reflektált hangok által megtett  $l$  távolság ismert. Az előző részben bemutatott számítási eljárásnál  $l$ -t  $x$ ,  $y$  és  $z$  irányú összetevőkre bontottam. Ez azért jó, mert így a megtett út vektor formájában is felírható:  $\mathbf{l} = [x_{rs}, y_{rs}, z_{rs}]$ . A vektor abba az irányba mutat, ahonnan a hang utóujára visszaverődött a falról.

A hangszórókra történő panorámázáshoz ismerni kell a visszaverődések, illetve a hangszórók irányába mutató egységvektorokat. Ha  $\mathbf{l}_i$  az  $i$ -edik reflexióhoz tartozó vektor, az egységvektort a következőképpen kapom meg:

$$\mathbf{p}_i = \frac{1}{l} [x_{rs}, y_{rs}, z_{rs}]. \quad (4.6)$$

Ha ismerem az összes hangszóró helyzetét a hallgatóhoz képest, akkor ugyanígy legenerálhatók a hozzájuk tartozó egységvektorok.

### 4.1.3. Frekvenciafüggő elnyelés modellezése

A falak általi elnyelés szimulálásához ugyanazt az elsőfokú aluláteresztő szűrőt alkalmazom, amelyiket a veszteségi paraméterek beállításához a Feedback Delay Network esetén. A kevesebb műveletvégzés végett minden késleltetővonalhoz ugyanazok a szűrőparaméterek tartoznak. A plug-in mindegyik késleltetett jelről tudja, hogy az hányadrendű. Elsőfokú visszaverődés esetén a késleltetett jelet egyszer, másodfokúnál pedig kétszer szűröm, vagyis az utóbbi annak felel meg, hogy két sorba kapcsolt elsőfokú aluláteresztő szűrőt alkalmazok.

A szűrőt a plug-in két állítható paraméterével állítom be: a  $g$  erősítés megegyezik a korai visszaverődések erősítési tényezőjével (Echo), a határfrekvencia pedig az Echo Cut paraméterrel változtatható. A szűrő  $-3 \text{ dB}$ -hez tartozó határfrekvenciáját a 3.5 átviteli függvény alapján specifikáltam:

$$|H(e^{j\vartheta})| = \frac{1+a}{\sqrt{1+a^2+2a\cos\vartheta}} = \frac{1}{\sqrt{2}}. \quad (4.7)$$

Az egyenletből a szűrő  $a$  paraméterét kell kifejezni:

$$a = -2 + \cos \vartheta + \sqrt{3 + \cos^2 \vartheta - 4 \cos \vartheta}, \quad (4.8)$$

$\vartheta = \frac{2\pi f_c}{f_s}$  behelyettesítéssel a program elvégzi a számítást  $f_c$  frekvenciára.

## 4.2. Panorámázás a hangszórók között

Az audió jelfeldolgozásban egy klasszikus feladat a térhatás megvalósítása. Egy hangszóró használata esetén a látszólagos hangforrás pontszerű. Innen nagy lépés volt a kétcsatornás rendszerek alkalmazása: a hangforrás kiterjedt a két hangszórót összekötő vonalra. Ezután számos kísérletet végeztek a térhatás hatékonyabb kivitelezésére oly módon, hogy kettőnél több hangszórót alkalmaztak.

A feladat virtuális hangforrások pozíciójának a meghatározása, függetlenül a hangszórók elrendezésétől. A probléma megoldásához ismertetek egy módszert, mely alkalmazható tetszőleges két- és háromdimenziós elrendezésre, bármennyi hangszóró esetén. Az eljárás neve vector base amplitude panning (VBAP) [9]. A VBAP alkalmazásának egyik feltétele, hogy a hangszórók a hallgatótól körülbelül egyforma távolságra legyenek.

### 4.2.1. VBAP két dimenzióban

Az egyszerű panorámázásnál két hangszóró összefüggő hangjelet sugároz ki, általános esetben eltérő amplitúdóval. Amit ezáltal érzékelünk az egy illúzió: a helyzetünkől és a két hangszóró helyzetétől függően olyan benyomásunk lesz, mintha a hang forrása valahol a két hangszóró között helyezkedne el. A virtuális forrás iránya függ a hangszórókon kimenő jelek amplitúdóinak arányától.

#### Trigonometriai megközelítés

A kétcsatornás hangrendszert a mai napig gyakran alkalmazzák. A panorámázás egyszerűen megvalósítható ilyen konfiguráción. Az elrendezés a 4.2. ábrán látható. A fekete kör a hallgató pozíciója. A kimeneti hangszórókat jelöli a  $ch$  1 és a  $ch$  2, a kettő között pedig az érzékelt virtuális forrás látható. A pozíciókat a hangszórók, illetve a virtuális hangforrás irányába mutató egyenes és az  $x$  tengely által bezárt irányszögek adják meg:  $\pm\varphi_0$  a két hangszóróhoz tartozó szög,  $\varphi$  pedig a virtuális forrás irányáról ad információt, ahol  $0^\circ < \varphi_0 < 90^\circ$ , és  $-\varphi_0 < \varphi < \varphi_0$ .

Fülünkkel a hang irányát két mennyiség alapján vagyunk képesek érzékelni: a két fülhöz eljutó hangjelenség intenzitásának a különbsége (ILD, interaural level diffe-

rence) és a hanghullám véges terjedési sebességéből adódó idők különbsége alapján, mely idők elteltével a két fülünk érzékeli a hangot (ITD, interaural time difference). Blumlein ezen jellemzők alapján határozta meg a szinuszos formulát, mely alkalmazható a hangszórók panorámázására [9]:

$$\frac{\sin \varphi}{\sin \varphi_0} = \frac{g_1 - g_2}{g_1 + g_2}, \quad (4.9)$$

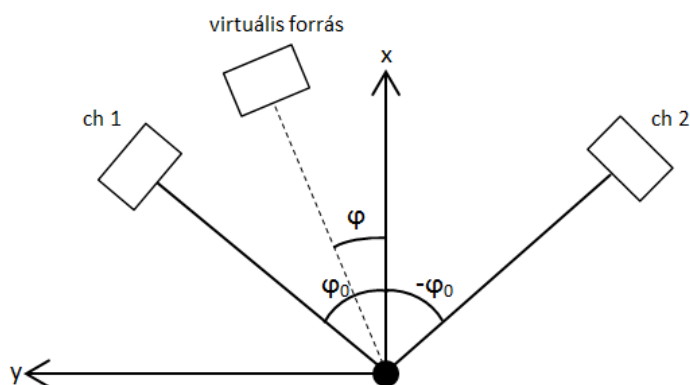
ahol  $g_1, g_2 \in [0, 1]$  a *ch 1* és a *ch 2* kimeneti jelek erősítése. Az összefüggés érvényes akkor is, ha azt a szögek tangenseivel írjuk föl [9]:

$$\frac{\tan \varphi}{\tan \varphi_0} = \frac{g_1 - g_2}{g_1 + g_2}. \quad (4.10)$$

A fenti két összefüggés nem ad információt külön-külön az erősítések nagyságáról, csak az arányukról. Függetlenül attól, hol helyezkedik el a virtuális forrás, a teljesítménye legyen minden esetben ugyanakkora. A feltételt az alábbi módon tudjuk teljesíteni [9]:

$$g_1^2 + g_2^2 = C, \quad (4.11)$$

ahol a  $C$  konstans az eredő hang intenzitását határozza meg. Kézenfekvő  $C = 1$  választása.



4.2. ábra. Kétsatornás elrendezés

## Vektoros megközelítés

A tangens tételt kielégítő megoldás, ha az alábbi vektoros bázisban végezzük el a számítást. A vektoros bázis definiálása egységnyi hosszú vektorokkal történik, melyek a *ch* 1, illetve *ch* 2 hangszórók irányába mutatnak:  $\mathbf{l}_1 = \begin{bmatrix} l_{11} & l_{12} \end{bmatrix}^T$ , és  $\mathbf{l}_2 = \begin{bmatrix} l_{21} & l_{22} \end{bmatrix}^T$ . A virtuális forrás irányába mutató egységvektor felírható  $\mathbf{l}_1$  és  $\mathbf{l}_2$  lineáris kombinációjaként [9]:

$$\mathbf{p} = g_1 \mathbf{l}_1 + g_2 \mathbf{l}_2. \quad (4.12)$$

Az egyenletben  $g_1$  és  $g_2$  erősítések nemnegatív skalárként kezelhetők. Ha bevezetjük  $\mathbf{g} = \begin{bmatrix} g_1 & g_2 \end{bmatrix}$  vektort és  $\mathbf{L} = \begin{bmatrix} \mathbf{l}_1 & \mathbf{l}_2 \end{bmatrix}^T$  mátrixot, az egyenlet fölírható az alábbi alakban:

$$\mathbf{p}^T = \mathbf{g} \mathbf{L}. \quad (4.13)$$

A hangszóró-elrendezést ismerem, a virtuális forrás pozíciója felé mutató vektor pedig a feladat szerint a korai visszaverődések irányába mutat, amelyeket szintén meghatároztam. A visszavert hanghullámok teljesítményét kell elosztani a kimenetek között, vagyis a 4.13 egyenletet  $\mathbf{g}$ -re kell megoldanom:

$$\mathbf{g} = \mathbf{p}^T \mathbf{L}^{-1} = \begin{bmatrix} p_1 & p_2 \end{bmatrix} \begin{bmatrix} l_{11} & l_{12} \\ l_{21} & l_{22} \end{bmatrix}^{-1}. \quad (4.14)$$

A megoldás létezik, ha  $\mathbf{L}$  invertálható, ez pedig teljesül minden esetben, ha  $\varphi_0 \neq 0^\circ$  és  $\varphi_0 \neq 90^\circ$ . Mindkét elrendezés olyan, melyet gyakorlatban nem használunk. Az erősítéseket végül normalizálni kell, hogy a 4.11 egyenlet teljesüljön:

$$\mathbf{g}^{norm} = \frac{\sqrt{C} \mathbf{g}}{\sqrt{g_1^2 + g_2^2}}. \quad (4.15)$$

Ha konfiguráció kettőnél több hangszóróból áll, a módszer ugyanúgy alkalmazható. Legyen a kimeneti csatornák száma  $N$ . Minden hangszóróhoz tartozik egy kétdimenziós egységvektor. A  $\mathbf{l}_i$  vektor az  $i$ -edik hangszóró irányát adja meg ( $0 < i \leq N$ ). A virtuális hangforrás teljesítményét ugyanúgy két csatorna között osztom ki, a kérdés az, hogy melyik kettőt választom ki.

A kiválasztást a  $\mathbf{p}^T$  és a  $\mathbf{l}_i^T$  vektorok alapján végeztem el. A reflexió irányát  $\mathbf{p}^T$ ,

az  $i$ -edik hangszóró helyzetét  $\mathbf{l}_i^T$  jelöli. Definiáljuk  $\mathbf{d}_i$  vektort az alábbi módon:

$$\mathbf{d}_i = \begin{bmatrix} d_{i1} & d_{i2} \end{bmatrix} = \mathbf{p}^T - \mathbf{l}_i^T. \quad (4.16)$$

A műveletet minden hangszóróra elvégezve kapok  $N$  darab különbségvektort. A lényeges információ  $\mathbf{d}_i$  hossza, azaz  $d_i = \sqrt{d_{i1}^2 + d_{i2}^2}$ . Ezek az értékek tulajdonképpen az  $i$ -edik hangszóró és a virtuális forrás közti relatív távolságot adják meg. Azt a két hangfalat választom ki, amelyekre a két legkisebb  $d_i$  értéket kapom. Ha a különbségvektorok abszolút értékeit egy tömbben eltárolom, egy egyszerű rendezési algoritmussal megtalálom a két legkisebb távolsághoz tartozó  $\mathbf{l}_i$  vektort.

### 4.2.2. VBAP három dimenzióban

A kétdimenziós módszer általánosítható a térben is. Ez úgy képzelhető el, hogy a hangszórókat egy képzeletbeli gömb felületén helyezzük el. Alkalmazásához legalább három kimeneti csatornával rendelkezünk kell, mert egy virtuális forrás hangenergiáját három hangszóró között osztjuk szét. A teljesítményre vonatkozó feltételt az alábbi összefüggés írja le:

$$g_1^2 + g_2^2 + g_3^2 = C. \quad (4.17)$$

A vektorokat ugyanúgy adjuk meg a hangszórókra, és a virtuális forrásokra, a különbség annyi, hogy kiegészítjük egy harmadik komponenssel is [9]:

$$\mathbf{p} = \begin{bmatrix} p_1 & p_2 & p_3 \end{bmatrix}^T, \quad (4.18a)$$

$$\mathbf{l} = \begin{bmatrix} l_1 & l_2 & l_3 \end{bmatrix}^T. \quad (4.18b)$$

A kétdimenziós eset analógiájára kifejezhető a virtuális forrás irányába mutató vektor  $\mathbf{l}_1$ ,  $\mathbf{l}_2$  és  $\mathbf{l}_3$  lineáris kombinációjaként:

$$\mathbf{p} = g_1\mathbf{l}_1 + g_2\mathbf{l}_2 + g_3\mathbf{l}_3, \quad (4.19a)$$

$$\mathbf{p}^T = \mathbf{g}\mathbf{L}, \quad (4.19b)$$

ahol  $\mathbf{L} = \begin{bmatrix} \mathbf{l}_1 & \mathbf{l}_2 & \mathbf{l}_3 \end{bmatrix}^T$ . Az egyenletet  $\mathbf{g}$ -re rendezve kapjuk meg a következő formulát:



$$\mathbf{g} = \mathbf{p}^T \mathbf{L}^{-1} = \begin{bmatrix} p_1 & p_2 & p_3 \end{bmatrix} \begin{bmatrix} l_{11} & l_{12} & l_{13} \\ l_{21} & l_{22} & l_{23} \\ l_{31} & l_{32} & l_{33} \end{bmatrix}^{-1}, \quad (4.20)$$

amely megoldható, ha létezik  $\mathbf{L}^{-1}$ . Ez teljesül, ha  $L$  mátrix által definiált vektor bázis három dimenziós teret feszít ki. [9]. A  $g$  vektort itt is skálázni kell:

$$\mathbf{g}^{norm} = \frac{\sqrt{C} \mathbf{g}}{\sqrt{g_1^2 + g_2^2 + g_3^2}}. \quad (4.21)$$

A hangszórók kiválasztásánál is ugyanúgy járok el, mint a kétdimenziós esetben, tehát  $\mathbf{d}_i = \mathbf{p}^T - \mathbf{l}_i^T$  vektorok abszolút értékeiből a három legkisebbhez tartozó hangszóróra panorámázok.

## 5. fejezet

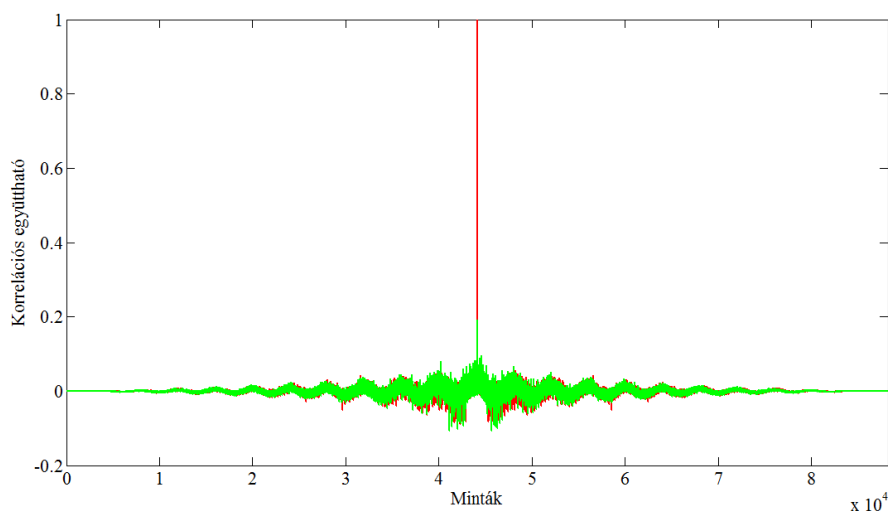
# A diffúz szakasz kimeneteinek korrelálatlansága

Ebben a fejezetben foglalkozom a Feedback Delay Network kimeneti jelének hangszórókra történő kiosztásával. A modell csak a terem méreteit veszi figyelembe, a hanghullámok irányát nem, ezért itt közvetlen irányszámításra nincs lehetőség. Vannak viszont egyéb módszerek arra, hogy a térhatást növeljük. Az egycsatornás zengetőt úgy valósítottam meg, hogy a kimenete egyszerűen a késleltett jelek összege.

A korrelálatlanság növelésére az egyik módszer kiindulópontja a Stautner-Pucket-féle négy bemenetű és négy kimenetű zengető. Ez volt tulajdonképpen a VST-ben implementált konstrukció elődje is, mely annyiban különbözött, hogy a késleltetővonalak kimenetei nem összegződtek, hanem mindegyik egy-egy külön csatornát képviselt. Hasonló az az elképzelés, hogy ha sok visszacsatolt késleltetéssel rendelkezem, akkor a kimeneten nem összegzem mindet, hanem csoportokra bontom. Például, ha négy hangszóróra szeretném kiosztani a kimenő jeleket, akkor harminckét késleltetővonalat 4x8-as csoportokra tudok osztani, és egy kimenetre csak minden negyedik késleltett jelet összegzem. A 2.11. ábrán piros színnel ábrázolt értékek az egyik kimenet autokorrelációs együtthatói, ahol a maximum érték 0 eltolásnál 1 értékű. Zöld színnel két kimenet keresztkorrelációs együtthatóit ábrázolom. Ennek a maximális értéke valóban sokkal kisebb, mint az autokorreláció esetében. A 32 késleltetésből álló Feedback Delay Network csatornánkénti impulzusválaszainak a korrelációs együtthatóit számoltattam ki.

Elsőre egyszerűnek tűnik, és a kimenetek közötti korrelálatlanságot is biztosítja. Az 5.1. ábrán látható két kimenet közötti kapcsolat nagysága, ahol az egyik kimenet  $out_1 = y_1 + y_5 + \dots + y_{29}$  módon írható föl, a másik pedig  $out_2 = y_4 + y_8 + \dots + y_{32}$  összegként, ahol  $y$  a késleltetővonalak kimenetei.

A késleltetővonalakat úgy célszerű csoportosítani, hogy egy csoportban a késleltetővonalak hossza közti különbség ne legyen túl kicsi, mert lüktető jellegű zengést



**5.1. ábra.** *Kimenetek közötti kapcsolat a késleltetővonalak 4x8-as csoportokra bontásával*

tapasztaltam. Egyéb problémák is felmerültek a módszer kipróbálása során.

Az egyik a pszichoakusztikából fakad. Azzal, hogy kevesebb késleltetővonalat összegzek, kellemetlenebbé válik a zengés hangzása. Legrosszabb esetben a hangszín is megváltozik.

Másrészt a késleltetővonalakat is valamilyen rendszer alapján ki kell osztani a kimenetekre. Általános esetben a késleltetők és a hangszórók számának a hányadosa nem egész szám, tehát nem tudom egyenlő arányban úgy elosztani a késleltetéseket, hogy egyik se jusson ki több kimenetre egyszerre. Ezzel pedig a korreláltságot növelhetem. A kimenetek előállítására így bonyolultabb algoritmust igényel.

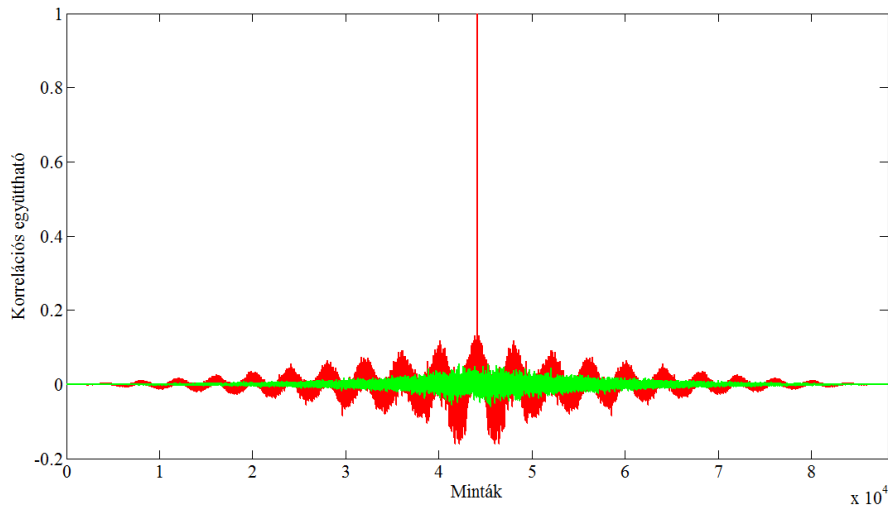
Egy másik lehetőség, hogy a késleltetővonalak kimeneteit nem közvetlenül összegzem, hanem a visszacsatolt késleltetések mögé beteszek egy-egy erősítést. Ha két csatornára különböző erősítési értékeket adok meg, azzal csökkenthető a két kimeneti jel közötti lineáris kapcsolat. Az erősítéseket úgy szeretném megválasztani, hogy a kimenetek korrelátlanságát a lehető legnagyobb mértékben növeljem.

Az alábbiakban  $\mathbf{c}_k$  vektor a  $k$ -adik hangszóróhoz tartozó erősítésekből álló vektor, mely hossza a késleltetővonalak számával egyezik meg. Kétcsatornás hangrendszerhez egy lehetséges megoldás, ha  $\mathbf{c}_1$  minden eleme 1 értékű,  $\mathbf{c}_2$ -t pedig  $\mathbf{c}_1$ -ből képezzük oly módon, hogy minden második elem helyére  $-1$ -t írok be [8]. Az így megadott vektorok ortogonálisak, és jelentős mértékben csökkentik a korrelációt, melyet szemléltet az 5.2. ábra is.

A cél, hogy kettőnél több kimenetre is meghatározzak olyan vektorokat, melyek két kimenet közötti korrelátlanságot növelik. Az előző gondolatmenetből kiindulva a vektorok elemeinek értékészlete legyen 1 és  $-1$ . Erre több esetet is megvizsgáltam. Az első két kimenetre  $\mathbf{c}_1$  és  $\mathbf{c}_2$  vektorok továbbra sem változnak:

$$\mathbf{c}_1 = \begin{bmatrix} 1 & 1 & \dots & 1 \end{bmatrix},$$

$$\mathbf{c}_2 = \begin{bmatrix} 1 & -1 & 1 & -1 & \dots & 1 & -1 \end{bmatrix}.$$



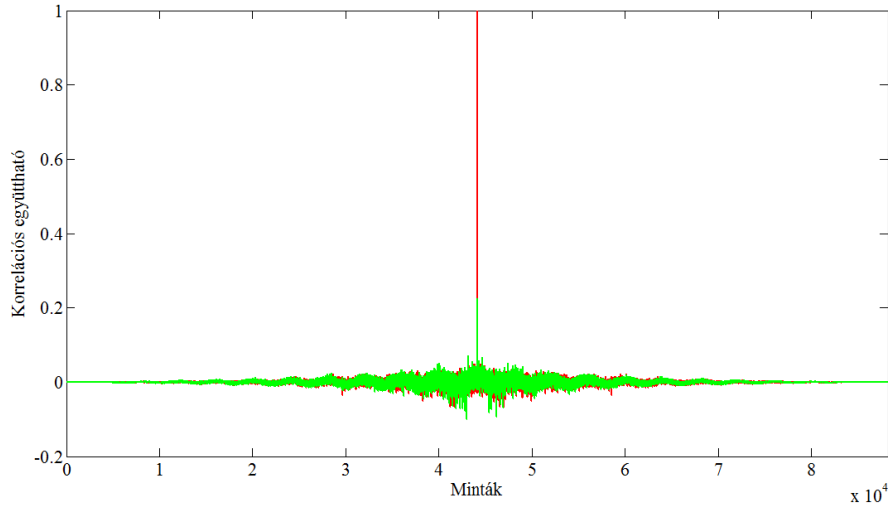
**5.2. ábra.** Keresztkorrelációs együtthatók sztereó hangrendszerénél

Először nézzük meg, milyen megoldást ad, ha a többi vektor minden harmadik, negyedik, stb. eleme -1 értékű. Ha az első kimenetet hasonlítom össze az  $i$ -edikkel, nagyobb  $i$ -re szorosabb kapcsolatot állapítottam meg. Például  $i = 5$  esetén a keresztkorreláció maximuma már körülbelül 0.75, de már  $i = 3$ -ra is 0.5 körüli értéket kapok. Ennél jobb megoldás az, hogy ha csupa egyesekből álló vektort elhagyom, így az  $i$ -edik csatornánál  $\mathbf{c}_i$ -t úgy adom meg, hogy minden  $i + 1$ -edik elem -1, a többi egy értékű. Nagy  $i$ -k esetére még így is egyre inkább összefüggő jeleket kapok. Négy hangszórós konfigurációra az eredmény még elfogadható. A 3-as és a 4-es hangszórójelekre a korreláció maximuma 0.43.

Különböző kimeneti jeleket kapok akkor is, ha a minden csatornára a késleltetővonalak kimeneteit véletlenszerűen szorzom meg 1-gyel vagy -1-gyel. Ekkor azt várom, hogy nagy valószínűséggel két tetszőlegesen választott csatorna korreláltsága kicsi lesz. Több kísérlet elvégzés után biztató eredményt kaptam. Legjobb esetben a keresztkorreláció maximuma 0.1-nél is kisebb, az egymástól való függés mértéke közel megegyezik azzal az esettel, amikor a két erősítésvektor ortogonális (5.3. ábra).

Leggyakrabban a csúcs értéke 0.1 és 0.3 tized közötti, ritkán előforduló eset az, ha 0.5-nél is nagyobb. Utóbbi körülbelül ötven kísérletből egyszer fordult elő.

Az előző megoldásokban két vektorra nem mindig teljesült az, hogy ortogonálisak. Ha szeretném, hogy bármelyik két vektor ortogonális legyen, akkor egy másik módszert kell alkalmazni  $\mathbf{c}_i$ -k definiálására. Ezt úgy tudom megtenni, hogy előállítok



**5.3. ábra.** *Kimenetek korreláltsága véletlenszerűen generált vektorokra*

egy speciális  $N \times N$  dimenziójú mátrixot, az úgynevezett Hadamard mátrixot [10]. A Hadamard mátrix elemei 1 és -1 értékűek lehetnek, és tetszőlegesen kiválasztott két sorából alkotott vektorra teljesül, hogy azok ortogonálisak. A zengető késleltetések száma 32, ezért egy 32-edrendű mátrix generálása szükséges. Egy  $2 \times 2$ -es Hadamard mátrix egy lehetséges megadása:

$$\mathbf{H} = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}. \quad (5.1)$$

Függetlenül a mátrix rendjétől igaz, ha  $\mathbf{H}$  Hadamard mátrix, az alábbi módon előállított mátrix is Hadamard mátrix:

$$\begin{bmatrix} \mathbf{H} & \mathbf{H} \\ \mathbf{H} & -\mathbf{H} \end{bmatrix}.$$

Egy  $2^k \times 2^k$  mátrixot tehát egyszerűen rekurzívan tudunk előállítani. A másodrendűből képezek negyedrendűt, a negyedrendűből nyolcadrendűt, és így tovább. Az eljárással most már rendelkezésemre áll harminckét egymásra ortogonális sorvektor, melyek megfelelnek a mátrix sorainak. Annyi sort kell kiválasztani, ahány kimeneti csatornánk van, ezek a sorok lesznek  $\mathbf{c}_i$  erősítésvektorok. Kézenfekvő  $i$  csatornára a mátrix első  $i$  sorát alkalmazni. A módszer nagy előnye, hogy az eddigiek közül ez biztosítja a legnagyobb mértékben a kimeneti jelek korrelátlanságát, viszont valamivel nagyobb memóriaméretet igényel a négyzetes mátrix miatt.

## 6. fejezet

# A VST plug-in

A Steinberg VST fejlesztőkörnyezet útmutatójának segítségével [7] C++ nyelven implementáltam a zenetű effektet. A fordító a kódból egy reverb.dll fájlt generál, ez maga a VST plug-in. Az effekt működik bármilyen VST-t támogató host programmal. Ha a host program csak sztereó kimeneti csatornát tud kezelni, akkor az a plug-in első két kimenetét használja. Az effektet az AudioMulch hangszerkesztő programmal teszteltem, mely gyakorlatilag korlátlan számú kimeneti csatornát támogat.

A modell késleltetővonal alapú, téglatest alakú termet feltételez, melynek méretei alapján határozza meg a korai reflexiók irányát, az általuk megtett út hosszát, és a szabad átlagos úthosszt is.

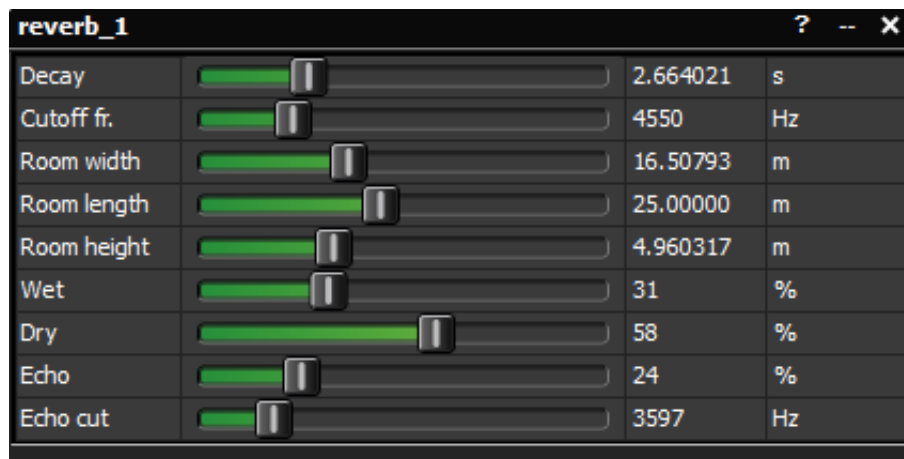
Az utözengést az egyszerűsített Feedback Delay Network valósítja meg, mely rendelkezik harminckét késleltetővonallal. A kimenetek korrelálatlanságának maximalizálásához a visszacsatolt késleltetések erősítését  $32 \times 32$  dimenziós Hadamard mátrix alkalmazásával határoztam meg. A veszteségi szűrőkkel módosítható lecsengési idő, és annak a frekvenciafüggése is, melyek külön-külön állítható paraméterek a VST plug-in-ban.

A korai visszaverődéseket a FIR szűrőnek megfelelően modelleztem, a reflexiók lát-szólagos irányát a Vector Base Amplitude Panning algoritmussal határoztam meg. A választott hangszóró-elrendezéstől függően alkalmaztam a két-, vagy háromdimenziós számítási módszert. A falak elnyelésének frekvenciafüggését egy elsőfokú aluláteresztő szűrő reprezentálja. A visszaverődéseket második rendig vettem figyelembe.

A hangszóró-elrendezést a channels.txt fájlban lehet megadni, melyet a host program a plug-in betöltésekor olvas be, melyet a c:\channels\channels.txt útvonalon keresztül tud beolvasni. A szöveges fájl három fontos paramétert tartalmaz. A CH\_NUM mellett szereplő szám a hangszórók számát adja meg. A DIM ad információt arról, hogy két- vagy háromdimenziós a konfiguráció. Végül a hangszórók x, y,

esetleg z irányvektorait adjuk meg, úgy, hogy egy hangszóróra egy sorban írjuk föl a koordinátákat, szóközzel elválasztva, a következő hangszóró paraméterei új sorban kezdődnek. Az x, y, z komponenseknek egységvektort kell alkotniuk. Ha nem készítünk inicializáló állományt, alapértelmezésben a zengető sztereó hangszóróelrendezéssel működik.

A plug-in kilenc állítható paraméterrel rendelkezik (6.1. ábra). Megválaszthatjuk a lecsengési időt (Decay) 0 és 10 s között, és a lecsengési időhöz tartozó vágási frekvenciát (Cutoff fr.). A teremméretek módosítására szolgál a Room width, a Room length és a Room height nevű paraméterek. Tetszőlegesen megváltoztathatjuk a direkt hang (Dry), a korai reflexiók (Echo), és az utózenge (Wet) intenzitását, továbbá a korai reflexiók frekvenciafüggő elnyelését az Echo cut paraméterrel módosíthatjuk.



Parameter	Value	Unit
Decay	2.664021	s
Cutoff fr.	4550	Hz
Room width	16.50793	m
Room length	25.00000	m
Room height	4.960317	m
Wet	31	%
Dry	58	%
Echo	24	%
Echo cut	3597	Hz

**6.1. ábra.** A zengető állítható paraméterei az Audiomulch hangszerkesztőben

## 7. fejezet

# Összefoglalás, értékelés

Szakedolgozatomban a zengés hangeffektussal foglalkoztam, ezen belül a mesterséges zengetés megvalósításának lehetőségeire fektettem a hangsúlyt. A bevezetőben szó esett a legfontosabb teremakusztikai fogalmakról, az effektek szoftveres megvalósításának elterjedéséről, illetve a hétköznapiakban is gyakran alkalmazott többcsatornás, úgynevezett surround hangrendszerekről.

Röviden megemlítettem a konvolúciós zengetést, kitérve az előnyeire, hátrányaira. A későbbi fejezetekben a késleltetővonal alapú konstrukciókkal foglalkoztam részletebben. Bemutattam a korai reflexiók modellezésének az elvét, realizálásának opcióit. A következő rész a késői zengetők konstrukcióiról szól, a fésű- és mindentáteresztő szűrőktől a Feedback Delay Network-ig.

A 3. fejezetben ismertettem az egyszerűsített Feedback Delay Network felépítését, melyet a VST zengetőben alkalmaztam. Megmutattam, hogy hogyan állítom be a veszteségi szűrőkkel a lecsengési időt és a vágási frekvenciát.

A 4. fejezetben a korai visszaverődések modellezésének a részleteit mutattam be. Kitértem a késleltetővonalak hosszának meghatározására, az irányok számítására, végül a reflexiók panorámázására a hangszórók között.

Az 5. fejezetben a Feedback Delay Network kimenetei jeleinek a hangfalakra történő kiosztásával foglalkoztam. Két megközelítést említettem meg: a késleltetővonalak kimeneteinek csoportokra bontását, és az erősítővektorok alkalmazását.

Végül a VST effekt konkrét megvalósításáról írtam. Bemutattam, hogyan állítható be a hangszóró-elrendezés, mik a módosítható paraméterek.

A zengetőt többféle hanggal is teszteltem. Ütős hangszerek vagy egyéb impulzusszerű hangok megszólaltatása (pl. taps) a legalkalmasabb arra, hogy pszichoakusztikai szempontból minősítsük az effektet. Az utózenge a veszteségi szűrőkkel együtt szépen szól, a lecsengési időt és a vágási frekvenciát megfelelően kombinálva hűen visszaadja egy valóságos terem hangzását. A kimenetek korrelálatlanságának a növelésével hallhatóan javult a térhatás, kellemes hangzást eredményezve.



A korai visszaverődéseket négy hangszórával teszteltem. Az érzés valóban az, mint-ha különböző irányból érkeznének meg a hangok. Kisebb teremméretekre ez is szépen szól. Egy  $20 \times 25 \times 6m$  nagyságú teremre hallás alapján még megfelelő, nagyobbakra már kicsit kellemetlen, visszhangszerű a hatás.

A plug-in legnagyobb problémája a számításigény. Mivel minden mintánál az összes korai reflexiót panorámázni kell a hangszórákra, ez jelentősen megnöveli számítógép által elvégzendő szorzások számát (mintánként több száz). Harmadrendű visszaverődéseket is figyelembe véve valós időben nem volt képes az effekt futtatására. A másodrendű esetén is módosítanom kellett a programot, hogy hatékonyabban végezze el a számításokat. Bár még most is nagyon sok műveletet kell elvégeznie a számítógépnek, még elfogadhatóan működik.

Elsősorban azon lehetne javítani, miképp módosítsam a plug-in forráskódját, hogy kevésbé terhelje a processzort. Egy másik fejlesztési lehetőség egy olyan szűrő tervezése, mely különböző anyagú falak elnyelésének valós frekvenciafüggését képes szimulálni.

# Irodalomjegyzék

- [1] [http://en.wikipedia.org/wiki/5.1\\_surround\\_sound](http://en.wikipedia.org/wiki/5.1_surround_sound). 5.1 surround sound.
- [2] <http://www.steinberg.net/en/company/aboutsteinberg.html>. About Steinberg.
- [3] <http://www.soundonsound.com/sos/dec99/articles/sonydres777.htm>. Sony DRE S777 Convolution Reverberator.
- [4] [http://www.kvraudio.com/news/vsl\\_releases\\_vienna\\_suite\\_v1\\_1\\_1301\\_incl\\_new\\_convolution\\_reverb\\_plug\\_in\\_11479](http://www.kvraudio.com/news/vsl_releases_vienna_suite_v1_1_1301_incl_new_convolution_reverb_plug_in_11479). Vienna Suite 1.1.
- [5] Wilfried Van Baelen, Tom Bert, Brian Claypool, and Tim Sinnaeve. Auro 3d: A new dimension in cinema sound. [http://www.barco.com/projection\\_systems/downloads/Auro-3D\\_v3.pdf](http://www.barco.com/projection_systems/downloads/Auro-3D_v3.pdf), 2011.
- [6] Balázs Bank. *Physics-based Sound Synthesis of String Instruments Including Geometric Nonlinearities*. PhD thesis, Budapest University of Technology and Economics, 2006. pages 39-41.
- [7] Steinberg Media Technologies GmbH. Vst plug-ins sdk documentation. <http://www.gersic.com/vstsdk/>, 2003.
- [8] M. Kahrs and K. Brandenburg. *Applications of Signal Processing to Audio and Acoustics*, chapter 3, pages 85–131. Kluwer, 2002. William G. Gardner: Reverberation Algorithms.
- [9] Ville Pulkki. Virtual sound source positioning using vector base amplitude panning. *Laboratory of Acoustics and Audio Signal Processing, Helsinki University of Technology, FIN-02015 HUT, Finland*, 1997.
- [10] Jennifer Seberry, Beata JWysocki, and Tadeusz A Wysocki. On some applications of hadamard matrices. *University of Wollongong Northfields Av., NSW 2522, Australia*, 2005.

- [11] Tarnóczy Tamás. *Teremakusztika*, pages 186–196, 461–466, 300–304. Akadémiai Kiadó, Budapest, 1986.
- [12] Anders Torger and Angelo Farina. Real-time partitioned convolution for ambisonics surround sound. *University of Parma, Industrial Engineering Dept. V.Sienze 181/A, 43100 PARMA, ITALY*, 2001.

# Függelék

## F.1. A VST plug-in C++ osztályai és függvényei

A C++ project az alábbi forrásfájlokat foglalja magában:

- values.h (konstans értékek és a channels osztály definiálása)
- channels.cpp (channel osztály fájlbeolvasó tagfüggvénye)
- EchoFilt.h, EchoFilt.cpp (EchoFilt osztály)
- matrix.h, matrix.cpp (matrix osztály)
- delay.h, delay.cpp (delay osztály)
- loss.h, loss.cpp (lp filter osztály)
- reverb.h, reverb.cpp (reverb osztály)
- main.cpp (a VST plugin main fájlja)

A values.h fájlban definiáltam a számításokhoz az alábbi konstansokat:

- $\pi$  értéke
- FDN késleltetővonalak száma
- korai visszaverődések maximális rendje
- hangsebesség
- mintavételi frekvencia
- referencia értékek az FDN késleltetővonalak hosszához
- átlagos késleltetővonalhossz referencia értéke
- hangszórók helyvektorai különböző hangszóró elrendezések esetén
- channels osztály, a channels.txt beolvasására

- dimenziók száma (channels.txt-ből)
- csatornák száma (channels.txt-ből)

#### A channels osztály függvénye:

- *void readCH()*: A channels.txt-ben megadott hangszóró-konfigurációt olvassa be

#### A mátrix osztály függvényei:

- *void setMatrix(float x, float y, float z)*: A korai visszaverődéseket panorámázza a hangszórókon. Paraméterként a korai visszaverődés irányába mutató vektort kapja meg.
- *void inverse()*: A VBAP alapján létrehozott mátrixot invertálja.
- *void setG()*: kiszámítja a VBAP alapján az egyes hangszórókon kijövő jel erősítését

#### A delay osztály függvényei:

- *void setRoomX(float x)*: Beállítja a téglalast alakú terem x hosszúsági paraméterét. Hasonlóan működnek a *setRoomY()* és a *setRoomZ()* függvények is.
- *void setEarlyDelay()*: A hangforrás és a hallgató pozíciója, és a terem mérete alapján kiszámítja a korai visszaverődések hosszát mintákban. Azt is megadja, hogy az adott visszaverődés hányadrendű.
- *void setMeanDelay()*: Kiszámítja az FDN késleltetővonalainak átlagos hosszát a terem méret alapján
- *long setReverbDelay(int index)*: Az aktuális átlagos késleltetővonalhossz és a referencia késleltetővonal hossz hányadosával skálázva kiszámítja az FDN egy késleltetővonalának hosszát. Az index paraméter jelöli, hányadik késleltetővonalról van szó.

#### Az lp filter osztály függvényei:

- *void setg(float decay, float fo)*: Beállítja a veszteségi szűrő *g* paraméterét. Paraméterként a lecsengési időt és  $f_0$ -t kapja meg.
- *void seta(long cutoff, float fo, long size, float b3)*: Beállítja a veszteségi szűrő *a* paraméterét. Átadott paraméterek: vágási frekvencia,  $f_0$ , mintavételi frekvencia,  $b_3$  paraméter.

### Az EchoFilt osztály függvényei:

- *void setg(float g)*: Beállítja a korai reflexiók  $g$  kisfrekvenciás erősítési tényezőjét.
- *void seta(long fc)*: Beállítja a szűrő  $a$  paraméterét  $fc$  határfrekvencia alapján.

### reverb osztály függvényei:

- *void setDelay()*: Az FDN késleltetővonalainak a hosszát és a veszteségi szűrők paramétereit állítja be.
- *void setDecay(float Decay)*: A paraméterként megadott lecsengési idő alapján módosítja a veszteségi szűrők paramétereit.
- *void setCutoff(float fCutoff)*: A vágási frekvenciától függően módosítja a veszteségi szűrő  $a$  paraméterét.
- *void setX(float x)*, *void setY(float y)*, *void setZ(float z)*: a terem méretének változásakor módosítja az ettől függő változókat
- *void Hadamard(int ord)*: rekurzív módon megad egy Hadamard mátrixot. Az  $ord$  paraméter értéke 2, azaz másodrendű mátrix a kiindulópont

## F.2. A forráskód és a VST plug-in linkje

*C++ forrás:*

<https://sourceforge.net/p/multireverb/reverbvst/24/tree/trunk/>

*reverb.dll:*

<https://sourceforge.net/projects/multireverb/files/latest/download>

## F.3. CD melléklet

A szakdolgozathoz mellékelt CD az alábbiakat tartalmazza:

- az ingyenes VST SDK 2.4-es változata
- az FDN-t, és a veszteségi szűrőket megvalósító matlab kódok
- a zengető VST plug-in C++ forráskódja
- a reverb.dll nevű VST plug-in
- a channels.txt fájl sztereó beállításokkal
- hangfájlok a teszteléshez